



TI-89 Titanium
Voyage™ 200
Calculadora Gráfica

Importante

Texas Instruments no ofrece garantía alguna, ya sea explícita o implícita, incluidas, sin limitarse a ellas, garantías implícitas de comerciabilidad o idoneidad para un uso concreto, en lo que respecta a los programas o manuales y ofrece dichos materiales únicamente “tal y como son”.

En ningún caso Texas Instruments será responsable ante ninguna persona por daños especiales, colaterales, accidentales o consecuentes relacionados o causados por la adquisición o el uso de los materiales mencionados, y la responsabilidad única y exclusiva de Texas Instruments, independientemente de la forma de acción, no sobrepasará el precio de compra del artículo o material que sea aplicable. Asimismo, Texas Instruments no puede hacerse responsable de las reclamaciones de cualquier clase contra el uso de dichos materiales por cualquier otra parte.

Antes de usar (ó ensamblar) el producto lea cuidadosamente este instructivo.

Índice de contenido

Introducción	v
Calculadoras gráficas TI-89 Titanium y Voyage™ 200.....	v
Cómo utilizar este manual.....	v
1 Primeros pasos	1
Primeros pasos.....	1
Teclas de la TI-89 Titanium y la Voyage™ 200.....	6
Valores de configuración de modo.....	15
Acceso a las órdenes mediante el Catalog (Catálogo).....	17
Pantalla Home (Principal) de la calculadora.....	20
Uso de Apps.....	22
Comprobación de la información de estado.....	30
Desactivación del escritorio de Apps.....	31
Uso del reloj.....	32
Uso de los menús.....	38
Uso de la pantalla dividida.....	45
Gestión de Apps y versiones de sistema operativo (SO).....	49
Conexión de la TI-89 Titanium o la Voyage™ 200 con otros dispositivos.....	51
Pilas.....	53
2 Presentaciones en pantalla	57
Realización de operaciones.....	57
Cálculo simbólico.....	63
Constantes y unidades de medida.....	65
Representación gráfica básica de funciones I.....	66
Representación gráfica básica de funciones II.....	69
Gráficas en paramétricas.....	72
Gráficas en polares.....	74
Representación gráfica de sucesiones.....	76
Representación gráfica en 3D.....	77
Representación gráfica de ecuaciones diferenciales.....	80
Temas complementarios de gráficos.....	83
Tablas.....	85
Pantalla dividida.....	86
Data/Matrix Editor.....	88
Gráficos estadísticos y de datos.....	89
Programación.....	96
Text Editor.....	98
Analizador numérico.....	100
Bases de numeración.....	102
Gestión de la memoria y de las variables.....	104

3	Actividades	109
	Análisis del problema poste-esquina.....	109
	Obtención de la solución de una ecuación de segundo grado.....	110
	Estudio de una matriz.....	112
	Estudio de $\cos(x) = \sin(x)$	113
	Cálculo del área mínima de un paralelepípedo.....	115
	Ejecución de un documento mediante Text Editor.....	116
	Descomposición de una función racional.....	118
	Estudio de estadísticas: Filtrado de datos por categorías.....	120
	Programa del CBL 2™/CBL™ para la TI-89 Titanium / Voyage™ 200.....	123
	Estudio de la trayectoria de una pelota de béisbol.....	125
	Visualización de raíces complejas de un polinomio de tercer grado.....	126
	Solución de un problema de interés compuesto.....	129
	Cálculo de amortizaciones.....	130
	Cálculo de factores racionales, reales y complejos.....	132
	Simulación de una extracción sin reemplazamiento.....	133
	Uso de vectores para determinar la velocidad.....	134
4	Conectividad	137
	Conexión de dos unidades.....	137
	Transmisión de variables, aplicaciones Flash y carpetas.....	139
	Transmisión de variables con control de programas.....	146
	Actualización del sistema operativo (SO).....	149
	Recopilación y transmisión de listas ID.....	153
	Compatibilidad entre TI-89 Titanium, Voyage™ 200, TI-89 y TI-92 Plus.....	155
5	Gestión de memoria y variables	157
	Comprobación y reinicio de la memoria.....	157
	Presentación de la pantalla VAR-LINK.....	158
	Presentación de información sobre variables en la pantalla Home (Principal).....	160
	Manejo de variables y carpetas con VAR-LINK.....	161
	Pegado de un nombre de variable en una aplicación.....	168
	Archivo y extracción de variables.....	169
	Mensaje Garbage Collection.....	171
	Error de memoria al acceder a una variable del archivo.....	173
A	Apéndice A: Funciones e instrucciones	177
	Localizador rápido.....	178
	Listado alfabético de operaciones.....	182
B	Apéndice B: Información general	311
	Información sobre productos, servicios y garantías de TI.....	311
	Índice Alfabético	313

Introducción

Calculadoras gráficas TI-89 Titanium y Voyage™ 200

Este manual ofrece información sobre dos potentes y avanzados dispositivos gráficos de Texas Instruments: la TI-89 Titanium y la Voyage™ 200 calculadora gráfica.

La calculadora TI-89 Titanium o la Voyage™ 200 calculadora gráfica llevan instaladas una serie de aplicaciones de software (Apps) para calculadoras con importantes funciones para la mayoría de asignaturas y diseños curriculares.

Gracias a la memoria Flash de sólo lectura (ROM) que poseen la TI-89 Titanium (4 megabytes [MB] disponibles) o la Voyage™ 200 (2,7 MB disponibles), puede instalar otras aplicaciones y aumentar así las capacidades del dispositivo. El proceso para instalar aplicaciones y actualizaciones del sistema operativo (SO) es idéntico al que utiliza para instalar software en un ordenador. Sólo necesita el software de TI Connect™ y un TI Connectivity Cable.

Con la interfaz de usuario gráfica (GUI) y el escritorio de Apps configurable de la TI-89 Titanium y la Voyage™ 200 es muy fácil organizar las Apps en categorías propias.

Puede ampliar la capacidad de la TI-89 Titanium o la Voyage™ 200 con distintos accesorios, como los sistemas Calculator-Based Laboratory™ (CBL 2™) y Calculator-Based Ranger™ (CBR™), el adaptador de vídeo TI-Presenter™ y el panel de proyección TI ViewScreen™.

Los sistemas CBL 2 y CBR permiten la recogida de datos reales y estáticos. El adaptador de vídeo TI-Presenter permite conectar la TI-89 Titanium o la Voyage™ 200 a dispositivos de reproducción y grabación de vídeo, como televisores, VCR, cámaras de vídeo y monitores de ordenador. Con el panel de proyección TI ViewScreen es posible proyectar una imagen ampliada de la pantalla de la TI-89 Titanium o la Voyage™ 200 para que pueda verla toda la clase.

Cómo utilizar este manual

Para facilitar el uso de este manual y aprovechar al máximo su rendimiento se han incorporado ciertas convenciones tipográficas.

- Símbolos de teclas que aparecen en el manual.
- La mayoría de las teclas sirven para realizar más de una función. Para utilizar las funciones secundarias presadas sobre las teclas, debe

pulsar primero [2nd], [alpha] o [◊]. Estas funciones adicionales aparecen en el manual entre paréntesis.

Por ejemplo, un procedimiento puede incluir esta secuencia de teclas para abrir un menú de caracteres especiales:

Pulse [2nd] [CHAR]. (Pulse y suelte la tecla [2nd], pulse luego la tecla [CHAR] que corresponde a la función secundaria de [◊].)

- La expresión *mantenga pulsada* se aplica a las órdenes de tecla que requieren pulsar dos teclas a la vez. Por ejemplo, la instrucción para oscurecer el contraste de la pantalla es:

Mantenga pulsada [◊] y pulse [+].

- La calculadora gráfica utiliza menús para acceder a muchas operaciones. En la mayor parte de los casos puede utilizar uno de estos dos métodos para acceder a las opciones de menú. Por ejemplo,

Pulse [F2] **9:Trig (9:Trigonometría)**

significa que puede elegir la opción **Trig** pulsando primero la tecla [F2] y luego las teclas **9** o [↻] tantas veces como sean necesarias para seleccionar **Trig**, y pulsar por último [ENTER].

Este manual incluye los capítulos siguientes:

Primeros pasos – Ofrece a estudiantes y educadores de todas las disciplinas educativas una descripción general de las operaciones básicas de la TI-89 Titanium y la Voyage™ 200.

Presentaciones en pantalla – Contiene una serie de ejemplos breves con procedimientos paso a paso, pulsaciones de tecla y pantallas de muestra.

Actividades – Incluye una serie de ejemplos más amplios con procedimientos para resolver, analizar y mostrar problemas matemáticos.

Conectividad – Indica los procedimientos para conectar una calculadora gráfica con otra o con un ordenador por medio del puerto USB o el puerto E/S, además de instrucciones para transmitir variables y aplicaciones y actualizar el sistema operativo.

Gestión de memoria y variables – Explica los procedimientos para gestionar las variables guardadas en la memoria de la calculadora gráfica y en el archivo de datos, un área de memoria protegida e independiente de la RAM (memoria de acceso aleatorio).

Referencia técnica – Incluye la sintaxis y los parámetros de acción de cada función e instrucción del sistema operativo, así como una lista

alfabética de operaciones, mensajes de error y otra información de referencia.

La restante información sobre el producto está disponible en formato electrónico. Este grupo completo de capítulos electrónicos está incluido en el CD-ROM que se suministra con la TI-89 Titanium o la Voyage™ 200. Esta misma información se encuentra disponible en línea y para descarga gratuita en la dirección: education.ti.com/guides

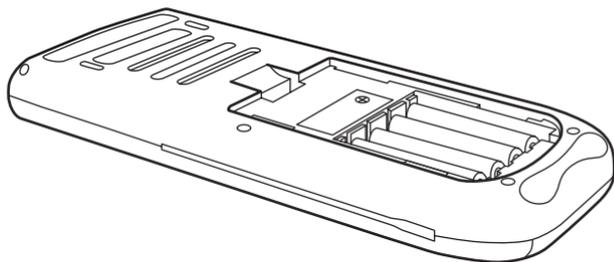
Primeros pasos

Primeros pasos

Instalación de las pilas AAA

La TI-89 Titanium utiliza cuatro pilas alcalinas AAA y una pila de óxido de plata (SR44SW o 303) de reserva. La Voyage™ 200 utiliza cuatro pilas alcalinas AAA y una pila de litio (CR1616 o CR1620) de reserva. Las unidades vienen con la pila de reserva ya instalada; las pilas alcalinas AAA se suministran con el producto correspondiente.

1. Retire la cubierta de las pilas, situada en la parte posterior de la unidad.
2. Extraiga de su embalaje las cuatro pilas alcalinas AAA suministradas con el producto, e introdúzcalas en el compartimento de las pilas siguiendo el diagrama de polaridad (+ y -) indicado en el mismo.



3. Vuelva a colocar la cubierta de las pilas en la unidad y encájela en su sitio.

Cómo encender la TI-89 Titanium o la Voyage™ 200 por primera vez

Tras instalar las pilas incluidas con la unidad, pulse **ON**. Aparece el escritorio de Apps.

Nota: Si la unidad inicializa las Apps preinstaladas, aparece una barra de progreso con el mensaje "Installation in progress . . . Do not interrupt!" en lugar del escritorio de Apps. Para evitar la pérdida de Apps, no quite las pilas durante la inicialización. (Puede volver a instalar las Apps desde el CD-ROM de recursos o desde education.ti.com.)

Barra de progreso



Ajuste del contraste

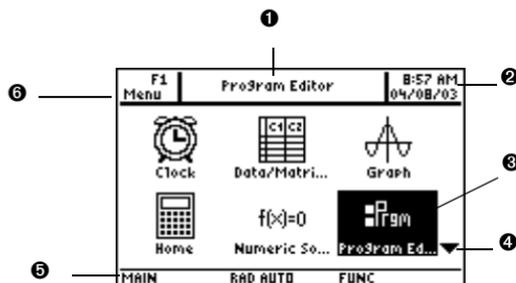
- Para aclarar la pantalla, mantenga pulsada \blacktriangle y pulse \square .
- Para oscurecer la pantalla, mantenga pulsada \blacktriangledown y pulse \oplus .



Escritorio de Apps

El escritorio de Apps constituye el punto de partida para utilizar la herramienta TI-89 Titanium o la Voyage™ 200. En él aparecen las Apps instaladas organizadas por categorías para facilitar su acceso. Desde el escritorio de Apps es posible:

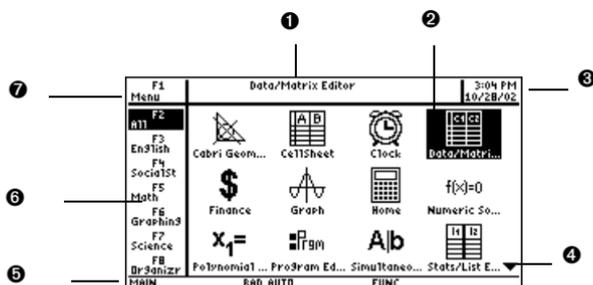
- Abrir Apps.
- Seleccionar y editar categorías de Apps.
- Ver todas las Apps instaladas en la unidad.
- Ver el nombre completo de la App resaltada.
- Ver y editar la fecha y la hora.
- Comprobar la información de la línea de estado.
- Ver información del modo de pantalla dividida.



Escritorio de Apps de la TI-89 Titanium

- 1 Ver el nombre completo de la App resaltada.
- 2 Ver la fecha y la hora.
- 3 Pulsar **[ENTER]** para abrir la App resaltada.
- 4 Desplazar el contenido hacia abajo para ver otras Apps.
- 5 Comprobar la información de la línea de estado.
- 6 Editar categorías.

Nota: Dado que la pantalla es menor, el escritorio de Apps de la TI-89 Titanium presenta ligeras diferencias con respecto al escritorio de Apps de la Voyage™ 200. Aunque la lista de Apps, incluida en la parte izquierda del escritorio de la Voyage™ 200, no aparece en el de la TI-89 Titanium, las categorías se seleccionan de igual modo en ambos productos.



Escritorio de Apps de la Voyage™ 200

- 1 Ver el nombre completo de la App resaltada.
- 2 Pulsar **[ENTER]** para abrir la App resaltada.
- 3 Ver la fecha y la hora.
- 4 Desplazar el contenido hacia abajo para ver otras Apps.
- 5 Comprobar la información de la línea de estado.
- 6 Seleccionar categorías de Apps.
- 7 Editar categorías.

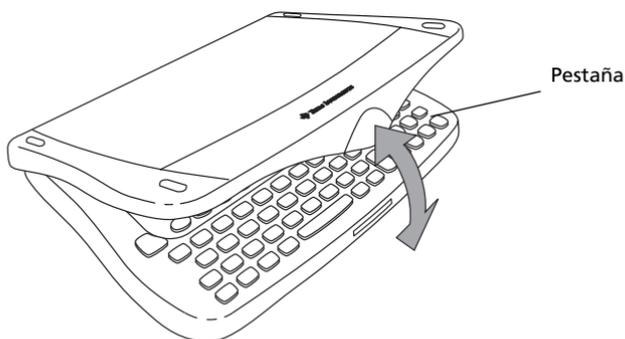
Para volver al escritorio de Apps en cualquier momento, pulse **[APPS]**. Aparece la última categoría seleccionada con la última App abierta resaltada.

Cómo retirar y poner la tapa (Voyage™ 200)

Para retirar la tapa:

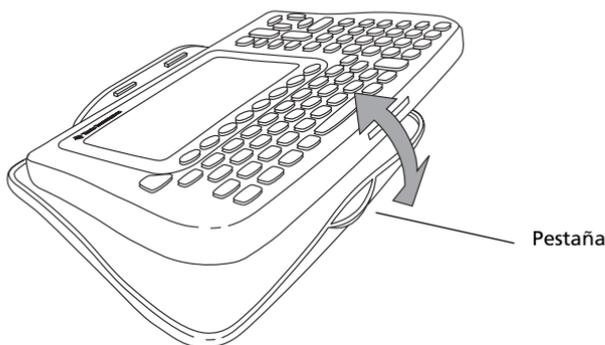
1. Sujete la unidad con una mano.
2. Coja la tapa por la pestaña con la otra mano.
3. Levante la tapa por la pestaña.

Para volver a poner la tapa, colóquela sobre la unidad con la pestaña hacia delante y encájela.



Cómo plegar la tapa

Para plegar la tapa, gírela hasta dejarla debajo de la unidad con la pestaña hacia delante y encájela.



Apagado de la unidad

Pulse **[2nd]** [OFF]. La próxima vez que encienda la unidad, aparecerá el escritorio de Apps con la misma configuración y el mismo contenido en la memoria. (Si hubiera desactivado el escritorio de Apps, aparecerá la pantalla Home (Principal) de la calculadora.)

Para apagar la TI-89 Titanium / Voyage™ 200, puede utilizar cualquiera de las teclas siguientes.

Pulse:	Descripción
2nd [OFF] (pulse 2nd y luego [OFF])	La función Constant Memory™ permite conservar los parámetros de configuración y el contenido de la memoria. <ul style="list-style-type: none">• No obstante, no podrá utilizar 2nd [OFF] si la pantalla muestra un mensaje de error.• Cuando encienda la TI-89 Titanium / Voyage™ 200 de nuevo, mostrará la pantalla Home (Inicio) o el escritorio de Apps (sea cual sea la última aplicación utilizada).
♦ [OFF] (pulse ♦ y luego [OFF])	Similar a 2nd [OFF] excepto en lo siguiente: <ul style="list-style-type: none">• Puede utilizar ♦ [OFF] si la pantalla muestra un mensaje de error.• Cuando encienda la TI-89 Titanium / Voyage™ 200 de nuevo, aparecerá exactamente igual que cuando la apagó.

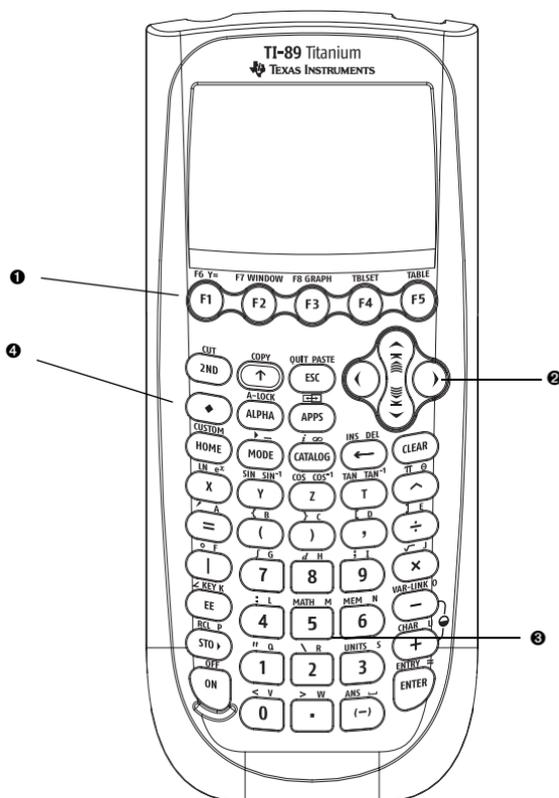
Nota: [OFF] es la función secundaria de la tecla **ON**.

La función Automatic Power Down™ (APD™) prolonga la duración de las pilas porque desactiva la unidad automáticamente si está inactiva durante varios minutos. Cuando se enciende la unidad tras APD:

- El contenido de la pantalla, el cursor y las posibles condiciones de error son exactamente las mismas que antes de APD.
- Se conserva la configuración y todo el contenido de la memoria.

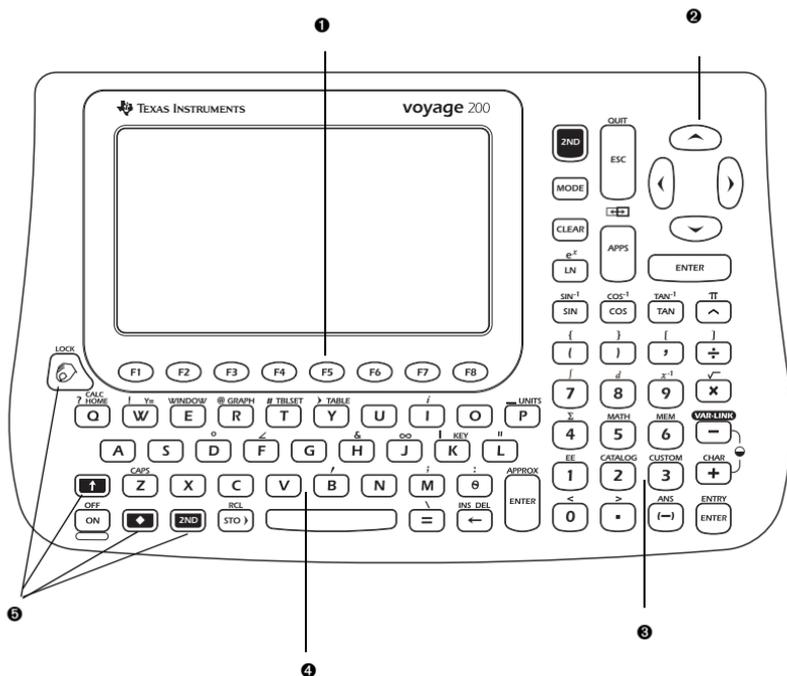
Nota: La función APD no se activará mientras haya un cálculo en progreso o se esté ejecutando un programa, a menos que lo detenga deliberadamente. Si el programa en ejecución está a la espera de que el usuario pulse una tecla, la función APD se activará automáticamente después de varios minutos de inactividad.

Teclas de la TI-89 Titanium y la Voyage™ 200



Teclas de la TI-89 Titanium

- 1** Teclas de función (**F1**–**F8**), abren los menús de la barra de herramientas y permiten acceder a Apps y editar categorías de Apps.
- 2** Teclas del cursor (**←**, **→**, **↑**, **↓**), mueven el cursor.
- 3** Teclado numérico, realizan operaciones matemáticas y funciones científicas.
- 4** Teclas de modificador (**2nd**, **↕**, **↑**), añaden funciones incrementando el número de las órdenes de tecla.



Teclas de la Voyage™ 200

- ❶ Teclas de función ($[F1]$ – $[F8]$), abren los menús de la barra de herramientas y permiten acceder a Apps y editar categorías de Apps.
- ❷ Teclas del cursor (\uparrow , \downarrow , \leftarrow , \rightarrow) mueven el cursor.
- ❸ Teclado numérico, realiza operaciones matemáticas y funciones científicas.
- ❹ El teclado QWERTY es similar al teclado de un ordenador.
- ❺ Teclas de modificador ($[2nd]$, \rightarrow , \uparrow , \leftarrow), añaden funciones incrementando el número de las órdenes de tecla.

Teclado QWERTY (Voyage™ 200 solamente)

Si está acostumbrado al teclado del ordenador, no le resultará difícil usar el teclado QWERTY de la Voyage™ 200, con algunas diferencias:

- Para escribir un solo carácter alfabético en mayúsculas, pulse \uparrow y la tecla de la letra.
- Para activar el bloqueo de mayúsculas, pulse $[2nd]$ [CAPS]. Para desactivar el bloqueo de mayúsculas, vuelva a pulsar $[2nd]$ [CAPS].

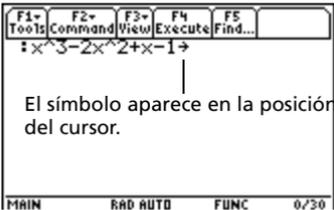
Introducción de caracteres especiales

Para introducir caracteres especiales se utiliza el menú CHAR (CARACTERES) y órdenes de teclado. El menú CHAR (CARACTERES) proporciona acceso a caracteres griegos, matemáticos, internacionales y otros especiales. Un mapa del teclado en pantalla muestra las posiciones de los métodos abreviados que sirven para introducir otros caracteres habituales.

Para seleccionar caracteres en el menú CHAR (CARACTERES):

1. Pulse $\boxed{2nd}$ [CHAR]. Aparece el menú CHAR (CARACTERES).
2. Use las teclas del cursor para elegir una categoría. Aparece un submenú con los caracteres de esa categoría.
3. Use las teclas del cursor para elegir un carácter y pulse \boxed{ENTER} .

Ejemplo: Introducir el símbolo de la flecha derecha (\rightarrow) en el Text Editor (Editor de texto).

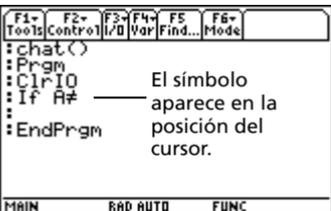
Pulse	Resultado
$\boxed{2nd}$ [CHAR]	
4	 <p>Baje para ver más caracteres.</p>
9 — o bien — Pulse \ominus varias veces para seleccionar 9:→ y pulse luego \boxed{ENTER}	 <p>El símbolo aparece en la posición del cursor.</p>

Para abrir el mapa del teclado, pulse $\boxed{\blacklozenge}$ [KEY]. Aparece el mapa del teclado.

Para escribir la mayoría de los caracteres, pulse  y la tecla correspondiente. Pulse  para cerrar el mapa.

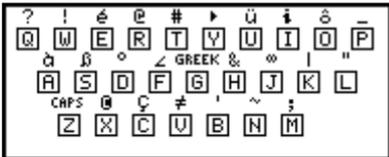
TI-89 Titanium

Ejemplo: Utilice el mapa del teclado para localizar el método abreviado correspondiente al símbolo “de desigualdad”, (\neq), e introduzca el símbolo en Program Editor (Editor de programas).

Pulse	Resultado
 [KEY]	
 	

Voyage™ 200

Ejemplo: Utilice el mapa del teclado para localizar el método abreviado correspondiente al símbolo “comillas”, (“), e introduzca el símbolo en Program Editor (Editor de programas).

Método abreviado	Caracteres generados
 [KEY]	

Método abreviado

Caracteres generados

2nd L



Teclas de modificador

Las teclas de modificador aumentan las funciones al incrementar el número de operaciones que pueden realizarse con el teclado. Para acceder a una función de modificador, pulse una tecla de modificador y luego la tecla de la operación correspondiente.

Teclas

Descripción

2nd

(Secundaria)

Accede a Apps, opciones de menú y otras operaciones. Las funciones secundarias se indican sobre las teclas correspondientes en el mismo color que la tecla **2nd**.

◆

(Diamante)

Accede a Apps, opciones de menú y otras operaciones. Las funciones de diamante se indican sobre las teclas correspondientes en el mismo color que la tecla **◆**.

↑

(Mayús)

Genera en mayúscula la letra de la siguiente tecla que se pulse. También se usa con **⏏** y **⏏** para resaltar caracteres al editar.

alpha

(Alpha; TI-89 Titanium solamente)

Permite escribir caracteres alfabéticos sin un teclado QWERTY. Los caracteres Alpha se imprimen por encima de las teclas correspondientes y en el mismo color que las teclas **alpha**.

☞

(Mano)
(Voyage 200 solamente)

Permite utilizar las teclas del cursor para manipular objetos geométricos. También se utiliza al dibujar sobre un gráfico.

Ejemplo: Acceder a la pantalla **VAR-LINK [AII] (CONEXIÓN DE VARIABLES [Todo])**, donde puede gestionar variables y Apps.

Pulse	Resultado
$\boxed{2nd}$ [VAR-LINK]	

Teclas de función

Las teclas de función sirven para realizar estas operaciones:

- En el escritorio de Apps, abrir Apps y seleccionar o editar categorías de Apps.
- En la pantalla Home (Principal) de la calculadora, abrir menús de barras de herramientas para seleccionar operaciones matemáticas.
- Dentro de Apps, abrir menús de barras de herramientas para seleccionar opciones de App.

Teclas del cursor

Al pulsar \leftarrow , \rightarrow , \uparrow o \downarrow el cursor se desplaza en la dirección correspondiente. Dependiendo de la App y de la tecla de modificador que se esté utilizando, $\boxed{2nd}$ o \blacklozenge , las teclas del cursor desplazan el cursor de forma diferente.

- \uparrow o \downarrow suben o bajan el cursor de línea en línea.
- $\boxed{2nd}$ \leftarrow o $\boxed{2nd}$ \rightarrow desplazan el cursor al principio o el final de una línea.
- $\boxed{2nd}$ \uparrow o $\boxed{2nd}$ \downarrow suben o bajan el cursor de página en página.
- \blacklozenge \leftarrow o \blacklozenge \rightarrow desplazan el cursor al principio o el final de una página.
- \uparrow y \leftarrow , \downarrow y \rightarrow , \downarrow y \leftarrow , o \uparrow y \rightarrow desplazan el cursor en diagonal. (Pulse simultáneamente el par de teclas indicado.)

Teclado numérico

El teclado numérico permite introducir números positivos y negativos.

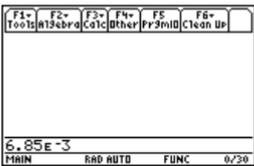
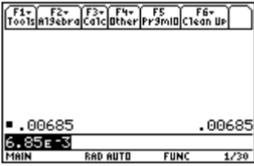
Para introducir un número negativo, pulse $\boxed{(-)}$ antes de escribir el número.

Nota: No confunda la tecla negativa ($\boxed{(-)}$) con la de sustracción ($\boxed{-}$).

Para introducir un número con notación científica:

1. Escriba los números que preceden al exponente. (Este valor puede ser una expresión).
2. Pulse \boxed{EE} (TI-98 Titanium) o $\boxed{2nd} \boxed{EE}$ (Voyage™ 200). El símbolo de exponente (E) aparece detrás de los números que ha introducido.
3. Escriba el exponente como un número entero de tres cifras como máximo. (Como refleja el ejemplo siguiente, puede usar un exponente negativo).

Ejemplo: En la pantalla Home (Principal) de la calculadora, introducir 0.00685 en notación científica.

Pulse	Resultado
<p>6 $\boxed{.}$ 8 5</p> <p>TI-89 Titanium: \boxed{EE}</p> <p>Voyage™ 200: $\boxed{2nd} \boxed{EE}$</p>	 <p>6.85E-3</p>
<p>$\boxed{(-)}$ 3</p>	
<p>\boxed{ENTER}</p>	 <p>.00685 .00685</p> <p>6.85E-3</p>

Otras teclas importantes

Comando del teclado	Descripción
<p>$\boxed{\blacklozenge} \boxed{[Y=]}$</p> <p>TI-89 Titanium solamente</p>	Muestra Y= Editor (Editor Y=).
<p>$\boxed{\blacklozenge} \boxed{[WINDOW]}$</p> <p>TI-89 Titanium solamente</p>	Muestra Window Editor (Editor de ventanas).
<p>$\boxed{\blacklozenge} \boxed{[GRAPH]}$</p> <p>TI-89 Titanium solamente</p>	Muestra la pantalla Graph (Gráfico).

Comando del teclado	Descripción
◆ [TBLSET] TI-89 Titanium solamente	Establece los parámetros de la pantalla Table (Tabla).
◆ [TABLE] TI-89 Titanium solamente	Muestra la pantalla Table (Tabla).
TI-89 Titanium: ◆ [CUT] ◆ [COPY] ◆ [PASTE] Voyage™ 200: ◆ X (cortar) ◆ C (copiar) ◆ V (pegar)	Permiten editar la información introducida mediante operaciones de cortar, copiar o pegar.
◆ S Voyage™ 200 solamente	Muestra el cuadro de diálogo SAVE COPY AS (GUARDAR COMO), donde debe seleccionar una carpeta y escribir un nombre de variable para guardar los datos introducidos en la pantalla.
◆ N Voyage™ 200 solamente	Crea un archivo nuevo.
◆ O Voyage™ 200 solamente	Abre el archivo que el usuario especifique.
◆ F Voyage™ 200 solamente	Muestra el cuadro de diálogo FORMATS (FORMATOS) o GRAPH FORMATS (FORMATOS DE GRÁFICO), donde se introduce información de formato para la App activa.
[APPS]	Muestra el escritorio de Apps.
◆ [APPS]	Con el escritorio de Apps desactivado, muestra el menú FLASH APPLICATIONS (APLICACIONES FLASH).
[2nd] [⇄]	Conmuta las dos últimas Apps elegidas.
[2nd] [CUSTOM]	Activa y desactiva el menú personalizado.

Comando del teclado	Descripción
$\boxed{2nd}$ $\boxed{\blacktriangleright}$	Convierte unidades de medida.
TI-89 Titanium: $\boxed{\blacklozenge}$ $\boxed{[-]}$	Designa una unidad de medida.
Voyage™ 200: $\boxed{2nd}$ $\boxed{[-]}$	
$\boxed{\leftarrow}$	Borra el carácter situado a la izquierda del cursor (retroceso).
$\boxed{\blacklozenge}$ $\boxed{[DEL]}$	Borra el carácter situado a la derecha del cursor.
$\boxed{2nd}$ $\boxed{[INS]}$	Alterna los modos de inserción y sobrescritura.
$\boxed{2nd}$ $\boxed{[MEM]}$	Muestra la pantalla MEMORY (MEMORIA).
TI-89 Titanium: $\boxed{[CATALOG]}$	Muestra una lista de las órdenes de la unidad.
Voyage™ 200: $\boxed{2nd}$ $\boxed{[CATALOG]}$	
$\boxed{2nd}$ $\boxed{[RCL]}$	Recupera el contenido de una variable.
$\boxed{[STO\blacktriangleright]}$	Almacena un valor en una variable.
$\boxed{2nd}$ $\boxed{[CHAR]}$	$\boxed{2nd}$ $\boxed{[UNITS]}$ Muestra el menú CHAR (CARACTERES), que proporciona acceso a caracteres griegos, internacionales acentuados y otros caracteres especiales.
$\boxed{2nd}$ $\boxed{[QUIT]}$	<ul style="list-style-type: none"> • En el modo de pantalla completa, muestra el escritorio de Apps. • En el modo de pantalla dividida, muestra la vista en pantalla completa de la App activa. • Con el escritorio de Apps desactivado, muestra la pantalla Home (Principal) de la calculadora.

Valores de configuración de modo

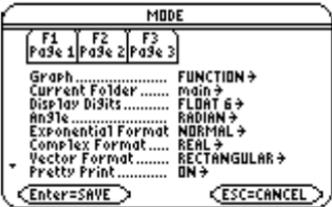
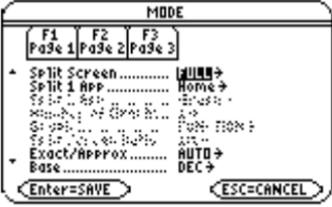
Los modos determinan cómo presentan e interpretan (plural) la información la TI-89 Titanium o la Voyage™ 200. Todos los números, incluidos los elementos de matrices y listas, se presentan con arreglo a los valores de configuración de modo activos. Cuando se apaga la TI-89 Titanium o la Voyage™ 200, la función Constant Memory™ conserva todos los valores de configuración de modo seleccionados.

Para ver los valores de configuración de modo de la TI-89 Titanium o la Voyage™ 200:

1. Pulse **[MODE]**. Aparece la página 1 del cuadro de diálogo MODE (MODO).
2. Pulse **[F2]** o **[F3]** para ver los modos incluidos en las páginas 2 ó 3.

Nota: Los modos que aparecen atenuados sólo están disponibles si se seleccionan otros valores de configuración de modo necesarios. Por ejemplo, el modo Custom Units (Unidades personalizadas) de la página 3 sólo está disponible si el modo Unit System (Sistema de unidades) está definido en CUSTOM (PERSONAL).

Visualización de valores de configuración de modo

Pulse	Resultado
[MODE]	
[F2]	

Pulse**Resultado**

F3

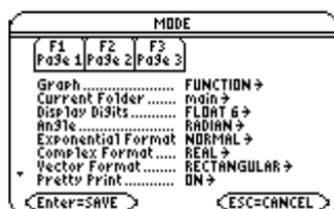


Cambio de los valores de configuración de modo

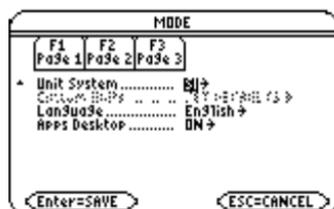
Ejemplo: Cambie el valor de configuración de modo Language (Idioma) a Spanish (*Español*).

Pulse**Resultado**

MODE



F3

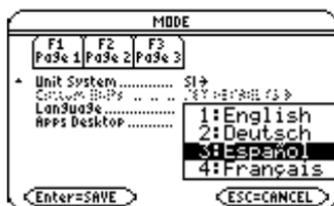
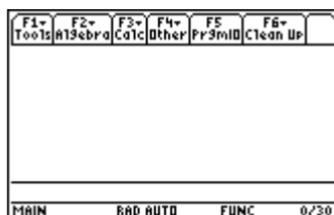


Baje al campo Language.

⏵



Pulse**Resultado**

Pulse \rightarrow y luego \leftarrow hasta resaltar la opción **3:Español**.**Nota:** La lista puede variar según los idiomas que haya instalados.**ENTER****ENTER****Nota:** Aparece la última App abierta (en este ejemplo, la pantalla Home (Principal) de la calculadora).Para devolver el modo Language (Idioma) a English, repita los mismos pasos, pero seleccione **1:English** en el campo Language (Idioma).

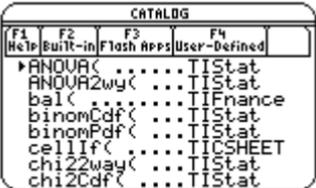
Acceso a las órdenes mediante el Catalog (Catálogo)

El Catalog (Catálogo) proporciona acceso a una lista de órdenes de la TI-89 Titanium o la Voyage™ 200, que incluye funciones, instrucciones y programas definidos por el usuario. Las órdenes se enumeran por orden alfabético. Las órdenes que no empiezan por una letra se hallan al final de la lista (&, /, +, -, etc.).

El sistema de ayuda de Catalog contiene información detallada sobre cada orden.

Las opciones que no son válidas en un determinado momento aparecen atenuadas. Por ejemplo, la opción de menú Flash Apps (F3) aparece atenuada si no hay ninguna aplicación Flash instalada en la TI-89 Titanium o la Voyage™ 200; la opción de menú User-Defined (Definido por el usuario) (F4) está atenuada si el usuario no ha creado ninguna función ni programa.

Nota: Al escribir una letra se va a la primera orden de la lista que comienza por dicha letra.

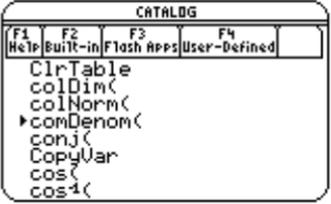
Pulse	Resultado
TI-89 Titanium: [CATALOG] Voyage™ 200: [2nd] [CATALOG] (muestra las órdenes intregradas en el programa)	
[F3] (muestra las órdenes de Flash Apps, si las hubiera)	
[F4] (muestra las órdenes definidas por el usuario, si las hubiera)	

Puede seleccionar órdenes en el Catalog (Catálogo) e insertarlas en la línea de entrada de la pantalla Home (Principal) de la calculadora, o bien pegarlos en otras Apps, como Y= Editor, Text Editor (Editor de texto) o CellSheet™.

Ejemplo: Insertar la orden **comDenom(** (**Denominador común(**) en la línea de entrada de la pantalla Home (Principal) de la calculadora.

Nota: Antes de seleccionar una orden, coloque el cursor donde desee que aparezca la misma.

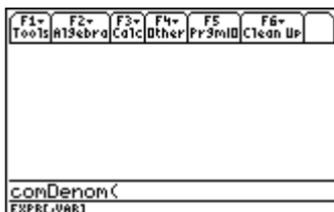
Al pulsar $\boxed{2\text{nd}}$ \odot se avanza por la lista del Catalog (Catálogo) de página en página.

Pulse	Resultado
TI-89 Titanium: $\boxed{\text{CATALOG}}$ $\boxed{\text{alpha}}$ C	
Voyage™ 200: $\boxed{2\text{nd}}$ $\boxed{\text{CATALOG}}$ C	

$\boxed{2\text{nd}}$ \odot

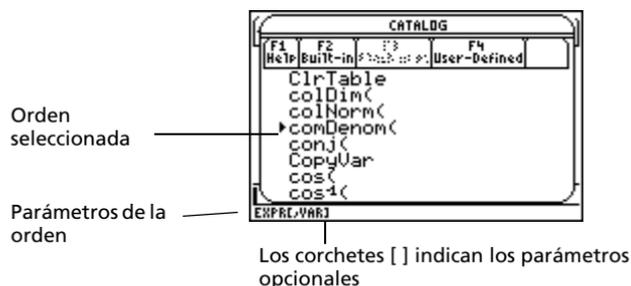
A continuación, pulse \odot repetidamente hasta que el puntero se encuentre en la función **comDenom(**.

$\boxed{\text{ENTER}}$



La línea de estado muestra tanto los parámetros necesarios como los opcionales de la orden seleccionada. Los parámetros opcionales se indican entre corchetes.

Nota: Al pulsar $\boxed{\text{F1}}$ también aparecen los parámetros de la orden seleccionada.



Para salir del Catalog (Catálogo) sin seleccionar ninguna orden, pulse $\boxed{\text{ESC}}$.

Pantalla Home (Principal) de la calculadora

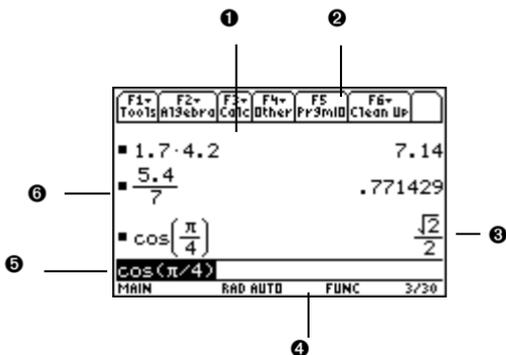
La pantalla Home (Principal) de la calculadora es el punto de partida para realizar operaciones matemáticas, lo que incluye ejecutar instrucciones, realizar cálculos y ver resultados.

Para acceder a la pantalla Home (Principal) de la calculadora, pulse:

TI-89 Titanium: **[HOME]**

Voyage™ 200: **[◀] [CALC HOME]**.

También puede acceder a la pantalla Home (Principal) de la calculadora desde el escritorio de Apps resaltando el icono Home (Principal) y pulsando **[ENTER]**.



- 1 El área de historia muestra una lista de todos los pares entrada/respuesta introducidos.
- 2 Fichas de presentación de menús para seleccionar listas de operaciones. Pulse **[F1]**, **[F2]**, y sucesivamente, para mostrar los menús.
- 3 Lugar en que aparece el resultado de la última entrada. (Los resultados no se muestran en la línea de entrada.)
- 4 Línea de estado que muestra el estado actual de la TI-89 Titanium o la Voyage™ 200.
- 5 Línea de entrada con la entrada actual.
- 6 Lugar en que aparece la entrada anterior.

Para volver al escritorio de Apps desde la pantalla Home (Principal) de la calculadora, pulse **[APPS]**.

Acerca del área de historia

El área de historia muestra un máximo de ocho pares entrada/respuesta según la complejidad y longitud de las expresiones. Cuando dicha área se llena, la información asciende desapareciendo por la parte superior de la pantalla. El área de historia sirve para:

- Revisar entradas y respuestas anteriores. Use las teclas del cursor para ver las entradas y respuestas que han quedado fuera de la pantalla.
- Recuperar o pegar automáticamente una entrada o respuesta anterior en la línea de entrada para reutilizarla o editarla. Para obtener más información, consulte el módulo electrónico *Funcionamiento de la calculadora* de la TI-89 Titanium o la Voyage™ 200.

El cursor, que suele permanecer en la línea de entrada, puede trasladarse al área de historia. En la tabla siguiente se explica cómo desplazar el cursor por el área de historia.

Para	Haga lo siguiente
Ver entradas/respuestas que han quedado fuera de la pantalla	Desde la línea de entrada, pulse \ominus para resaltar la última respuesta. Siga utilizando \ominus para trasladar el cursor a lo largo de las respuestas y entradas dentro del área de historia.
Ir al par entrada/respuesta más antiguo o más reciente	Si el cursor se encuentra en el área de historia, pulse $\blacklozenge \ominus$ o $\blacklozenge \omin�$.
Ver una entrada o respuesta demasiado larga para una sola línea (\blacktriangleright aparece al final de la línea)	Traslade el cursor a la entrada o respuesta. Use \uparrow o \downarrow para desplazarse a la izquierda o la derecha y $2^{nd} \uparrow$ o $2^{nd} \downarrow$ para ir al principio o al final.
Devolver el cursor a la línea de entrada	Pulse \boxed{ESC} , o bien pulse \ominus hasta que el cursor vuelva a la línea de entrada.

Interpretación de la información histórica de la línea de estado

El indicador de historia de la línea de estado proporciona información sobre los pares entrada/respuesta. Por ejemplo:

Si el cursor se halla en la línea de entrada:

Número total de pares guardados actualmente	_____ 8/30 _____	Máximo número de pares que pueden guardarse
---	------------------	---

Si el cursor se halla en el área de historia:

Número de par de la entrada/respuesta resaltada	_____ 8/30 _____	Número total de pares guardados actualmente
---	------------------	---

Modificación del área de historia

Para cambiar el número de pares que pueden guardarse:

1. Desde la pantalla Home (Principal) de la calculadora, pulse **[F1]** y seleccione **9:Format**.
2. Pulse **⬇** y use **⬅** o **➡** para resaltar el nuevo número.
3. Pulse **[ENTER]** **[ENTER]**.

Para limpiar el área de historia y borrar todos los pares guardados:

- Desde la pantalla Home (Principal) de la calculadora, pulse **[F1]** y seleccione **8:Clear Home (8:Borrar Principal)**.
– o bien –
- Introduzca **ClrHome (BorPrinc)** en la línea de entrada de la pantalla Home (Principal) de la calculadora.

Para borrar un par entrada/respuesta, traslade el cursor a la entrada o la respuesta y pulse **⬅** o **[CLEAR]**.

Uso de Apps

La TI-89 Titanium o la Voyage™ 200 organizan las Apps por categorías en el escritorio de Apps. Para seleccionar una categoría, pulse una tecla de función (de **[F2]** a **[2nd]** **[F8]**, o bien de **[F2]** a **[F8]**). Los iconos de App correspondientes a la categoría elegida aparecen en el escritorio de Apps.

Nota: Si el nombre que aparece bajo un icono del escritorio de Apps está truncado, resalte el icono mediante las teclas del cursor. Así podrá ver el nombre completo en la parte superior del escritorio de Apps.

Cómo abrir Apps

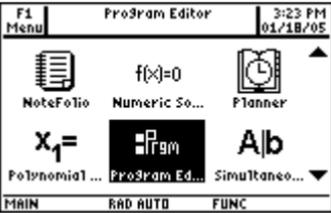
Use las teclas del cursor para resaltar el icono de Apps en el escritorio de Apps y pulse **[ENTER]**. Se abre directamente la App o aparece un cuadro de diálogo. El cuadro de diálogo más frecuente contiene las siguientes opciones de la App:

Nota: En la TI-89 Titanium o la Voyage™ 200, el término general *variable* designa los archivos de App que crea el usuario.

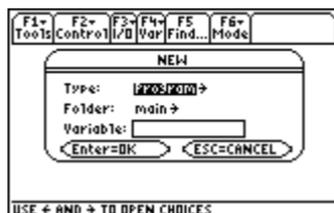
Opción	Descripción
Current (Actual)	Muestra la pantalla que había visible la última vez que se accedió a la App. Si no hay ninguna variable de App actual, aparece el cuadro de diálogo New (Nuevo).
Open (Abrir)	Permite abrir un archivo existente.
New (Nuevo)	Crea un archivo nuevo con el nombre que se escriba en el campo correspondiente.

Seleccione una opción, introduzca la información necesaria y pulse **[ENTER]**. Aparece la App.

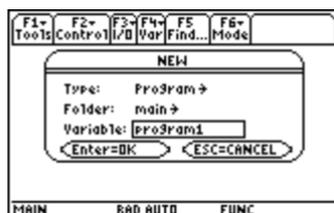
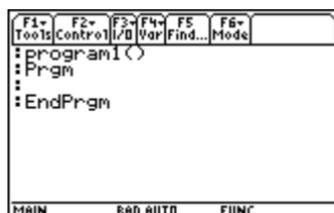
Ejemplo: Crear un programa nuevo con el Program Editor (Editor de programas).

Pulse	Resultado
Use las teclas del cursor para resaltar  Program Ed...	
[ENTER]	
3	

Pulse**Resultado**

[ENTER]

⏴ ⏵
program 1

**[ENTER] [ENTER]**

La variable de programa recién creada, *program1*, se guarda en la carpeta main (principal).

Cómo volver al escritorio de Apps desde una App

Pulse **[APPS]**. Los iconos de la última categoría de Apps elegida aparecen en el escritorio de Apps con el icono correspondiente a la última App resaltado.

También puede volver al escritorio de Apps pulsando **[2nd] [QUIT]** en el modo de pantalla completa. En el modo de pantalla dividida, pulse **[2nd] [QUIT]** dos veces.

Para volver a la última App abierta desde el escritorio de Apps, pulse **[2nd] [+]**.

Selección de una categoría de Apps

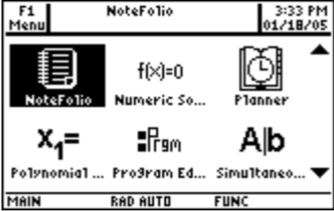
En la TI-89 Titanium, los nombres de categoría de Apps aparecen sólo en la opción **F1 Menu**. Para seleccionar una categoría de Apps, pulse **F1 2:Select Category** y utilice las teclas del cursor para resaltar una categoría de Apps; a continuación, pulse **ENTER** para seleccionar la categoría seleccionada. También puede utilizar los métodos abreviados de las teclas de función para seleccionar una categoría mediante el teclado (utilice la tecla **2nd** si fuese necesario). Los iconos de App de la categoría seleccionada aparecen en el escritorio de Apps

En la Voyage™ 200, los nombres de categoría de Apps aparecen en el lado izquierdo del escritorio de Apps. Para seleccionar una categoría de Apps, pulse la tecla de función correspondiente (indicada sobre el nombre de la categoría en el escritorio de Apps).

Los iconos de App de la categoría seleccionada aparecen en el escritorio de Apps..

Tecla	Descripción
F2 All (Todo)	Se ven los iconos de todas las Apps instaladas. No personalizable.
F3 English (Inglés)	Categoría personalizable. English (Inglés) es el valor predeterminado.
F4 SocialSt (EstSoc)	Categoría personalizable. SocialSt (EstSoc) (Estudios sociales) es el valor predeterminado.
F5 Math (Matemáticas)	Categoría personalizable. Math (Matemáticas) es el valor predeterminado.
2nd F6 Graphing (Representación gráfica) o bien F6 Graphing (Representación gráfica)	Categoría personalizable. Graphing (Representación gráfica) es el valor predeterminado.
2nd F7 Science (Ciencia) o bien F7 Science (Ciencia)	Categoría personalizable. Science (Ciencia) es el valor predeterminado.
2nd F8 Organizr (Organizador) o bien F8 Organizr (Organizador)	Categoría personalizable. Organizr (Organizador) es el valor predeterminado.

Ejemplo: Seleccione la categoría All (Todo).

Pulse	Resultado
[F2]	

Si selecciona una categoría de Apps que no contiene ninguna App, un mensaje confirma que dicha categoría está vacía y apunta al menú **[F1] 1:Edit Categories (1:Editar categorías)**, donde puede añadir métodos abreviados de App para la categoría. (La personalización de las categorías del escritorio de Apps se explica en *Personalización de categorías de Apps* a continuación.)

Pulse **[ENTER]** o **[ESC]** para eliminar el mensaje y volver al escritorio de Apps.

Personalización de categorías de Apps

La TI-89 Titanium o la Voyage™ 200 organiza las Apps en siete categorías, seis de las cuales pueden ser personalizadas con arreglo a sus necesidades particulares. (La categoría All (Todo) contiene todas las Apps instaladas y no puede editarse.)

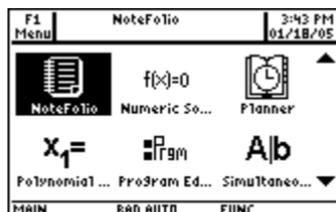
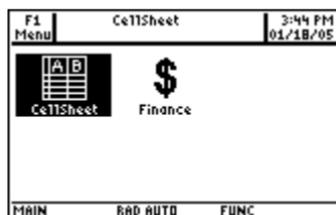
Para personalizar las categorías **[F3]** a **[2nd] [F8]** () o **[F8]** () de Apps:

1. Seleccione **[F1] 1:Edit Categories (1:Editar categorías)**. Un submenú presenta los nombres de las seis categorías de Apps personalizables. (No aparece la categoría All (Todo).)
2. Resalte una categoría de Apps y pulse **[ENTER]**. Aparece el cuadro de diálogo Edit Categories (Editar categorías) con una lista de las Apps instaladas y un cuadro de texto con el nombre de la categoría resaltada.
3. Para cambiar el nombre de la categoría de Apps, escriba el que prefiera.
Nota: Introduzca un nombre con ocho caracteres como máximo, incluidas letras mayúsculas o minúsculas, números, signos de puntuación y caracteres acentuados.
4. Para añadir o suprimir un método abreviado de App de la categoría, pulse **⊖** hasta resaltar el cuadro adjunto a la App y pulse **⊕** para añadir o suprimir la marca de selección (✓).

5. Para guardar los cambios y regresar al escritorio de Apps, pulse **ENTER**.

Ejemplo: Sustituir la categoría Social Studies (Estudios sociales) por Business (Empresariales) y añada los métodos abreviados de las Apps CellSheet™ y Finance.

Pulse	Resultado
F1	
↓	
2 - 0 - ENTER	
TI-89 Titanium: 2nd [a-lock] ↑ Business Voyage™ 200: ↑ Business	

Pulse**Resultado****ENTER****F4**

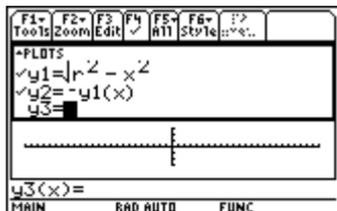
Apps abiertas y estado de pantalla dividida

La TI-89 Titanium o la Voyage™ 200 permiten dividir la pantalla para ver dos Apps simultáneamente. Por ejemplo, puede tener visibles al mismo tiempo las pantallas de Y= Editor y Graph (Gráf) para ver la lista de funciones y cómo se representan gráficamente.

Seleccione el modo Split Screen (Pantalla dividida) en la página 2 de la pantalla MODE (MOD0). La TI-89 Titanium o la Voyage™ 200 presentan las Apps seleccionadas en la vista de pantalla dividida, tal como se ilustra.

Divida la pantalla horizontalmente (arriba-abajo) o verticalmente (izquierda-derecha).

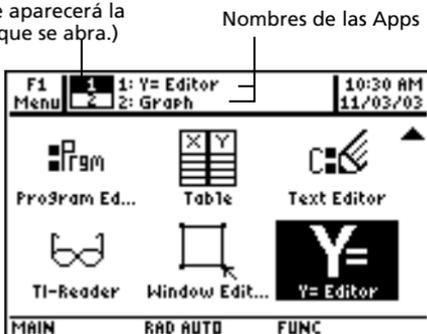
Pantalla dividida arriba-



Para volver al escritorio de Apps, pulse [APPS]. El estado de pantalla dividida aparece en la parte superior del escritorio de Apps con los nombres de las Apps abiertas y las partes de la pantalla donde aparece cada una de ellas. El símbolo de flecha (►) apunta a la pantalla donde aparecerá la siguiente App que se abra. En el modo de pantalla completa el estado de pantalla dividida no aparece en el escritorio de Apps.

Nota: El escritorio de Apps siempre aparece en la vista de pantalla completa.

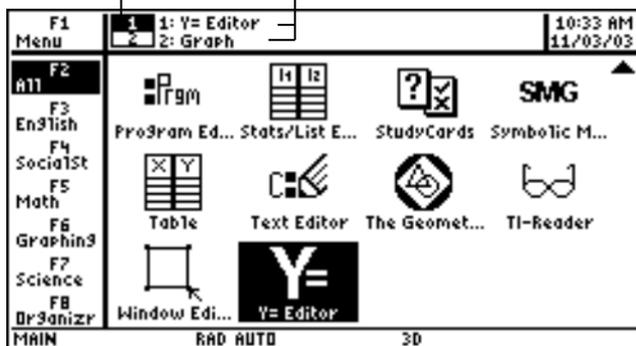
Estado de pantalla dividida (el resaltado indica la parte de la pantalla donde aparecerá la siguiente App que se abra.)



Indicadores de pantalla dividida en el escritorio de Apps de la TI-89 Titanium

Estado de pantalla dividida (el resaltado indica la parte de la pantalla donde aparecerá la siguiente App seleccionada.)

Nombres de las Apps

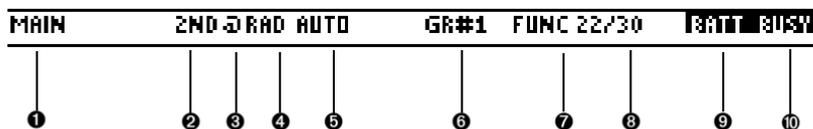


Indicadores de pantalla dividida en el escritorio de Apps de la Voyage™ 200

Hay más información disponible sobre el uso de las pantallas divididas. (Para obtener más información, consulte el capítulo electrónico *Pantallas divididas*.)

Comprobación de la información de estado

La línea de estado, situada en la parte inferior de la pantalla, muestra información sobre el estado actual de la TI-89 Titanium o la Voyage™ 200.



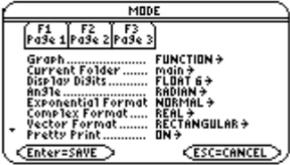
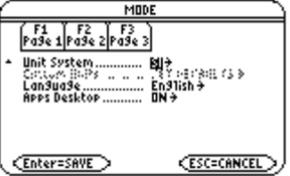
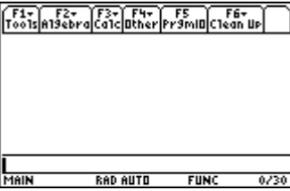
Indicador	Significado
① Carpeta actual	Nombre de la carpeta seleccionada (la carpeta predeterminada es Main (Principal).)
② Tecla de modificador	Tecla de modificador seleccionada ([2nd] , [◀] , [↑]), en su caso.
③ Tecla de modificador de mano (sólo para Voyage™ 200).	Se ha seleccionado la tecla de modificador [◀] . (Sólo para Voyage™ 200)

Indicador	Significado
④ Modo de Ángulo	Unidades en las que se van a mostrar e interpretar los valores de ángulo (RAD, DEG, GRAD)
⑤ Modo Exact/Approx	Modo en que se calculan y presentan las respuestas (AUTO, EXACT (EXACTO), APPROX (APROX))
⑥ Número de gráfico	A Gráfico activo de dos gráficos independientes en el modo de pantalla dividida (GR#1, GR#2)
⑦ Modo Gráf	Tipo de gráfico seleccionado para su representación (FUNC, PAR, POL, SEQ (SUC), 3D, DE (ED))
⑧ Pares entrada/respuesta	22/30-Número de pares entrada/respuesta (el número predeterminado es 30 y el máximo 99) en el área de historia de la pantalla principal de la calculadora.
⑨ Sustituir pilas	Aparece cuando las pilas están gastadas (BATT (PILA)). Si BATT (PILA) está resaltada sobre fondo negro, cambie las pilas tan pronto como pueda (BATT).
⑩ Ocupado/Pausa, Variable bloqueada/archivada	BUSY (OCUPADO) -Se está realizando un cálculo o una representación gráfica PAUSE (PAUSA) -El usuario ha detenido una representación gráfica o un programa La variable ■ abierta en el editor actual está bloqueada o archivada y no puede modificarse

Desactivación del escritorio de Apps

Puede desactivar el escritorio de Apps desde el cuadro de diálogo MODE (MODO). Si lo hace, ha de abrir Apps desde el menú APPLICATIONS (APLICACIONES). Para abrir el menú APPLICATIONS (APLICACIONES), pulse **APPS**.

Ejemplo: Desactivar el escritorio de Apps.

Pulse	Resultado
MODE	 <p>A screenshot of the MODE menu. The menu items are: Graph, Current Folder, Display Digits, ANS mode, Exponential Format, Complex Format, Vector Format, and Fractions Print. The 'Apps Desktop' option is highlighted with a cursor. At the bottom, there are buttons for 'Enter=SAVE' and 'ESC=CANCEL'.</p>
F3	 <p>A screenshot of the MODE menu. The 'Unit System' option is highlighted with a cursor. At the bottom, there are buttons for 'Enter=SAVE' and 'ESC=CANCEL'.</p>
Left Arrow, Down Arrow, Up Arrow, Right Arrow	 <p>A screenshot of the MODE menu. The 'Apps Desktop' option is highlighted, and a sub-menu is shown with 'ON' selected. At the bottom, there are buttons for 'Enter=SAVE' and 'ESC=CANCEL'.</p>
ENTER ENTER	 <p>A screenshot of the Home screen of the calculator. The screen shows the top status bar with function keys (F1-F6) and the bottom status bar with 'MAIN', 'RAD AUTO', 'FUNC', and '0/30'.</p>

Nota: Aparece la última App abierta (en este ejemplo, la pantalla Home (Principal) de la calculadora).

Para activar el escritorio de Apps, repita el procedimiento, pero seleccione ON en el campo de modo Desktop (Escritorio) de Apps. Para volver al escritorio de Apps desde la pantalla Home (Principal) de la calculadora, pulse **APPS**.

Uso del reloj

El cuadro de diálogo **CLOCK (RELOJ)** sirve para ajustar la fecha y la hora, seleccionar el formato de presentación del reloj y activarlo y desactivarlo.

El reloj está activado de forma predeterminada. Si lo desactiva, se atenuarán todas las opciones del cuadro de diálogo CLOCK (RELOJ), salvo Clock ON/OFF (Reloj ON/OFF).

▼ indica desplazamiento hacia abajo para las opciones de día y activación/desactivación del reloj



Visualización del cuadro de diálogo CLOCK (RELOJ)

1. Use las teclas del cursor para resaltar el icono de Clock (Reloj) en el escritorio de Apps.
2. Pulse **[ENTER]**. Aparece el cuadro de diálogo CLOCK (RELOJ) con el campo Time Format (Formato de hora) resaltado.

Nota: Como el cuadro de diálogo CLOCK (RELOJ) muestra la configuración activa en el momento de abrir el cuadro de diálogo, tal vez deba actualizar la hora antes de salir.

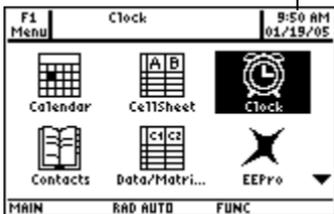
Ajuste de la hora

1. Pulse **[↑]** para abrir la lista de formatos de tiempo.
2. Pulse **[↓]** o **[↑]** para resaltar una opción y después pulse **[ENTER]**. El formato elegido aparece en el campo Time Format (Formato de tiempo).
3. Pulse **[↓]** para resaltar el campo Hour (Hora).
4. Escriba la hora y pulse **[↓]** para resaltar el campo Minute (Minuto).
5. Escriba los minutos.
6. Si se usa el formato de 24 horas, siga en el paso 9.
– o bien –
Si se usa el formato de 12 horas, pulse **[↓]** para resaltar el campo AM/PM.
7. Pulse **[↑]** para abrir la lista de opciones de AM/PM.
8. Pulse **[↓]** o **[↑]** para resaltar una opción de AM/PM y luego pulse **[ENTER]**. Aparece la opción de AM/PM seleccionada.
9. Ajuste la fecha (el procedimiento se explica en *Ajuste de la fecha*).
– o bien –
Pulse **[ENTER]** para guardar la configuración y salir. La hora se actualiza en la esquina superior derecha del escritorio de Apps.

Ajuste de la fecha

1. Pulse \leftarrow o \rightarrow hasta resaltar el campo Date Format (Formato de fecha).
2. Pulse \downarrow para abrir la lista de formatos de fecha.
3. Pulse \leftarrow o \rightarrow para resaltar una opción y después pulse **ENTER**. El formato elegido aparece en el campo Date Format (Formato de fecha).
4. Pulse \leftarrow para resaltar el campo Year (Año).
5. Escriba el año y pulse \downarrow para resaltar el campo Month (Mes).
6. Pulse \downarrow para abrir la lista de meses.
7. Pulse \leftarrow o \rightarrow para resaltar una opción y después pulse **ENTER**. El mes elegido aparece en el campo Month (Mes).
8. Pulse \leftarrow para resaltar el campo Day (Día).
9. Escriba el día y pulse **ENTER** **ENTER** para guardar la configuración y salir. La fecha se actualiza en la esquina superior derecha del escritorio de Apps.

Ejemplo: Ajustar la fecha y la hora en 19/10/02 (19 de octubre de 2002) a las 1:30 p.m.

Pulse	Resultado
Use las teclas del cursor para resaltar  Clock	Fecha y hora 
ENTER	

Pulse**Resultado**

⏮ 1 ⏭

CLOCK

Time Format: 12 Hour →

Hour: 1

Minute: 0

AM/PM: AM →

Date Format: MM/DD/YY →

Year: 1997

Month: January →

Enter=OK ESC=CANCEL

3 0 ⏭

CLOCK

Time Format: 12 Hour →

Hour: 1

Minute: 30

AM/PM: AM →

Date Format: MM/DD/YY →

Year: 1997

Month: January →

Enter=OK ESC=CANCEL

▶ ⏭

CLOCK

Time Format: 12 Hour →

Hour: 1

Minute: 30

AM/PM: 1: AM

Date Format: 2: PM

Year: 1997

Month: January →

Enter=OK ESC=CANCEL

[ENTER] ⏭

CLOCK

Time Format: 12 Hour →

Hour: 1

Minute: 30

AM/PM: PM →

Date Format: MM/DD/YY →

Year: 1997

Month: January →

Enter=OK ESC=CANCEL

▶ ⏭

CLOCK

Time Format: 12 Hour →

Hour: 1: MM/DD/YY

Minute: 2: MM/DD/YY

AM/PM: 3: MM. DD. YY

Date Format: 4: DD. MM. YY

Year: 5: YY. MM. DD

Month: 6: MM-DD-YY

7: DD-MM-YY

8: YY-MM-DD

Enter=OK

Pulse

Resultado

ENTER ↵

CLOCK

Time Format: 12 Hour ↵
Hour: 1
Minute: 30
AM/PM: PM ↵
Date Format: DD/MM/YY ↵
Year: 1997
Month: January ↵

Enter=OK ESC=CANCEL

2 0 0 2

CLOCK

Time Format: 12 Hour ↵
Hour: 1
Minute: 30
AM/PM: PM ↵
Date Format: DD/MM/YY ↵
Year: 2002
Month: January ↵

Enter=OK ESC=CANCEL

↵ ↵

CLOCK

Time Format: 12 Hour ↵
Hour: 1
Minute: 30
AM/PM: PM ↵
Date Format: DD/MM/YY ↵
Year: 2002
Month: 1: January
2: February
3: March
4: April
5: May
6: June
7: July
8: August

Enter=OK

Pulse ↵ o ↵ hasta
resaltar octubre y
pulse **ENTER**

CLOCK

Time Format: 12 Hour ↵
Hour: 1
Minute: 30
AM/PM: PM ↵
Date Format: DD/MM/YY ↵
Year: 2002
Month: **October** ↵

Enter=OK ESC=CANCEL

↵ **1 9**

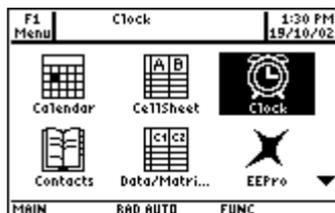
CLOCK

Hour: 1
Minute: 30
AM/PM: PM ↵
Date Format: DD/MM/YY ↵
Year: 2002
Month: October ↵
Day: 19

Enter=OK ESC=CANCEL

Pulse**Resultado****ENTER** **ENTER**

Fecha y hora revisadas

**Desactivación del reloj**

Desde el escritorio de Apps, abra el cuadro de diálogo **CLOCK (RELOJ)** y seleccione OFF en el campo Clock (Reloj).

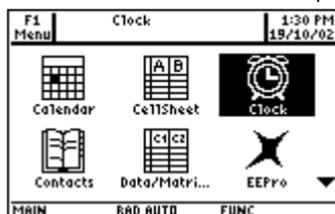
Ejemplo: Desactivar el reloj.

Pulse**Resultado**

Use las teclas del cursor para resaltar



Reloj activado

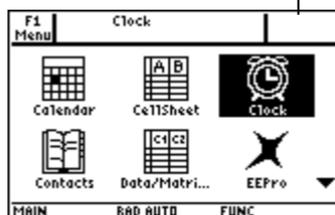
**ENTER**

Baje al campo Clock.


ENTER


[ENTER]

Reloj desactivado



Para activar el reloj, repita el procedimiento, pero seleccione ON en el campo Clock (Reloj). No olvide reajustar la fecha y la hora.

Uso de los menús

Para seleccionar la mayoría de los menús de la TI-89 Titanium o la Voyage™ 200, basta con pulsar las teclas de función correspondientes a las barras de herramientas que hay en la parte superior de la pantalla Home (Principal) de la calculadora y de casi todas las pantallas de App. Los demás menús se seleccionan mediante órdenes del teclado.

Menús de barras de herramientas

El punto de partida para realizar operaciones matemáticas con la TI-89 Titanium o la Voyage™ 200, la pantalla Home (Principal) de la calculadora, contiene menús de barras de herramientas donde puede seleccionar diversas operaciones matemáticas (consulte el ejemplo de la página siguiente).

Los menús de barras de herramientas también aparecen en la parte superior de casi todas las pantallas de App. Contienen las funciones más habituales de la App activa.

Otros menús

Use órdenes del teclado para seleccionar los menús siguientes, que contienen las mismas opciones, con independencia de la pantalla que se vea o de la App activa.

Pulse	Para acceder al menú
--------------	-----------------------------

[2nd] **[CHAR]**

CHAR (CARACTERES). Muestra caracteres no disponibles en el teclado, organizados por categorías: griegos, matemáticos, signos de puntuación, especiales e internacionales).

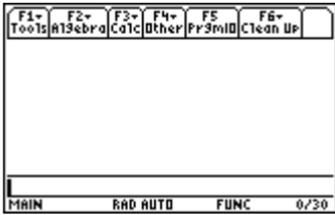
Pulse	Para acceder al menú
$\boxed{2nd}$ \boxed{MATH}	MATH (MATEMÁTICAS). Muestra las operaciones matemáticas por categorías.
\boxed{APPS}	APPLICATIONS (APLICACIONES). Muestra las Apps instaladas. Este menú sólo está disponible si está desactivado el escritorio de Apps, desde el que se accede normalmente a las Apps.
$\boxed{\blacklozenge}$ \boxed{APPS}	FLASH APPLICATIONS (APLICACIONES FLASH). Muestra las Apps Flash instaladas. Este menú sólo está disponible si está desactivado el escritorio de Apps, desde el que se accede normalmente a las Apps Flash.

Selección de opciones de menú

- Pulse el número o la letra que hay a la izquierda de la opción que desea seleccionar.
– o bien –
- Pulse \leftarrow o \rightarrow para seleccionar la opción y después pulse \boxed{ENTER} .

Nota: Si está seleccionada la primera opción del menú, pulse \leftarrow para seleccionar la última. Si está seleccionada la última opción del menú, pulse \rightarrow para seleccionar la primera.

Ejemplo: Seleccionar **factor**(en el menú Algebra (Álgebra) de la pantalla Home (Principal) de la calculadora.

Pulse	Resultado
Pulse:	
TI-89 Titanium: \boxed{HOME}	
Voyage™ 200: $\boxed{\blacklozenge}$ $\boxed{CALC HOME}$	
– o –	
Desde el escritorio de Apps, use las teclas del cursor para resaltar	
	
y pulse \boxed{ENTER}	

Pulse**Resultado**

F2

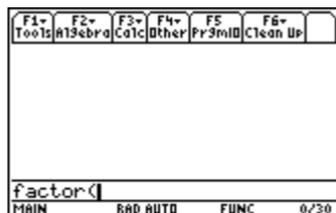


▼ indica que el menú Algebra (Álgebra) se abrirá cuando pulse ".

2

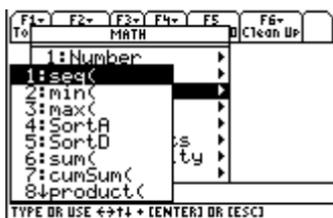
- 0 -

⇩ ENTER



Selección de opciones de submenú

Un símbolo de flecha pequeño (▶) a la derecha de una opción de menú indica que, al seleccionarla, se abre un submenú.

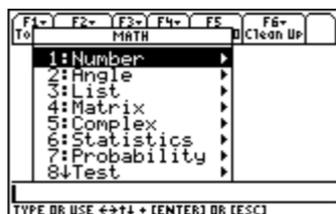


↓ apunta a otras opciones.

Ejemplo: Seleccionar **ord()** en el menú MATH (MATEMÁTICAS) de la pantalla Home (Principal) de la calculadora.

Pulse**Resultado**

2nd [MATH]



Pulse	Resultado
<p>D</p> <p>— 0 —</p> <p>← →</p>	
<p>B</p> <p>— 0 —</p> <p>→ [ENTER]</p>	

Uso de cuadros de diálogo

Los puntos suspensivos (...) al final de una opción de menú indican que, al seleccionarla, se abre un cuadro de diálogo. Seleccione la opción y pulse [ENTER].



Ejemplo: Abrir el cuadro de diálogo SAVE COPY AS (GUARDAR COMO) desde el Window Editor (Editor de ventanas).

Pulse	Resultado
<p>[APPS]</p> <p>Use las teclas del cursor para resaltar</p> <p>Window Edi...</p> <p>y pulse [ENTER]</p>	

Pulse**Resultado**

F1**2**

- 0 -

⇩ **ENTER**Pulse **(2)** para ver una lista de carpetas

Escriba el nombre de la variable.

Pulse **ENTER** dos veces para guardar y después cierre el cuadro de diálogo.

Nota: Al pulsar el método abreviado de teclado **⇩ S** también se abre el cuadro de diálogo **SAVE COPY AS (GUARDAR COMO)** en la mayoría de las Apps.

Cancelación de un menú

Para cancelar un menú sin seleccionar nada, pulse **ESC**.

Desplazamiento entre los menús de barras de herramientas

Para desplazarse entre los menús de barras de herramientas sin seleccionar opciones de menú:

- Pulse la tecla de función (**F1** a **F8**) de un menú de barras de herramientas.
- Pulse una tecla de función y después **⇨** o **⇩** para pasar de un menú de barras de herramientas al siguiente. Pulse **⇨** desde el último menú para trasladarse al primero. Pulse **⇩** para trasladarse del primer menú al último.

Nota: Si pulsa \odot cuando hay seleccionada una opción de menú que abre un submenú, aparece éste en lugar del siguiente menú de barras de herramientas. Vuelva a pulsar \odot para ir al siguiente menú.

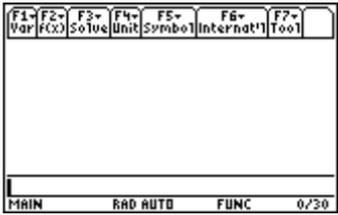
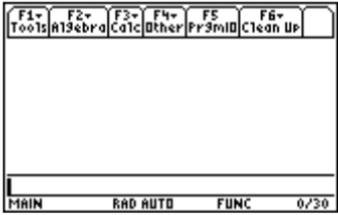
Para obtener más información sobre los menús, consulte el capítulo electrónico *Funcionamiento de la calculadora*.

Menú personalizado

El menú personalizado proporciona acceso rápido a las opciones de uso más frecuente. Use el menú personalizado predeterminado o cree uno propio con el Program Editor (Editor de programas). Puede incluir cualquier orden o carácter disponible en la TI-89 Titanium o la Voyage™ 200.

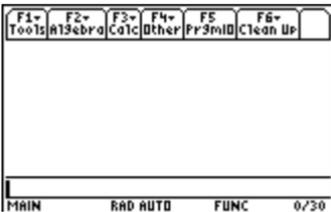
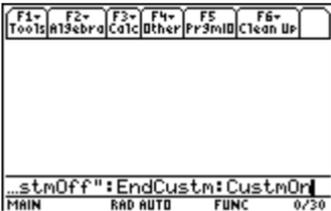
El menú personalizado sustituye al menú de barras de herramientas estándar en la pantalla Home (Principal) de la calculadora. La creación de menús personalizados se explica en el módulo en línea *Programación* de la TI-89 Titanium o la Voyage™ 200.

Ejemplo: Activar y desactivar el menú personalizado desde la pantalla Home (Principal) de la calculadora.

Pulse	Resultado
2nd [CUSTOM]	Menú personalizado predeterminado 
2nd [CUSTOM]	Menú de barras de herramientas 

Ejemplo: Restaurar el menú personalizado predeterminado.

Nota: Al restaurar el menú personalizado predeterminado se borra el menú personalizado anterior. Si ha creado éste con un programa, puede volver a ejecutar dicho programa para reutilizar el menú.

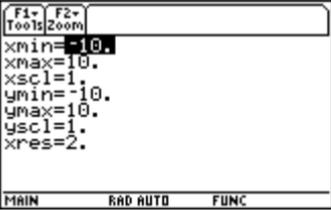
Pulse	Resultado
<p>2nd [CUSTOM] (para desactivar el menú personalizado y activar el menú de barras de herramientas estándar)</p>	
<p>TI-89 Titanium: 2nd [F6] Voyage™ 200: [F6]</p>	
<p>3 - 0 - ⏏ ⏏ [ENTER]</p>	
<p>[ENTER]</p>	

Cómo abrir Apps con el escritorio de Apps desactivado

Si desactiva el escritorio de Apps, use el menú APPLICATIONS (APLICACIONES) para abrir Apps. Para abrir el menú APPLICATIONS (APLICACIONES) con el escritorio de Apps desactivado, pulse [APPS].

Nota: Si pulsa [APPS] con el escritorio de Apps activado, aparecerá éste en lugar del menú APPLICATIONS (APLICACIONES).

Ejemplo: Con el escritorio de Apps desactivado, abrir el Window Editor (Editor de ventanas) desde el menú APPLICATIONS (APLICACIONES).

Pulse	Resultado
<p>APPS</p>	
<p>3</p> <p>- 0 -</p> <p>⏪ ⏩ ENTER</p>	

Para acceder a Apps no incluidas en el menú APPLICATIONS (APLICACIONES), seleccione **1:FlashApps (1:ApFlash)**.

Uso de la pantalla dividida

La TI-89 Titanium o la Voyage™ 200 permiten dividir la pantalla para ver dos Apps simultáneamente. Por ejemplo, puede mostrar al mismo tiempo las pantallas de Y= Editor y Graph (Gráf) para comparar la lista de funciones y ver cómo se representan gráficamente.

Definición del modo de pantalla dividida

En el cuadro de diálogo MODE (MODO) puede dividir la pantalla horizontal o verticalmente. La opción elegida permanece vigente hasta que vuelva a cambiarla.

1. Pulse **MODE** para abrir el cuadro de diálogo MODE (MODO).
2. Pulse **F2** para ver la definición del modo Split Screen (Pantalla dividida).
3. Pulse **⏪** para acceder al menú del modo Split Screen (Pantalla dividida).
4. Pulse **⏩** hasta resaltar TOP-BOTTOM (ARRIBA-ABAJO) o LEFT-RIGHT (IZQ-DCHA).
5. Pulse **ENTER**. La definición del modo Split Screen (Pantalla dividida) muestra la opción elegida.

6. Pulse **[ENTER]** de nuevo para guardar este cambio y mostrar la pantalla dividida.

Ejemplo: Definir el modo de pantalla dividida en TOP-BOTTOM (ARRIBA-ABAJO).

Pulse	Resultado
[MODE]	
[F2]	
[RIGHT] [LEFT]	
[ENTER]	
[ENTER]	

Definición de las Apps iniciales para pantalla dividida

Tras seleccionar el modo de pantalla dividida TOP-BOTTOM (ARRIBA-ABAJO) o LEFT-RIGHT (IZQ-DCHA), quedan disponibles otras definiciones de modo.

Modo de pantalla completa



Modo de pantalla dividida

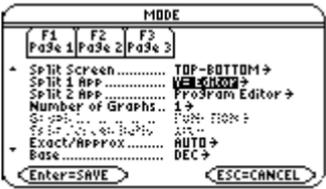
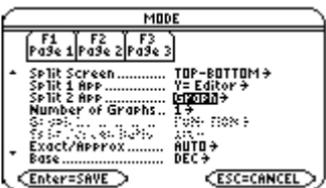
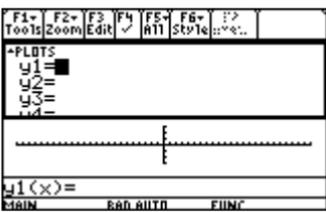


Modo	Descripción
Split (División) 2 App	Permite especificar la App que aparece en la parte inferior o derecha de la pantalla dividida. Funciona en combinación con Split (División) 1 App, que sirve para especificar la App que aparece en la parte superior o izquierda de la pantalla dividida.
Number of Graphs (Número de gráficos)	Sirve para configurar y mostrar dos gráficos independientes.
Split Screen Ratio (Proporción de pantalla dividida)	Cambia la proporción de las dos partes de una pantalla dividida.

Para definir la App inicial de cada parte de la pantalla dividida:

1. Seleccione la definición de modo Split (División) 1 App y pulse **Ⓢ** para acceder a un menú de las Apps disponibles.
2. Pulse **⊖** o **⊕** para resaltar la App y después pulse **[ENTER]**.
3. Repita los pasos 1 y 2 con la definición de modo Split (División) 2 App.

Ejemplo: Muestre $Y=$ Editor en la parte superior de la pantalla y la App Graph (Gráf) en la inferior.

Pulse	Resultado
<p>⏏ ⏏</p>	
<p>2</p>	
<p>⏏ ⏏</p>	
<p>4</p>	
<p>ENTER</p>	

Si define Split (División) 1 App y Split (División) 2 App en la misma App que no dé lugar a una representación gráfica o si el Number of Graphs (Número de gráficos) está ajustado en 1, la TI-89 Titanium o la Voyage™ 200 sale del modo de pantalla dividida y presenta la App en el modo de pantalla completa.

Selección de la App activa

En el modo de pantalla dividida no es posible tener dos Apps activas simultáneamente.

- Para alternar entre las Apps activas, pulse $\boxed{2nd} \boxed{[APP]}$.
- Para abrir una tercera App, pulse \boxed{APPS} y selecciónela. Esta App sustituirá a la App activa en la pantalla dividida.

Salida del modo de pantalla dividida

Existen varias formas de salir del modo de pantalla dividida:

- Pulsar $\boxed{2nd} \boxed{[QUIT]}$ para cerrar la App activa y ver en pantalla completa la otra App abierta.
- Si el escritorio de Apps está desactivado, al pulsar $\boxed{2nd} \boxed{[QUIT]}$ se sustituye la App activa de la pantalla dividida por la pantalla Home (Principal) de la calculadora. En caso de volver a pulsar $\boxed{2nd} \boxed{[QUIT]}$ se desactiva el modo de pantalla dividida y aparece la pantalla Home (Principal) de la calculadora en el modo de pantalla completa.
- Seleccionar Split Screen (Pantalla dividida) en la página 2 del cuadro de diálogo MODE (MODO), definir el modo de pantalla dividida en FULL (COMPLETA) y pulsar \boxed{ENTER} .
- Pulsar $\boxed{2nd} \boxed{[QUIT]}$ dos veces para acceder al escritorio de Apps.

Gestión de Apps y versiones de sistema operativo (SO)

Las funciones de conectividad de la TI-89 Titanium o la Voyage™ 200 permiten descargar Apps de:

- El sitio Web E&PS (Educational & Productivity Solutions) de TI, en la dirección: education.ti.com/latest
- El CD-ROM incluido con la TI-89 Titanium o la Voyage™ 200.
- Cualquier calculadora gráfica compatible.

Añadir Apps a la TI-89 Titanium o la Voyage™ 200 es igual que cargar software en un ordenador. Sólo hace falta el software TI Connect™ y el cable USB que se suministra con la TI-89 Titanium o la Voyage™ 200.

Encontrará los requisitos del sistema e instrucciones para conectar calculadoras compatibles y descargar el software TI Connect, Apps y versiones del SO en el sitio Web E&PS de TI.

Antes de descargar Apps en la TI-89 Titanium o la Voyage™ 200, le rogamos que lea el acuerdo de licencia que encontrará en el CD-ROM y en el sitio Web de TI.

Como encontrar la versión y el número de identificación (ID) del SO

Si adquiere software en el sitio Web E&PS de TI o llama al número del soporte al cliente, se le pedirá información sobre su TI-89 Titanium o Voyage™ 200. Dicha información se halla en la pantalla ABOUT (ACERCA DE).

Para acceder a la pantalla ABOUT (ACERCA DE), pulse **F1 3:About** (Acerca de) desde el escritorio de Apps. La pantalla ABOUT (ACERCA DE) contiene la siguiente información sobre la TI-89 Titanium o la Voyage™ 200:



❶ Versión del SO

❷ Versión del hardware

❸ Número de identificación de la unidad (Unit ID). Es similar al número de serie y necesario para obtener los certificados para instalar las Apps que adquiera. Anote el número y guárdelo en un lugar seguro. Sirve para identificar la unidad en caso de pérdida o robo.

❹ Número de revisión certificada (Rev. cert.) de Apps

❺ Número de identificación del producto (Product ID). Es similar al número de modelo.

Tenga en cuenta que el contenido de su pantalla será distinto del que muestra la ilustración.

Eliminación de una aplicación

Al eliminar una aplicación se suprime de la TI-89 Titanium o la Voyage™ 200 dejando más espacio para otras aplicaciones. Antes de eliminar una aplicación conviene guardarla en un ordenador para poder volver a instalarla en cualquier momento posterior.

1. Salga de la aplicación.
2. Pulse **[2nd]** **[VAR-LINK]** para acceder a la pantalla VAR-LINK (All) (CONEXIÓN DE VARIABLES (Todo)).
3. Pulse **[2nd]** **[F7]** (TI-89 Titanium) o **[F7]** para ver la lista de aplicaciones instaladas.
4. Para seleccionar la aplicación que desea eliminar, pulse **[F4]**. (Vuelva a pulsar **[F4]** para deseleccionarla).
5. Pulse **[F1]** **1:Delete (Borrar)**. Aparece un cuadro de diálogo de VAR-LINK (CONEXIÓN DE VARIABLES) para que confirme la eliminación.
6. Pulse **[ENTER]** para borrar la aplicación.

Nota: Sólo es posible eliminar Apps Flash.

Conexión de la TI-89 Titanium o la Voyage™ 200 con otros dispositivos

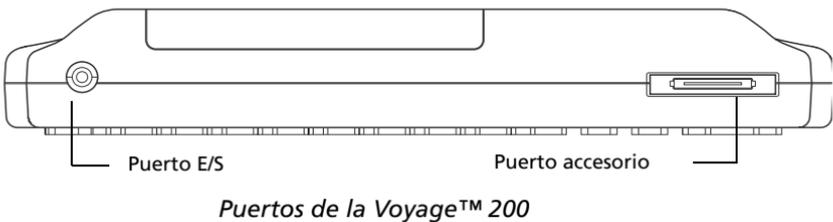
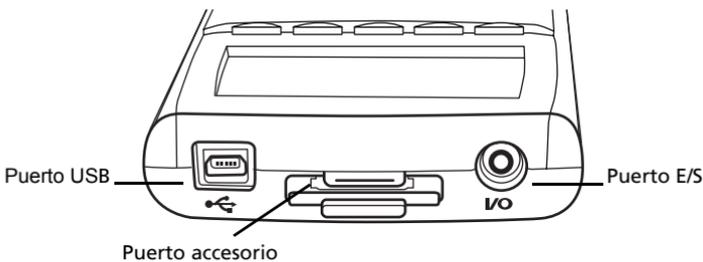
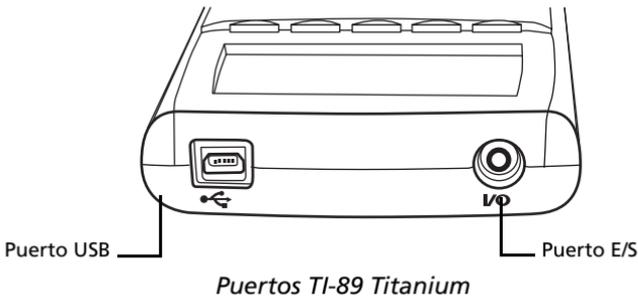
La TI-89 Titanium incluye un mini puerto USB. Tanto la TI-89 Titanium como la Voyage™ 200 incluyen un puerto E/S estándar. Los puertos se pueden utilizar para conectar entre sí dos calculadoras gráficas compatibles, o bien una calculadora gráfica con un ordenador o con un periférico.

El modelo para profesor de todas las calculadoras TI-89 Titanium y de todos los dispositivos Voyage™ 200 incluye, además, un puerto accesorio. El puerto se utiliza para la salida visual de datos, de forma que todos los alumnos puedan ver la pantalla del dispositivo en cuestión en un vídeo o un proyector.

Para conectar la calculadora a un ordenador – Conecte la TI-89 Titanium mediante el puerto USB y el USB cable incluido, o bien conecte la calculadora Voyage™ 200 al puerto E/S y el cable USB de conectividad de TI.

Para conectar dos calculadoras – Utilice el Cable USB de unidad-a-unidad o el Cable estándar de unidad-a-unidad para conectar la TI-89 Titanium o Voyage™ 200 con un dispositivo gráfico compatible, por ejemplo, una TI-89 o TI-92 Plus o los sistemas CBL 2™ y CBR™.

Para mostrar la pantalla del dispositivo a la clase – Utilice el puerto accesorio para conectar el adaptador de vídeo TI-Presenter™ a la Voyage™ 200. El adaptador de vídeo TI-Presenter proporciona una interfaz de vídeo entre la Voyage™ 200 y dispositivos de reproducción y grabación de vídeo. O bien utilice el puerto accesorio para conectar el panel de proyección TI ViewScreen™ al dispositivo. El panel de proyección TI ViewScreen se amplía y proyecta la pantalla para que toda la clase pueda verla. Para obtener más información sobre el adaptador de vídeo TI-Presenter y el panel de proyección TI ViewScreen, consulte el sitio Web de TI E&PS, en la dirección education.ti.com/latest.

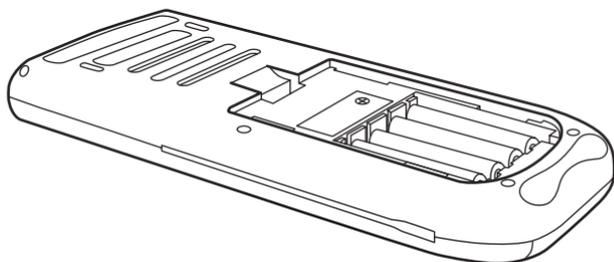


Pilas

La TI-89 Titanium utiliza cuatro pilas alcalinas AAA y una pila de óxido de plata (SR445W o 303) de reserva. La Voyage™ 200 utiliza cuatro pilas alcalinas AAA y una pila de litio (CR1616 o CR1620) de reserva. Las unidades vienen con las pilas de reserva ya instaladas; las pilas alcalinas AAA se suministran con el producto correspondiente.

Instalación de las pilas AAA

1. Retire la cubierta de las pilas, situada en la parte trasera de la unidad.
2. Extraiga de su embalaje las cuatro pilas. AAA suministradas con el producto, e introdúzcalas en el compartimento de las pilas siguiendo el diagrama de polaridad (+ y -) indicado en el mismo.



3. Vuelva a colocar la cubierta de las pilas en la unidad y encájela en su sitio.

Sustitución de las pilas AAA (alcalinas)

Cuando las pilas pierden potencia, la pantalla empieza a oscurecerse, sobre todo durante los cálculos. Si tiene que aumentar el contraste con frecuencia, sustituya las pilas alcalinas AAA.

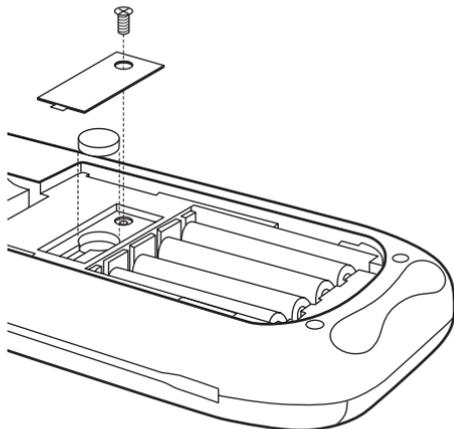
La línea de estado también informa sobre las pilas..

Indicador	Significado
	Las pilas están bajas.
	Sustituya las pilas tan pronto como sea posible.

Antes de sustituir las pilas, apague la TI-89 Titanium o la Voyage™ 200 pulsando **[2nd] [OFF]** para no perder la información almacenada en la memoria. No quite la pila de reserva y las pilas alcalinas AAA al mismo tiempo.

Sustitución de la pila de reserva (óxido de plata)

1. Para cambiar la pila de óxido de plata, retire la cubierta de la pila y extraiga el pequeño tornillo que sujeta la cubierta de la PILA DE RESERVA en su lugar.



2. Retire la pila gastada e instale una pila SR44SW o 303 nueva, con el polo positivo (+) en la parte superior. Vuelva a colocar el tornillo en su lugar.

Información importante sobre descarga de SO

Las pilas nuevas deben instalarse antes de empezar a descargar un SO.

En el modo de descarga de SO, la función APD™ no funciona. Si deja la calculadora en el modo de descarga durante mucho tiempo antes de empezar la descarga real, las pilas pueden agotarse, en cuyo caso deberá sustituirlas por otras nuevas antes de descargar.

También puede transferir el SO a otra TI-89 Titanium o Voyage™ 200 a través de un cable de unidad a unidad. Si la transferencia se interrumpe accidentalmente antes de terminar, tendrá que volver a instalar el SO desde un ordenador. También en este caso no olvide instalar pilas nuevas antes de descargar.

Si surge algún problema, póngase en contacto con Texas Instruments como se indica en el apartado Información de servicio y soporte.

Precauciones con las pilas

Tome estas precauciones al sustituir las pilas.

- No deje las pilas al alcance de los niños.

- No mezcle pilas nuevas y usadas. No mezcle marcas de pilas (ni tipos de una misma marca).
- No mezcle pilas recargables y no recargables.
- Instale las pilas siguiendo los diagramas de polaridad (+ y -).
- No coloque pilas no recargables en un cargador de pilas.
- Deseche las pilas usadas inmediatamente en la forma adecuada.
- No quemé ni desmonte las pilas.

Presentaciones en pantalla

Realización de operaciones

Esta sección incluye ejemplos que puede realizar desde la pantalla Home (Principal) de la calculadora para familiarizarse con las funciones de la TI-89 Titanium / Voyage™ 200. El área de historia de cada pantalla se ha limpiado pulsando **F1** y seleccionando **8:Clear Home** antes de efectuar cada ejemplo con el fin de ilustrar únicamente los resultados de las pulsaciones de teclas del ejemplo.

Presentación de operaciones

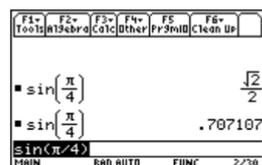
Pasos y pulsaciones

Calcule $\sin(\pi/4)$ y presente el resultado en los formatos simbólicos y numéricos. Para limpiar el área de historia de resultados anteriores, pulse **F1** y seleccione **8:Clear Home**.

 **2nd** **[SIN]** **2nd** **[π]** **÷** **4** **)** **ENTER** **♦** **[≈]**

 **[SIN]** **2nd** **[π]** **÷** **4** **)** **ENTER** **♦** **[≈]**

Visualización



Obtención del factorial de un número

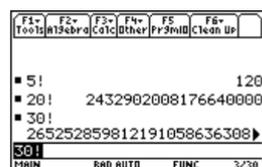
Pasos y pulsaciones

Calcule el factorial de varios números para ver cómo la TI-89 Titanium / Voyage™ 200 maneja números enteros muy grandes. Para obtener el operador factorial (!), pulse **2nd** **[MATH]**, seleccione **7:Probability**, y después seleccione **1:!**.

 **5** **2nd** **[MATH]** **7** **1** **ENTER** **20** **2nd** **[MATH]** **7** **1** **ENTER**
30 **2nd** **[MATH]** **7** **1** **ENTER**

 **5** **2nd** **[!]** **ENTER** **20** **2nd** **[!]** **ENTER** **30** **2nd** **[!]**
ENTER

Visualización



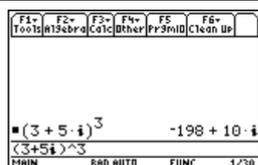
Desarrollo de números complejos

Pasos y pulsaciones

Calcule $(3+5i)^3$ para ver cómo la TI-89 Titanium / Voyage™ 200 realiza las operaciones en que intervienen números complejos.

Pulse $\boxed{(\ } \boxed{3} \boxed{+} \boxed{5} \boxed{2nd} \boxed{[i]} \boxed{)} \boxed{^} \boxed{3} \boxed{ENTER}$

Visualización



Obtención de factores primos

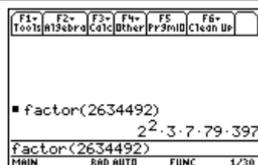
Pasos y pulsaciones

Calcule los factores del número entero 2634492. Puede introducir "factor" en la línea de entrada escribiendo **FACTOR** con el teclado, o pulsando $\boxed{F2}$ y seleccionando **2:factor(**.

Pulse $\boxed{F2} \boxed{2} \boxed{2634492} \boxed{)} \boxed{ENTER}$

(Opcional) Introduzca otros números cualesquiera.

Visualización



Obtención de raíces

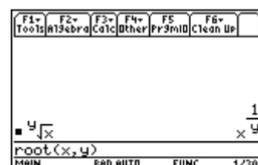
Pasos y pulsaciones

Halle la raíz de la expresión (x,y) . Puede introducir "root" en la línea de entrada escribiendo **ROOT** con el teclado, o pulsando $\boxed{\blacklozenge}$ 9.

Este ejemplo muestra el uso de la función raíz y cómo aparece la expresión en el modo "Pretty Print" en el área de historia.

Pulse $\boxed{\blacklozenge} \boxed{9} \boxed{X} \boxed{,} \boxed{Y} \boxed{)} \boxed{ENTER}$

Visualización



Expansión de expresiones

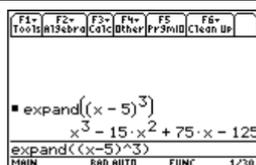
Pasos y pulsaciones

Expanda la expresión $(x-5)^3$. Puede introducir "expand" en la línea de entrada escribiendo "expand" con el teclado, o pulsando $\boxed{F2}$ y seleccionando **3:expand(**.

Pulse $\boxed{F2}$ 3 $\boxed{[X]}$ 5 $\boxed{[]}$ ^ 3 $\boxed{[]}$ \boxed{ENTER}

(Opcional) Introduzca otras expresiones cualesquiera.

Visualización



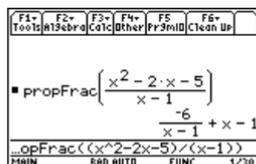
Reducción de expresiones

Pasos y pulsaciones

Reduzca la expresión $(x^2-2x-5)/(x-1)$ a su forma más simple. Puede introducir "propFrac" en la línea de entrada escribiendo **PROPFrac** con el teclado, o pulsando $\boxed{F2}$ y seleccionando **7:propFrac(**.

Pulse $\boxed{F2}$ 7 $\boxed{[X]}$ ^ 2 $\boxed{[-]}$ 2 $\boxed{[X]}$ $\boxed{[-]}$ 5 $\boxed{[]}$ $\boxed{[]}$ ÷ $\boxed{[X]}$ $\boxed{[-]}$ 1 $\boxed{[]}$ \boxed{ENTER}

Visualización



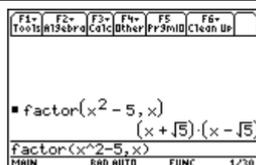
Factorización de polinomios

Pasos y pulsaciones

Descomponga en factores el polinomio (x^2-5) respecto de x. Puede introducir "factor" en la línea de entrada escribiendo **FACTOR** con el teclado, o pulsando $\boxed{F2}$ y seleccionando **2:factor(**.

Pulse $\boxed{F2}$ 2 $\boxed{[X]}$ ^ 2 $\boxed{[-]}$ 5 $\boxed{[]}$, $\boxed{[X]}$ $\boxed{[]}$ \boxed{ENTER}

Visualización



Resolución de ecuaciones

Pasos y pulsaciones

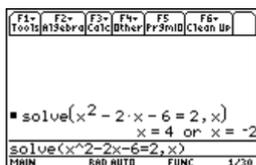
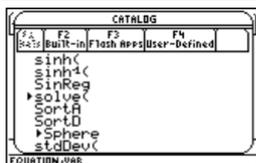
Resuelva la ecuación $x^2 - 2x - 6 = 2$ respecto de x .

Puede introducir "solve(" en la línea de entrada seleccionando "solve(" en el menú Catalog, escribiendo **SOLVE**(con el teclado, o pulsando **F2** y seleccionando **1:solve**(.

En la línea de estado aparece la sintaxis requerida por el elemento seleccionado en el menú **Catalog**.

Pulse **F2** 1 X **^** 2 **-** 2 X **-** 6 **=** 2 **,** X **)** **ENTER**

Visualización



Resolución de ecuaciones en un cierto dominio

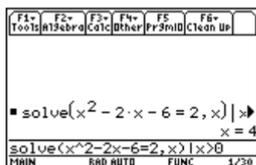
Pasos y pulsaciones

Resuelva la ecuación $x^2 - 2x - 6 = 2$ respecto de x , para x mayor que cero. El operador "with" (|>) proporciona restricción del dominio.

F2 1 X **^** 2 **-** 2 X **-** 6 **=** 2 **,** X **)** |> 1 X **2nd** [**>**] 0 **ENTER**

F2 1 X **^** 2 **-** 2 X **-** 6 **=** 2 **,** X **)** **2nd** [**1**] X **2nd** [**>**] 0 **ENTER**

Visualización



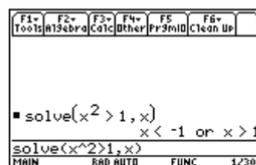
Resolución de desigualdades

Pasos y pulsaciones

Resuelva la desigualdad $(x^2 > 1, x)$ respecto de x .

Pulse **F2** 1 X **^** 2 **2nd** [**>**] 1 **)** **ENTER**

Visualización



Obtención de la derivada de una función

Pasos y pulsaciones

Halle la derivada de $(x-y)^3/(x+y)^2$ respecto de x .

Este ejemplo muestra el uso de la función derivada, y cómo aparece la función en el modo "Pretty Print" en el área de historia.

Pulse 2nd $[d]$ $($ X $-$ Y $)^3$ \div $($ X $+$ Y $)^2$ \wedge 2 $,$ X $)$ ENTER

Visualización

$$\frac{d}{dx} \left(\frac{(x-y)^3}{(x+y)^2} \right)$$
$$\frac{(x-y)^2 \cdot (x+5 \cdot y)}{(x+y)^3}$$

Obtención de derivadas implícitas

Pasos y pulsaciones

Calcule derivadas implícitas para ecuaciones con dos variables en la que una de ellas resulta implícitamente definida por los términos de la otra.

Este ejemplo muestra el uso de la función de cálculo de derivadas implícitas.

Pulse F3 D X \wedge 2 $+$ Y \wedge 2 $=$ 100 $,$ X $,$ Y $)$ ENTER

Visualización

$$\text{impDif}(x^2 + y^2 = 100, x, y)$$
$$-\frac{x}{y}$$

Obtención de la integral de una función

Pasos y pulsaciones

Halle la integral de $x \cdot \sin(x)$ respecto de x .

Este ejemplo muestra el uso de la función integral.

2nd $[f]$ X \times 2nd $[\text{SIN}]$ X $)$ $,$ X $)$ ENTER

2nd $[f]$ X \times $[\text{SIN}]$ X $)$ $,$ X $)$ ENTER

Visualización

$$\int (x \cdot \sin(x)) dx$$
$$\sin(x) - x \cdot \cos(x)$$

Solución de problemas con vectores

Pasos y pulsaciones

1. Introduzca una fila o una columna de vectores.

2nd [C] (-) 6 , 0 , 0 2nd [1] STO alpha d
 ENTER 2nd [C] 4 , 0 , 2 2nd [1] STO alpha a
 ENTER 2nd [C] (-) 1 , 2 , 1 2nd [1] STO alpha b
 ENTER 2nd [C] 7 , 6 , 5 2nd [1] STO alpha c
 ENTER

2nd [C] (-) 6 , 0 , 0 2nd [1] STO d
 ENTER 2nd [C] 4 , 0 , 2 2nd [1] STO a ENTER
 2nd [C] (-) 1 , 2 , 1 2nd [1] STO b ENTER
 2nd [C] 7 , 6 , 5 2nd [1] STO c ENTER

Visualización

F1- Tools	F2- R13Ebro	F3- ColC	F4- Other	F5- Pr3mD	F6- Clean Up
■ [-6 0 0] → d		[-6 0 0]			
■ [4 0 2] → a		[4 0 2]			
■ [-1 2 1] → b		[-1 2 1]			
■ [7 6 5] → c		[7 6 5]			
[[-6, 0, 5]] → c					
Main		Grid AUTO		Func 4/30	

2. Resuelva $(x \cdot a + y \cdot b + z \cdot c = d \{x, y, z\})$

F2 1 X alpha a + y X alpha b + z X
 alpha c = alpha d , 2nd [1] X , Y , Z 2nd [1]
) ENTER

F2 1 X X a + y X b + z X c = d ,
 2nd [1] X , Y , Z 2nd [1]) ENTER

F1- Tools	F2- R13Ebro	F3- ColC	F4- Other	F5- Pr3mD	F6- Clean Up
■ [-6 0 0] → d		[-6 0 0]			
■ [4 0 2] → a		[4 0 2]			
■ [-1 2 1] → b		[-1 2 1]			
■ [7 6 5] → c		[7 6 5]			
■ solve(x·a+y·b+z·c=d, x, y, z)					
■ solve(x·a+y·b+z·c=d, x, y, z)					
Main		Grid AUTO		Func 10/30	

Obtención de logaritmos de cualquier base

Pasos y pulsaciones

Halle el logaritmo (x, b) . Puede introducir "log" en la línea de entrada escribiendo **LOG** con el teclado, o pulsando 7.

7 X , alpha b) ENTER

7 X , b) ENTER

Visualización

F1- Tools	F2- R13Ebro	F3- ColC	F4- Other	F5- Pr3mD	F6- Clean Up
■ log _b (x)		log _b (x)			
log(x, b)					
Main		Grid AUTO		Func 1/30	

Conversión de medidas de ángulos

Pasos y pulsaciones

Visualización

1. Presente el cuadro de diálogo **MODE**. Para el modo **Angle** seleccione **DEGREE**. Convierta 345 a grados centesimales.

Puede introducir "►**Grad**" en la línea de entrada seleccionando "►**Grad**" en los menús Catalog (Catálogo) o Math (Matemáticas) pulsando [2nd] [MATH] y seleccionando **2:angle**, **A:►Grad**.

 [MODE] [◀] [◀] [◀] [▶] 2 [ENTER] 345 [2nd] [°] [2nd] [MATH] 2 [alpha] A [ENTER]

 [MODE] [◀] [◀] [◀] [▶] 2 [ENTER] 345 [2nd] [°] [2nd] [MATH] 2 A [ENTER]

F1+ Tools	F2+ 13eBrj	F3+ CclC	F4+ Other	F5 Pr3vID	F6+ Clean Up
■ (345 °)►Grad $\left(\frac{1150}{3}\right)^{\circ}$					
(345°)►Grad					
MIN		DEGRAD		FUNC 1/30	

2. Convierta 345 grados a radianes.

Puede introducir "►**Rad**" en la línea de entrada seleccionando "►**Rad**" en los menús Catalog (Catálogo) o Math (Matemáticas) pulsando [2nd] [MATH] y seleccionando **2:angle**, **B:►Rad**.

 [MODE] [◀] [◀] [◀] [▶] 2 [ENTER] 345 [2nd] [°] [2nd] [MATH] 2 [alpha] B [ENTER]

 [MODE] [◀] [◀] [◀] [▶] 2 [ENTER] 345 [2nd] [°] [2nd] [MATH] 2 B [ENTER]

F1+ Tools	F2+ 13eBrj	F3+ CclC	F4+ Other	F5 Pr3vID	F6+ Clean Up
■ (345 °)►Rad $\left(\frac{23 \cdot \pi}{12}\right)^{\circ}$					
345°►Rad					
MIN		DEGRAD		FUNC 1/30	

Nota: También puede utilizar las teclas $^{\circ}$, $^{\circ}$ o $^{\circ}$ para anular temporalmente el valor de configuración del modo de ángulo.

Cálculo simbólico

Resuelva el sistema de ecuaciones $2x - 3y = 4$ and $-x + 7y = -12$. Resuelva la primera ecuación para que x se exprese en función de y . Sustituya la expresión de x en la segunda ecuación, y calcule el valor de y . Después, vuelva a sustituir el valor de y en la primera ecuación para hallar el valor de x .

Pasos y pulsaciones

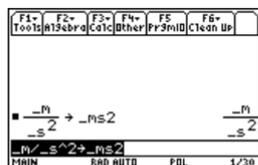
Visualización

1. Presente la pantalla Home y limpie la línea de entrada. Resuelva la ecuación $2x - 3y = 4$ para x .

F2 1 selecciona **solve(** en el menú Algebra. También puede escribir **solve(** directamente del teclado o seleccionarlo en **Catalog**.

 **HOME** **CLEAR** **CLEAR** **F2** 1 2 X **□** 3 Y **□** 4
□ X **□** **ENTER**

 **♦** **[CALC HOME]** **CLEAR** **CLEAR** **F2** 1 2 X **□**
3 Y **□** 4 **□** X **□** **ENTER**



2. Empiece a resolver la ecuación $-x + 7y = -12$ para y , pero no pulse aún **ENTER**.

Pulse **F2** 1 **(-)** X **+** 7 Y **□** **(-)** 12 **□** Y **□**

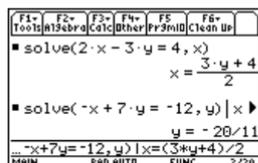
3. Utilice el operador "with" para sustituir la expresión de x que calculó con la primera ecuación. Con ello se obtiene el valor de y .

El operador "with" aparece como | en la pantalla.

Utilice la función de pegado automático para resaltar la última respuesta en el área de historia y pegarla en la línea de entrada.

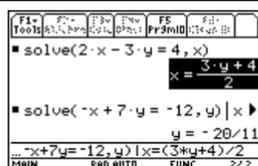
 **□** **←** **ENTER** **ENTER**

 **2nd** **[1]** **←** **ENTER** **ENTER**



4. Resalte la ecuación en x en el área de historia.

Pulse **←** **←** **←**



Pasos y pulsaciones

1. Pegue automáticamente la expresión resaltada en la línea de entrada. Después, sustituya el valor de y obtenido en la segunda ecuación.

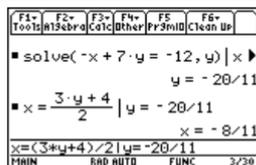
 **ENTER** **|** **→** **ENTER** **ENTER**

 **ENTER** **2nd** **|** **→** **ENTER** **ENTER**

La solución es:

$x = -8/11$ and $y = -20/11$

Visualización



Este ejemplo es una demostración del cálculo simbólico. Hay disponible una función para resolver sistemas de ecuaciones de forma directa.

Constantes y unidades de medida

Utilizando la ecuación $f = m \cdot a$, obtenga la fuerza para $m = 5$ kilogramos y $a = 20$ metros/segundo². ¿Cuál es la fuerza cuando $a = 9,8$ metros/segundo²? (Esta es la aceleración debida a la gravedad; es una constante llamada $_g$). Convierta el resultado de newtons a kilogramos fuerza.

Pasos y pulsaciones

1. Abra el cuadro de diálogo **MODE**, página 3. Para el modo **Unit System**, seleccione **SI** para el sistema métrico.

Los resultados se muestran según estas unidades predeterminadas.

Pulse **MODE** **F3** **↓** **1** **ENTER**

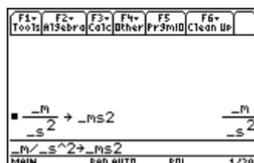
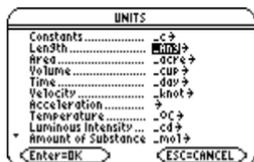
Visualización



2. Cree una unidad de aceleración para metros/segundo² llamada `_ms2`.

El cuadro de diálogo **UNITS** permite seleccionar unidades en una lista alfabética de categorías. Use $\boxed{2nd}$ \odot y $\boxed{2nd}$ \odot para desplazarse por las páginas de categorías (una cada vez).

Ahora, en lugar de volver a introducir `_m/_s^2` cada vez que sea preciso, puede utilizar `_ms2`. Si utiliza el cuadro de diálogo **UNITS** para seleccionar una unidad, el símbolo `_` se introduce de forma automática. Además, ahora puede usar el cuadro de diálogo **UNITS** para seleccionar `_ms2` en la categoría *Acceleration*.



$\boxed{2nd}$ \odot $\boxed{2nd}$ \odot M \boxed{ENTER} \div $\boxed{2nd}$ \odot \odot \odot \odot S \boxed{ENTER} \wedge 2 \boxed{STO} \blacklozenge $\boxed{[-]}$ $\boxed{2nd}$ $\boxed{[a-lock]}$ M S α 2 \boxed{ENTER}

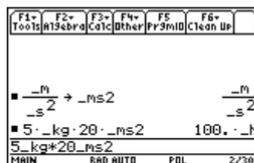
\blacklozenge $\boxed{[-]}$ $\boxed{2nd}$ \odot \odot \odot \odot S \boxed{ENTER} \wedge 2 \boxed{STO} $\boxed{2nd}$ $\boxed{[-]}$ M S 2 \boxed{ENTER}

3. Calcule la fuerza cuando
 $m = 5$ kilogramos (`_kg`) y
 $a = 20$ metros/segundo² (`_ms2`).

Si conoce la abreviatura de una unidad, puede escribirla desde el teclado.

$\boxed{5}$ \blacklozenge $\boxed{[-]}$ $\boxed{2nd}$ $\boxed{[a-lock]}$ K G α \times 2 0 \blacklozenge $\boxed{[-]}$ $\boxed{2nd}$ $\boxed{[a-lock]}$ M S α 2 \boxed{ENTER}

$\boxed{5}$ $\boxed{2nd}$ $\boxed{[-]}$ K G \times 20 $\boxed{2nd}$ $\boxed{[-]}$ M S 2 \boxed{ENTER}



Pasos y pulsaciones

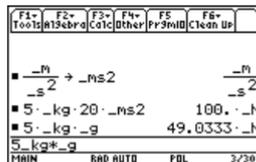
Visualización

4. Con el mismo valor m , calcule la fuerza para la aceleración debida a la gravedad (constante g).

Para g , puede usar la constante predefinida en el cuadro de diálogo **UNITS** o puede escribir g .

 5 \blacklozenge [] 2nd [a-lock] K G α \times 2nd [UNITS] \blacktriangleright α G ENTER ENTER

 5 2nd [] K G \times \blacklozenge [UNITS] \blacktriangleright G ENTER ENTER

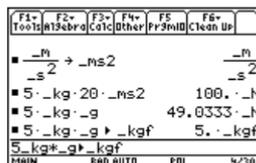


5. Convierta a kilogramos fuerza (kgf).

2nd \blacktriangleright muestra el operador de conversión \blacktriangleright .

 \blacktriangleright 2nd \blacktriangleright \blacklozenge [] 2nd [a-lock] K G F α ENTER

 \blacktriangleright 2nd \blacktriangleright 2nd [] K G F ENTER



Representación gráfica básica de funciones I

El ejemplo de esta sección demuestra algunas de las capacidades de representación gráfica de la TI-89 Titanium / Voyage™ 200. Enseña a representar gráficamente una función utilizando **Y= Editor**. Aprenderá a introducir una función, dibujar su gráfica, moverse a lo largo de ella, encontrar un mínimo y transferir las coordenadas del mismo a la pantalla de inicio.

Explore las capacidades gráficas de la TI-89 Titanium / Voyage™ 200 efectuando la representación gráfica de la función $y = (|x^2 - 3| - 10)/2$.

Pasos y pulsaciones

Visualización

1. Abra **Y= Editor**.

Pulse \blacklozenge [Y=]



Pasos y pulsaciones

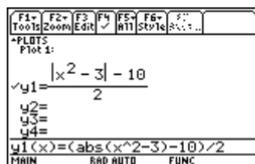
Visualización

2. Introduzca la función $(\text{abs}(x^2-3)-10)/2$.

La imagen muestra la "reproducción visual" de la función introducida en Y1=.



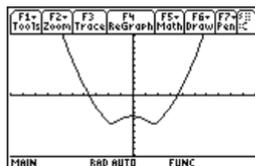




3. Presente la gráfica de la función.

Seleccione **6:ZoomStd** pulsando **6** o moviendo el cursor a **6:ZoomStd** y pulsando **[ENTER]**.

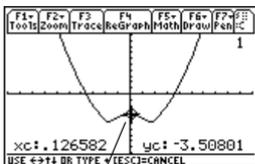
Pulse **[F2]** 6



4. Active **Trace**.

Aparecen el cursor traza y las coordenadas x e y.

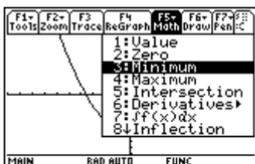
Pulse **[F3]**



cursor traza

5. Abra el menú **MATH** y seleccione **3:Minimum**.

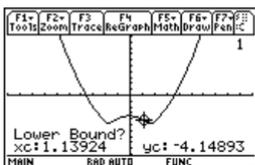
Pulse **[F5]** \leftarrow \leftarrow **[ENTER]**



6. Elija el extremo inferior.

Pulse \rightarrow (cursor derecho) para mover el cursor traza hasta que el extremo inferior para la x esté justo a la izquierda del mínimo y, posteriormente, pulse **[ENTER]**.

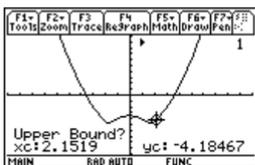
Pulse \rightarrow ... \rightarrow **[ENTER]**



7. Elija el extremo superior.

Pulse \rightarrow (cursor derecho) para mover el cursor traza hasta que el extremo superior para la x esté justo a la derecha del mínimo.

Pulse \rightarrow ... \rightarrow



Pasos y pulsaciones

- Obtenga el mínimo en la gráfica entre los extremos inferior y superior.

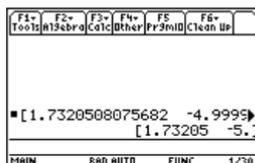
Pulse **ENTER**

Visualización



- Transfiera el resultado a la pantalla Home, y después muestre dicha pantalla.

HOME
H **[CALC HOME]**



Representación gráfica básica de funciones II

Represente una circunferencia de radio 5, centrada en el origen del sistema de coordenadas. Vea cómo aparece la circunferencia con la ventana de visualización estándar (**ZoomStd**). Después, utilice **ZoomSqr** para ajustar la ventana de visualización.

Pasos y pulsaciones

- Presente el recuadro de diálogo **MODE**. Para el modo **Graph**, seleccione **FUNCTION**.

Pulse **MODE** **1** **ENTER**

Visualización



- Vaya a la pantalla Home. Después almacene el radio, 5, en la variable **R**.

HOME **5** **STO** **alpha** **R** **ENTER**
[CALC HOME] **5** **STO** **R** **ENTER**

5 →

3. Presente y limpie Y= Editor. Defina

$y1(x) = \sqrt{r^2 - x^2}$, la mitad superior de una circunferencia.

En la representación de funciones, debe definir funciones separadas para las mitades superior e inferior de una circunferencia.

 \diamond [Y=] [F1] 8 [ENTER] [ENTER] [2nd] [√] [alpha] R
 \wedge 2 - X \wedge 2 [)] [ENTER]

 \diamond [Y=] [F1] 8 [ENTER] [ENTER] [2nd] [√] R \wedge 2
 - X \wedge 2 [)] [ENTER]

4. Defina $y2(x) = -\sqrt{r^2 - x^2}$, la función para la mitad inferior de la circunferencia.

La mitad inferior es la función opuesta de la mitad superior, por lo que puede definir $y2(x) = -y1(x)$.

Utilice el nombre completo de la función **y1(x)**, no sólo y1.

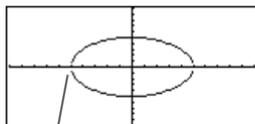
Pulse [ENTER] (-) Y 1 () X [)] [ENTER]



5. Seleccione la ventana de visualización **ZoomStd**, que representa las funciones automáticamente.

En la ventana de visualización estándar, los ejes x e y abarcan desde -10 hasta 10. Sin embargo, puesto que la longitud horizontal de la pantalla es más grande que la vertical, la circunferencia aparece como una elipse.

Pulse [F2] 6

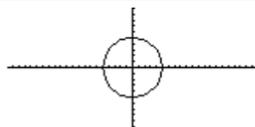


Observe la pequeña interrupción entre ambas mitades.

6. Seleccione **ZoomSqr**.

ZoomSqr incrementa el número de unidades en el eje x, para que las circunferencias y cuadrados se muestren en su proporción correcta.

Pulse [F2] 5



Nota: Hay un espacio entre las mitades superior e inferior de la circunferencia porque cada mitad es una función independiente. Los extremos analíticos de cada mitad son (-5,0) y (5,0). Dependiendo de la ventana de visualización, los extremos *representados* de cada mitad pueden variar ligeramente respecto a los extremos *analíticos*.

Representación gráfica básica de funciones III

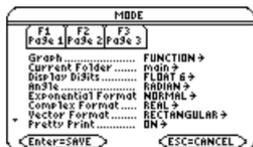
Utilice la función "Detect Discontinuities" para eliminar asíntotas y conexiones falsas en una discontinuidad de salto.

Pasos y pulsaciones

7. Presente el cuadro de diálogo **MODE**. Para el modo **Graph** seleccione **FUNCTION**. En modo **Angle** seleccione **RADIAN**.

Pulse $\boxed{\text{MODE}}$ \downarrow 1 \leftarrow \leftarrow \leftarrow \leftarrow \leftarrow \leftarrow 1 $\boxed{\text{ENTER}}$

Visualización



8. Abra Y= Editor e introduzca $y_1(x)=1/(x-1)$.

Pulse \blacklozenge $\boxed{Y=}$ 1 $\boxed{\div}$ $\boxed{(}$ X $\boxed{-}$ 1 $\boxed{)}$ $\boxed{\text{ENTER}}$

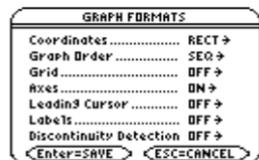


9. Presente el cuadro de diálogo **Graph Formats** y defina "Detect Discontinuities" en OFF

Nota: El segundo elemento del cuadro de diálogo **Graph Format** no aparece atenuado, lo que significa que se puede definir en consecutivo "Seq" o simultáneo "Simul".

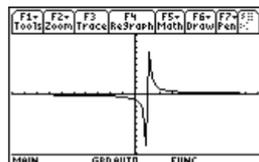
$\boxed{\text{GRAPH}}$ \blacklozenge $\boxed{\text{I}}$ \leftarrow \leftarrow \leftarrow \leftarrow \leftarrow \leftarrow \leftarrow \leftarrow 1 $\boxed{\text{ENTER}}$

$\boxed{\text{GRAPH}}$ \blacklozenge F \leftarrow \leftarrow \leftarrow \leftarrow \leftarrow \leftarrow \leftarrow \leftarrow 1 $\boxed{\text{ENTER}}$



10. Ejecute la orden **Graph** que muestra automáticamente la pantalla **Graph**. Observe las asíntotas "falsas" contenidas en la gráfica.

Pulse \blacklozenge $\boxed{\text{GRAPH}}$



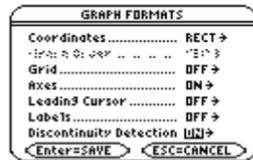
Pasos y pulsaciones

11. Presente el cuadro de diálogo **Graph Formats** y defina "Detect Discontinuities" en ON.
Nota: El segundo elemento del cuadro de diálogo **Graph Format** aparece atenuado lo que significa que el orden de la gráfica está definido en consecutivo "Seq".

  2 [ENTER]

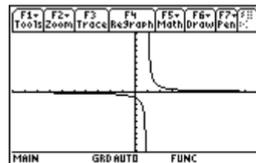
  2 [ENTER]

Visualización



12. Ejecute la orden **Graph** que muestra automáticamente la pantalla **Graph**. Observe que la gráfica no presenta asíntotas "falsas".
Nota: La velocidad de representación de la gráfica se puede ver afectada negativamente cuando "Detect Discontinuities" está definido en ON.

Pulse  [GRAPH]



Gráficas en paramétricas

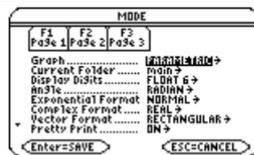
Represente las ecuaciones paramétricas que describen el recorrido de un balón lanzado con un ángulo (θ) de 60° y una velocidad inicial (v_0) de 15 metros/seg. Si la aceleración de la gravedad es $g = 9,8$ metros/seg² y se desprecia la resistencia del aire y de otras fuerzas de arrastre, ¿qué altura máxima alcanzará el balón y en qué instante caerá al suelo?

Pasos y pulsaciones

1. Presente el recuadro de diálogo **MODE**. Para el modo **Graph**, seleccione **PARAMETRIC**.

Pulse [MODE]  2 [ENTER]

Visualización



Pasos y pulsaciones

Visualización

2. Abra y limpie **Y= Editor**. A continuación, defina la componente horizontal
 $xt1(t) = v_0 t \cos \theta$.

$$xt1(t)=15t*\cos(60^\circ)$$

Introduzca valores para v_0 y θ .

\blacklozenge [Y=] [F1] 8 [ENTER] [ENTER] 15T \times [2nd] [COS] 60 [2nd] [$^\circ$] \square [ENTER]

\blacklozenge [Y=] [F1] 8 [ENTER] [ENTER] 15T \times [COS] 60 [2nd] [$^\circ$] \square [ENTER]

TI-89 Titanium: Escriba T \times [2nd] [COS], no T [2nd] [COS].

Voyage™ 200: Escriba T \times [COS], no T [COS].

Introduzca el símbolo $^\circ$ escribiendo [2nd] [$^\circ$] o [2nd] [MATH] 2 1. De esta forma se garantiza que los números se interpretarán como grados, independientemente del modo Angle.

3. Defina la componente vertical

$$yt1(t) = v_0 t \sin \theta - (g/2)t^2.$$

Introduzca valores para v_0 , θ , y g .

[ENTER] 15T \times [2nd] [SIN] 60 [2nd] [$^\circ$] \square - ([9.8 \div 2 \square T \wedge 2 [ENTER]

[ENTER] 15T \times [SIN] 60 [2nd] [$^\circ$] \square - ([9.8 \div 2 \square T \wedge 2 [ENTER]

```
F1- F2- F3- F4- F5- F6- F7-  
Tools Zoom Edit R11 St1918 x: t...  
-PLT5  
✓xt1=15*t*cos(60°)  
✓yt1=15*t*sin(60°)-9.8/2*t^2  
xt2=  
yt2=  
xt3=  
yt3=  
yt1(t)=15*t+sin(60°)-9.8/...  
MAIN RAD AUTO PAR
```

4. Presente **Window Editor**. Introduzca las variables de ventana apropiadas para este ejemplo.

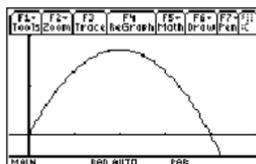
Puede pulsar \blacklozenge o [ENTER] para introducir un valor y pasar a la siguiente variable.

Pulse \blacklozenge [WINDOW] 0 \blacklozenge 3 \blacklozenge .02 \blacklozenge (-) 2 \blacklozenge 25 \blacklozenge 5 \blacklozenge (-) 2 \blacklozenge 10 \blacklozenge 5

```
F1- F2-  
Tools Zoom  
tmin=0  
tmax=3.  
tstep=.02  
xmin=-2.  
xmax=25.  
xsc1=5.  
ymin=-2.  
ymax=10.  
ysc1=5  
MAIN RAD AUTO PAR
```

5. Represente gráficamente las ecuaciones paramétricas para ver el modelo teórico del movimiento efectuado por el balón.

Pulse \blacklozenge [GRAPH]



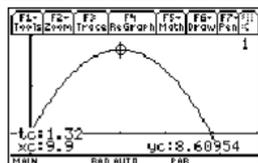
Pasos y pulsaciones

6. Seleccione **Trace**. A continuación, desplace el cursor por el recorrido para hallar:

- el valor y en la altura máxima.
- el valor t cuando el balón golpea el suelo.

Pulse **F3** \downarrow o \uparrow en caso necesario

Visualización



Gráficas en polares

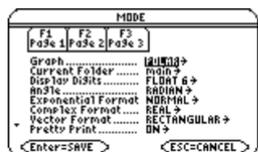
La representación de la gráfica de la ecuación polar $r_1(\theta) = A \sin B\theta$ es similar a una rosa. Represente gráficamente la rosa correspondiente a los valores $A=8$ y $B=2.5$, y, a continuación, realice un estudio de cómo sería su apariencia para otros valores de A y B .

Pasos y pulsaciones

1. Presente el recuadro de diálogo **MODE**. Para el modo **Graph**, seleccione **POLAR**. Para el modo **Graph**, seleccione **RADIAN**.

Pulse **MODE** \downarrow 3 \downarrow \downarrow \downarrow 1 **ENTER**

Visualización



2. Abra y limpie **Y= Editor**. A continuación, defina la ecuación polar $r_1(\theta) = A \sin B\theta$.

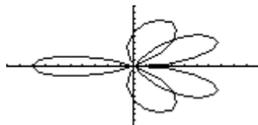
Introduzca 8 y 2.5 para A y B respectivamente.

Y= Editor \downarrow [Y=] **F1** 8 **ENTER** **ENTER** 8 **2nd** [SIN] 2.5 **ENTER**

Y= Editor \downarrow [Y=] **F1** 8 **ENTER** **ENTER** 8 **SIN** 2.5 θ **ENTER**



3. Seleccione la ventana de visualización **ZoomStd**, donde se representa la gráfica de la ecuación.



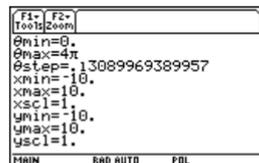
- En la gráfica sólo aparecen cinco pétalos de rosa.
 - En la ventana de visualización estándar, la variable de ventana $\theta_{\max} = 2\pi$. Los pétalos restantes tienen valores θ superiores a 2π .
- La rosa no tiene una apariencia simétrica.
 - Los ejes x e y oscilan entre -10 y 10 . Sin embargo, este rango es superior en el eje x dado que la longitud horizontal de la pantalla es mayor que la vertical.

Pulse **[F2]** 6

4. Presente **Window Editor** y cambie θ_{\max} a 4π .

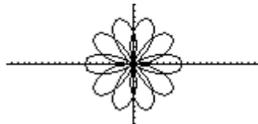
Se obtendrá el valor de 4π cuando se abandone **Window Editor**.

Pulse **[◀]** **[WINDOW]** **[↶]** 4 **[2nd]** **[π]**



5. Seleccione **ZoomSqr** para volver a representar la gráfica de la ecuación.

ZoomSqr incrementa el rango del eje x de forma que la gráfica se muestre en la proporción adecuada.



Pulse **[F2]** 5

Dé diferentes valores a A y a B y vuelva a representar la ecuación.

Representación gráfica de sucesiones

Un pequeño bosque tiene 4000 árboles. Cada año se corta el 20% de los árboles (dejando el 80% restante) y se plantan 1000 más. Utilizando una sucesión, calcule el número de árboles que hay en el bosque al final de cada año. ¿Se estabiliza esta cantidad en un cierto valor?

Inicio	Después de 1 año	Después de 2 años	Después de 3 años	...
4000	$.8 \times 4000 + 1000$	$.8 \times (.8 \times 4000 + 1000) + 1000$	$.8 \times (.8 \times (.8 \times 4000 + 1000) + 1000) + 1000$...

Pasos y pulsaciones

1. Presente el recuadro de diálogo **MODE**. Para el modo **Graph**, seleccione **SEQUENCE**.

Pulse $\boxed{\text{MODE}}$ \downarrow $\boxed{4}$ $\boxed{\text{ENTER}}$



2. Presente y limpie **Y= Editor**. Después defina la sucesión como $u1(n) = iPart(.8 * u1(n-1) + 1000)$.

Emplee **iPart** para obtener la parte entera del resultado. No se cortan fracciones de árboles.

Para acceder a **iPart**, puede utilizar $\boxed{2nd}$ $\boxed{\text{MATH}}$, simplemente escribirla o seleccionarla en **CATALOG**.

$\boxed{\text{2nd}}$ $\boxed{\text{MATH}}$ \downarrow $\boxed{\text{Y=}}$ $\boxed{F1}$ $\boxed{8}$ $\boxed{\text{ENTER}}$ $\boxed{\text{ENTER}}$ $\boxed{2nd}$ $\boxed{\text{MATH}}$ $\boxed{14.8}$

$\boxed{\alpha}$ $\boxed{U1}$ $\boxed{\downarrow}$ $\boxed{\alpha}$ \boxed{N} $\boxed{-}$ $\boxed{1}$ $\boxed{\downarrow}$ $\boxed{+}$ $\boxed{1000}$ $\boxed{\downarrow}$

$\boxed{\text{ENTER}}$

$\boxed{\text{2nd}}$ $\boxed{\text{MATH}}$ \downarrow $\boxed{\text{Y=}}$ $\boxed{F1}$ $\boxed{8}$ $\boxed{\text{ENTER}}$ $\boxed{\text{ENTER}}$ $\boxed{2nd}$ $\boxed{\text{MATH}}$ $\boxed{14.8}$

$\boxed{U1}$ $\boxed{\downarrow}$ \boxed{N} $\boxed{-}$ $\boxed{1}$ $\boxed{\downarrow}$ $\boxed{+}$ $\boxed{1000}$ $\boxed{\downarrow}$ $\boxed{\text{ENTER}}$



3. Defina $u1$ como el valor inicial del primer término.

Pulse $\boxed{\text{ENTER}}$ $\boxed{4000}$ $\boxed{\text{ENTER}}$

Pasos y pulsaciones

4. Presente **Window Editor**. Establezca las variables de ventana n y plot.

nmin=0 y **nmax=50** calculan el tamaño del bosque para un período de 50 años.

Pulse \blacklozenge [WINDOW] 0 \odot 50 \odot 1 \odot 1 \odot

Visualización

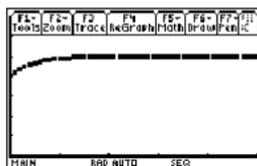
```
nmin=0.  
nmax=50.  
plotStart=1.  
plotStep=1.  
xmin=0.  
xmax=50.  
xsc1=10.  
ymin=0.  
ymax=6000.  
ytc1=1000.
```

5. Escriba los valores apropiados de este ejemplo para las variables x e y.

Pulse 0 \odot 50 \odot 10 \odot 0 \odot 6000 \odot 1000

6. Presente la pantalla Graph.

Pulse \blacklozenge [GRAPH]



7. Seleccione **Trace**. Mueva el cursor para desplazarse año por año. ¿Cuántos años (nc) tarda en estabilizarse el número de árboles (yc)?

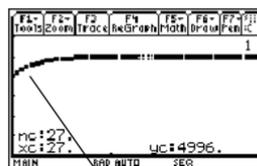
El desplazamiento empieza en nc=0.

nc es el número de años.

xc = nc ya que n se representa en el eje x.

yc = u1(n), número de árboles en el año n.

Pulse F3 \downarrow y \uparrow si es necesario



Por omisión, las sucesiones emplean el estilo de visualización Square.

Representación gráfica en 3D

Represente la ecuación 3D $z(x,y) = (x^3y - y^3x) / 390$. Anime la gráfica mediante el cursor para cambiar de forma interactiva los valores de la variable de ventana "eye" que controlan el ángulo de visualización. A continuación, vea la gráfica en distintos estilos de formato gráfico.

Pasos y pulsaciones

1. Presente el recuadro de diálogo **MODE**. En el modo **Graph**, seleccione **3D**.

Pulse [MODE] \downarrow 5 [ENTER]

Visualización



Pasos y pulsaciones

2. Presente y limpie **Y= Editor**. A continuación defina la ecuación 3D

$$z1(x,y) = (x^3y - y^3x) / 390.$$

Observe que se utiliza la multiplicación implícita.

Pulse \blacklozenge [Y=] [F1] 8 [ENTER] [ENTER] [] X \wedge 3 Y \square
Y \wedge 3 X \square \div 390 [ENTER]

Visualización

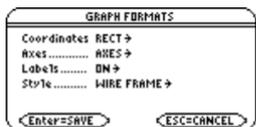


3. Cambie el formato gráfico para presentar y etiquetar los ejes. Además, establezca **Style = WIRE FRAME**.

Es posible animar cualquier estilo de formato gráfico, pero **WIRE FRAME** es el más rápido.

\square \blacklozenge [I] \leftarrow \rightarrow 2 \leftarrow \rightarrow 2 \leftarrow \rightarrow 1 [ENTER]

\square \blacklozenge F \leftarrow \rightarrow 2 \leftarrow \rightarrow 2 \leftarrow \rightarrow 1 [ENTER]



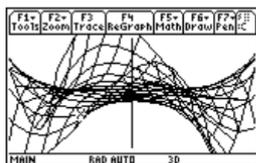
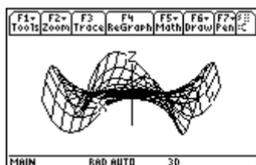
4. Seleccione el tipo de visualización **ZoomStd**, el cual representa la función de forma automática.

A medida que calcula la función (antes de presentarse su gráfica), muestra el "porcentaje calculado" en la esquina superior izquierda de la pantalla.

Pulse [F2] 6

Nota: Si ya está familiarizado con la representación gráfica en 3D, la gráfica puede presentarse en visualización ampliada. Al animar la gráfica, la pantalla vuelve a visualización normal de forma automática (excepto para la animación, ambos tipos de visualización permiten hacer las mismas cosas).

Pulse \square (pulse \square para cambiar entre visualización ampliada y normal)



Pasos y pulsaciones

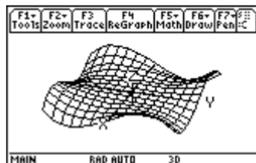
Visualización

5. Anime la gráfica disminuyendo el valor de la variable de ventana $\text{eye}\phi$.

⊖ o ⊕ pueden afectar a $\text{eye}\theta$ y $\text{eye}\psi$, pero en menor grado que $\text{eye}\phi$.

Para animar la gráfica de forma continua, mantenga pulsado el cursor durante aproximadamente 1 segundo y suéltelo a continuación. Para parar, pulse **[ENTER]**.

Pulse ⊖ ⊖ ⊖ ⊖ ⊖ ⊖ ⊖ ⊖



6. Restablezca la orientación original de la gráfica. A continuación, mueva el ángulo de visualización a lo largo de la "órbita de visualización" alrededor de la gráfica.

Pulse 0 (cero, no la letra O) ⊕ ⊕ ⊕



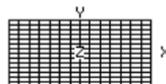
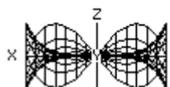
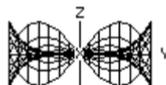
7. Vea la gráfica a lo largo del eje x, del eje y y del eje z.

Pulse X

Esta gráfica tiene idéntica forma tanto a lo largo del eje y como del eje x.

Pulse Y

Pulse Z



8. Vuelva a la orientación inicial.

Pulse 0 (cero)

Pasos y pulsaciones

9. Presente la gráfica en distintos estilos de formato gráfico.

 **I** (Pulse **I** para cambiar de un estilo al siguiente)

 **F** (Pulse **F** para cambiar de un estilo al siguiente)

Visualización



HIDDEN SURFACE



CONTOUR LEVELS
(calcular contornos puede requerir más tiempo)



WIRE AND CONTOUR



WIRE FRAME

Nota: También puede presentar la gráfica como una representación implícita mediante el cuadro de diálogo **GRAPH FORMATS** (**F1** 9 o TI-89 Titanium: **♦** **I**); Voyage™ 200: **♦** **F**). Si pulsa TI-89 Titanium: **I**; Voyage™ 200: **F** para conmutar entre estilos, la representación implícita no se presenta.

Representación gráfica de ecuaciones diferenciales

Represente gráficamente la solución de la ecuación diferencial logística de primer orden $y' = .001y*(100-y)$. Empiece dibujando solamente el campo de pendiente. A continuación, introduzca condiciones iniciales en **Y= Editor** y de forma interactiva desde la pantalla Graph.

Pasos y pulsaciones

1. Presente el cuadro de diálogo **MODE**. En el modo **Graph** seleccione **DIFF EQUATIONS**.

Pulse **MODE** **◂** **6** **ENTER**

Visualización



Pasos y pulsaciones

Visualización

2. Presente y limpie **Y= Editor**, y defina la ecuación diferencial de primer orden:

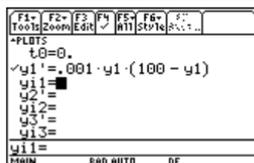
$$y1'(t)=.001y1*(100-y1)$$

Pulse \square para introducir el símbolo *. No utilice la multiplicación implícita entre la variable y el paréntesis. Si lo hiciera, se considerará como una llamada de función.

Deje la condición inicial **y1** en blanco.

Nota: Con **y1'** seleccionado, la calculadora representa gráficamente la curva solución **y1**, no la derivada **y1'**.

Pulse \square [Y=] [F1] 8 [ENTER] [ENTER] .001 Y1 \square () 100 \square Y1 \square [ENTER]

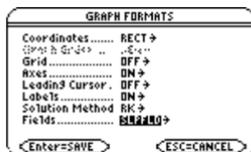


3. Presente el cuadro de diálogo **GRAPH FORMATS** y establezca **Axes = ON**, **Labels = ON**, **Solution Method = RK** y **Fields = SLPFLD**.

Importante: Para representar gráficamente una ecuación diferencial, Fields debe estar establecido en **SLPFLD** o **FLDOFF**. Si **Fields=DIRFLD**, aparecerá un error al realizar la representación gráfica.

\square \square | \square \square \square 2 \square \square 2 \square 1 \square 1
[ENTER]

\square F \square \square \square 2 \square \square 2 \square 1 \square 1
[ENTER]



4. Presente **Window Editor** y ajuste las variables de ventana como se indica a la derecha.

Pulse \square [WINDOW] 0 \square 10 \square .1 \square 0 \square (-) 10
 \square 110 \square 10 \square (-) 10 \square 120 \square 10 \square 0 \square
.001 \square 20

```
t0=0.
tmax=10.
tstep=.1
tplot=0.
xmin=-10.
xmax=110.
xsc1=10.
ymin=-10.
ymax=120.
ysc1=10.
ncurves=0.
diftol=.001
fldres=20.
```

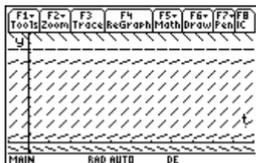
Pasos y pulsaciones

Visualización

5. Presente la pantalla Graph.

Dado que no se ha especificado ninguna condición inicial, sólo aparece el campo de pendiente (como especifica **Fields=SLPFLD** en el cuadro de diálogo **GRAPH FORMATS**).

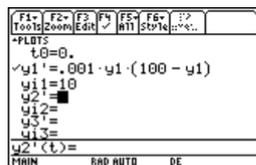
Pulse \blacklozenge [GRAPH]



6. Vuelva a **Y= Editor** e introduzca una condición inicial:

yi1=10

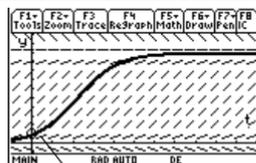
Pulse \blacklozenge [Y=] [ENTER] 10 [ENTER]



7. Vuelva a la pantalla Graph.

Las condiciones iniciales que se introducen en **Y= Editor** siempre se producen en t_0 . La gráfica comienza en la condición inicial, se traza hacia la derecha y, a continuación, hacia la izquierda.

Pulse \blacklozenge [GRAPH]

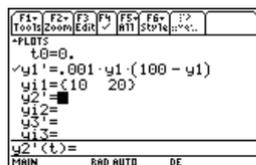


La condición inicial se indica con un círculo.

8. Vuelva a **Y= Editor** y cambie **yi1** para introducir dos condiciones iniciales en forma de lista:

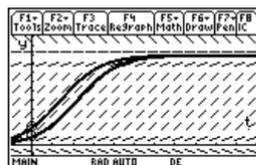
yi1={10,20}

Pulse \blacklozenge [Y=] \leftarrow [ENTER] 2nd [t] 10 , 20 2nd [)] [ENTER]



9. Vuelva a la pantalla Graph.

Pulse \blacklozenge [GRAPH]



Pasos y pulsaciones

10. Para seleccionar una condición inicial de forma interactiva, pulse:

 **2nd** **[F8]**

 **[F8]**

Cuando se le solicite, introduzca $t=40$ e $y_1=45$.

Al seleccionar una condición inicial de forma interactiva, puede especificar un valor para t distinto del t_0 introducido en **Y= Editor** o en **Window Editor**.

En lugar de introducir **t** e **y1** después de pulsar

 **2nd** **[F8]**

 **[F8]**,

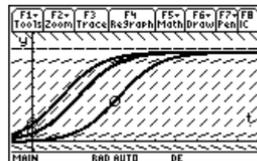
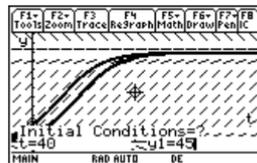
puede desplazar el cursor hasta un punto de la pantalla y pulsar **ENTER**.

Si lo desea, puede utilizar **[F3]** para trazar curvas para las condiciones iniciales especificadas en **Y= Editor**. Sin embargo, no pueden trazarse curvas para una condición inicial seleccionada de forma interactiva.

 **2nd** **[F8]** **40** **ENTER** **45** **ENTER**

 **[F8]** **40** **ENTER** **45** **ENTER**

Visualización



Temas complementarios de gráficos

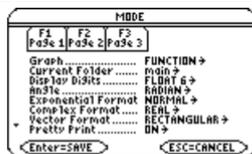
En la pantalla Home (Principal), dibuje la gráfica de la función definida a trozos siguiente: $y = -x$ cuando $x < 0$ y $y = 5 \cos(x)$ si $x \geq 0$. Trace una recta horizontal por la parte superior de la curva coseno y , a continuación, guarde un dibujo de la gráfica representada.

Pasos y pulsaciones

1. Presente el cuadro de diálogo **MODE**.
Seleccione **FUNCTION** para el modo **Graph**.
Seleccione **RADIAN** para el modo **Angle**.

Pulse **MODE** **1** **1** **1** **ENTER**

Visualización



Pasos y pulsaciones

Visualización

2. Presente la pantalla Home. Utilice la orden **Graph** y la función **when** para introducir la función definida a trozos.

Graph when($x < 0$, $-x$,
 $5 * \cos(x)$)

Con **[F4] 2** se selecciona **Graph** en el menú **Other** de la barra de herramientas y se añade automáticamente un espacio.

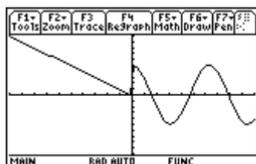
[HOME] [F4] 2 [2nd] [a-lock] WHEN [alpha] [] X
[2nd] [<] 0 [] [(-) X [] 5 [x] [2nd] [COS] X [] []

[] [] [CALC HOME] [F4] 2 WHEN [] X [2nd] [<]
0 [] [(-) X [] 5 [x] [COS] X [] []

3. Ejecute la orden **Graph**, la pantalla Graph aparece automáticamente.

El gráfico emplea las variables de ventana actuales, entendiendo que éstas representan sus valores estándar (**[F6] 6**) en el ejemplo.

Pulse **[ENTER]**



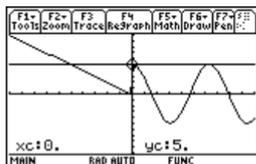
4. Dibuje una recta horizontal sobre la parte superior de la curva coseno.

La calculadora permanece en modo de "recta horizontal" hasta que se selecciona otra operación o se pulsa **[ESC]**.

[2nd] [F7] 5 (hasta situar la recta)

[ENTER]

[F7] 5 (hasta situar la recta) **[ENTER]**



5. Guarde un dibujo del gráfico. Utilice **PIC1** como nombre de variable del dibujo.

Asegúrese de ajustar **Type = Picture**. El ajuste por omisión es **GDB**.

[F1] 2 [] 2 [] 2 [] [alpha] 1 [ENTER] [ENTER]

[F1] 2 [] 2 [] 2 [] PIC1 [ENTER] [ENTER]



Pasos y pulsaciones

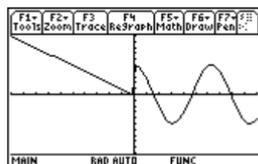
6. Borre la recta horizontal dibujada.

También puede pulsar **[F4]** para volver a dibujar el gráfico.

 **[2nd]** **[F6]** 1

 **[F6]** 1

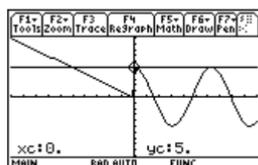
Visualización



7. Abra la variable del dibujo guardado para volver a mostrar el gráfico con la recta.

Asegúrese de ajustar **Type = Picture**. El ajuste por omisión es **GDB**.

Pulse **[F1]** 1 **[D]** 2 (si no aparece, ajuste también Variable = pic1) **[ENTER]**



Tablas

Calcule el valor de la función $y = x^3 - 2x$ en cada número entero comprendido entre -10 y 10. ¿Cuántos cambios de signo hay y dónde se producen?

Pasos y pulsaciones

1. Presente el cuadro de diálogo **MODE**. Para el modo **Graph**, seleccione **FUNCTION**.

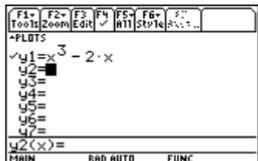
Pulse **[MODE]** **[D]** 1 **[ENTER]**

Visualización



2. Presente y limpie **Y= Editor**. Después defina $y_1(x) = x^3 - 2x$.

Pulse **[Y=]** **[F1]** 8 **[ENTER]** **[ENTER]** X **[^]** 3 **[-]** 2 X **[ENTER]**



Pasos y pulsaciones

3. Establezca los parámetros de tabla siguientes:

tblStart = -10

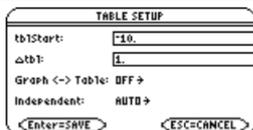
Δ tbl = 1

Graph < - > Table = OFF

Independent = AUTO

Pulse \blacktriangleright [TBLSET] \leftarrow 10 \odot 1 \odot 1 \odot 1 \odot 1
[ENTER]

Visualización



4. Presente la pantalla Table.

Pulse \blacktriangleright [TABLE]

F1	F2	F3	F4	F5	F6
Tools	Setup	1	2	3	4
OK	W1				
-10.	-980.				
-9.	-711.				
-8.	-496.				
-7.	-329.				
-6.	-204.				
x=-10.					
MAIN	RAD	AUTO	FUNC		

5. Desplácese por la tabla. Observe que **y**1 cambia de signo en $x = -1, 1$ y 2 .

Para desplazarse por las distintas páginas, utilice 2^{nd} \odot y 2^{nd} \ominus .

Pulse \odot y \ominus según se requiera

F1	F2	F3	F4	F5	F6
Tools	Setup	1	2	3	4
OK	W1				
-1.	1.				
0.	0.				
1.	-1.				
2.	4.				
3.	21.				
x=3.					
MAIN	RAD	AUTO	FUNC		

6. Veamos con más detalle lo que ocurre entre $x = -2$ y $x = -1$ cambiando los parámetros de tabla a:

tblStart = -2

Δ tbl = .1

Pulse $F2$ \leftarrow 2 \odot .1 [ENTER] [ENTER]

F1	F2	F3	F4	F5	F6
Tools	Setup	1	2	3	4
OK	W1				
-2.	-3.059				
-1.9	-2.232				
-1.7	-1.513				
-1.6	-.896				
x=-2.					
MAIN	RAD	AUTO	FUNC		

Pantalla dividida

Divida la pantalla para mostrar **Y= Editor** y la pantalla **Graph**. Después, estudie el comportamiento de una función polinómica a medida que cambian sus coeficientes.

Pasos y pulsaciones

1. Presente el cuadro de diálogo **MODE**.

Para **Graph**, seleccione **FUNCTION**.

Para **Split Screen**, seleccione **LEFT-RIGHT**.

Para **Split 1 App**, seleccione **Y= Editor**.

Para **Split 2 App**, seleccione **Graph**.

Pulse [MODE] \odot 1 $F2$ \odot 3 \odot 2 \odot 4 [ENTER]

Visualización



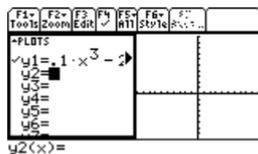
Pasos y pulsaciones

Visualización

2. Vacíe **Y= Editor** y desactive todos los gráficos estadísticos. Después, defina $y_1(x) = .1x^3 - 2x + 6$.

Un reborde grueso alrededor de **Y= Editor** indica que está activado. En este caso, la línea de entrada cruza toda la pantalla.

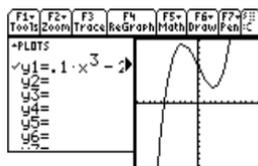
Pulse **[F1] 8 [ENTER] [F5] 5 [ENTER] .1 X \wedge 3 \square 2 X \square + 6 [ENTER]**



3. Seleccione la ventana de visualización **ZoomStd**, que conmuta a la pantalla Graph y representa la función.

Ahora el reborde grueso está alrededor de la pantalla Graph.

Pulse **[F2] 6**



4. Conmute a **Y= Editor**. Después edite **y1(x)** para cambiar $.1x^3$ por $.5x^3$.

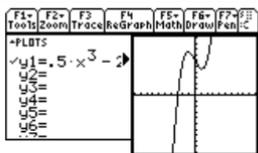
[2nd] [⇐] es la segunda función de **[APPS]**. El reborde grueso está alrededor de **Y= Editor**.

Pulse **[2nd] [⇐] \leftarrow [ENTER] \downarrow \downarrow \downarrow \leftarrow 5 [ENTER]**

5. Conmute a la pantalla Graph, que vuelve a representar la función editada.

El reborde grueso está alrededor de la pantalla Graph.

Pulse **[2nd] [⇐]**



6. Conmute a **Y= Editor**. Después abra **Window Editor** en su lugar.

Pulse **[2nd] [⇐] \blacktriangleright [WINDOW]**



7. Abra la pantalla Home. Después salga a una pantalla Home de tamaño completo.

[2nd] [QUIT] [HOME]

[2nd] [QUIT]

Data/Matrix Editor

Utilice **Data/Matrix Editor** para crear una lista de una sola columna y, a continuación, añada una segunda columna de datos. La lista (que sólo puede incluir una columna) se convierte automáticamente en una variable de datos (que puede incluir varias columnas).

Pasos y pulsaciones

1. Pulse **[APPS]** Inicie **Data/Matrix Editor** y cree una lista nueva denominada **TEMP**.

Pulse 3 \downarrow 3 \leftarrow TEMP **[ENTER]** **[ENTER]**



2. Introduzca una columna numérica. A continuación, desplace el cursor una celda hacia arriba (para verificar que el valor de la celda resaltada aparece en la línea de entrada).

LIST aparece en la parte superior izq. para indicar que se trata de una lista.

Para introducir información en una celda, puede utilizar \leftarrow en lugar de **[ENTER]**.

Pulse 1 **[ENTER]** 2 **[ENTER]** 3 **[ENTER]** 4 **[ENTER]** 5 **[ENTER]** 6 **[ENTER]** \leftarrow

F1- Tools	F2 Plot Setup	F3 Cell	F4 Header	F5 Calc	F6- Unit	F7 Stat
LIST						
	c1	c2	c3			
4	4					
5	5					
6	6					
7						

3. Desplácese hasta la columna 2 y defina la cabecera de columna de forma que su valor sea el doble que el de la columna 1.

En la parte superior izq. aparece **DATA** para indicar que la lista se ha convertido en una variable de datos.

\leftarrow **[F4]** 2 \times alpha C 1 **[ENTER]**

\leftarrow **[F4]** 2 \times C 1 **[ENTER]**

F1- Tools	F2 Plot Setup	F3 Cell	F4 Header	F5 Calc	F6- Unit	F7 Stat
DATA						
	c1	c2	c3			
4	4	8				
5	5	10				
6	6	12				
7						

$\mathbf{\leftarrow}$ indica que la celda es una columna definida.

4. Desplace el cursor hasta la celda de cabecera de la columna 2 para mostrar la definición en la línea de entrada.

Si el cursor se encuentra en la celda de cabecera, no necesita pulsar **[F4]** para definirla. Simplemente comience a escribir la expresión.

Pulse **[2nd]** \leftarrow \leftarrow

F1- Tools	F2 Plot Setup	F3 Cell	F4 Header	F5 Calc	F6- Unit	F7 Stat
DATA						
	c1	$\mathbf{\leftarrow}$ 2*c1	c3			
1	1	2				
2	2	4				
3	3	6				
4	4	8				

Pasos y pulsaciones

Visualización

5. Vacíe el contenido de la variable.

La variable de datos no se convierte de nuevo en lista con sólo borrar los datos.

Pulse **[F1] 8 [ENTER]**

	F1	F2	F3	F4	F5	F6	F7
	100%	Prot Setup	[Ct]	Mode	CtCt	[D]	Stat
DATA							
1	c1		c2		c3		
2							
3							
4							

r1c1=
MAIN RAD AUTO FUNC

Nota: Si no necesita guardar la variable actual, puede utilizarla como *memoria intermedia*. La próxima vez que necesite emplear una variable provisionalmente, limpie la variable actual y vuelva a utilizarla. Es una forma cómoda de introducir los datos que va a utilizar temporalmente sin necesidad de gastar memoria creando una nueva variable cada vez.

Gráficos estadísticos y de datos

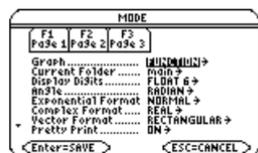
A partir de un muestreo de siete ciudades, introduzca datos en los que la población se asocie a edificios con más de 12 plantas. Utilizando regresión lineal, halle y represente gráficamente ecuaciones que se ajusten a los datos. En una ecuación de regresión, haga una estimación de la cantidad de edificios de más de 12 plantas que puede haber en una ciudad con 300.000 habitantes.

Pasos y pulsaciones

Visualización

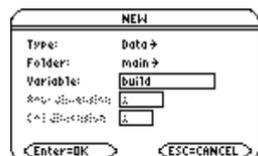
1. Presente el cuadro de diálogo **MODE**. Para el modo **Graph**, seleccione **FUNCTION**.

Pulse **[MODE] Ⓛ 1 [ENTER]**



2. Pulse **[APPS]** para presentar el **Data/Matrix Editor**. Cree una nueva variable de datos y asígnele el nombre **BUILD**.

Pulse **3 Ⓛ Ⓛ BUILD [ENTER] [ENTER]**



Pasos y pulsaciones

Visualización

3. Utilizando los siguientes datos del muestreo, introduzca la población en la columna 1.

Pob. (en millares)	Edif > 12 plantas
150	4
500	31
800	42
250	9
500	20
750	55
950	73

Pulse 150 **ENTER** 500 **ENTER** 800 **ENTER** 250
ENTER 500 **ENTER** 750 **ENTER** 950 **ENTER**

F1 Tools	F2 Plot Setup	F3 Cell	F4 Header	F5 Calc	F6 Unit	F7 Stat
DATA	c1	c2	c3			
5	500					
6	750					
7	950					
8						
r8c1=						
MAIN DEGRABO FUNC						

4. Desplace el cursor hasta la fila 1 de la columna 2 (r1c2). A continuación, introduzca el número de edificios correspondiente.

⬆ **⬅** desplaza el cursor a la parte superior de la página. Tras escribir los datos correspondientes en una celda, puede pulsar **ENTER** o **⬇** para introducir los datos y desplazar el cursor hasta la celda siguiente. Pulsando **⬅**, los datos se introducen y el cursor se desplaza a la celda anterior.

⬆ **⬅** 4 **ENTER** 31 **ENTER** 42 **ENTER** 9
ENTER 20 **ENTER** 55 **ENTER** 73 **ENTER**

⬆ **2nd** **⬅** 4 **ENTER** 31 **ENTER** 42 **ENTER** 9
ENTER 20 **ENTER** 55 **ENTER** 73 **ENTER**

F1 Tools	F2 Plot Setup	F3 Cell	F4 Header	F5 Calc	F6 Unit	F7 Stat
DATA	c1	c2	c3			
5	500	20				
6	750	55				
7	950	73				
8						
r8c2=						
MAIN DEGRABO FUNC						

5. Desplace el cursor hasta la fila 1 de la columna 1 (r1c1). Ordene los datos de población de manera ascendente.

De esta forma se ordena la columna 1 y se ajusta el resto de columnas para que mantengan el orden establecido en la columna 1. Esta clasificación es esencial para mantener la relación entre las columnas de datos.

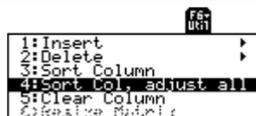
Para ordenar la columna 1, el cursor puede situarse en cualquier parte de dicha columna. En este ejemplo puede pulsar



para ver las cuatro primeras filas.

[F6] 4

[F6] 4



F1	F2	F3	F4	F5	F6	F7
Tools	Plot Setup	Cell Header	Calc	Util	Stat	
DATA						
	c1	c2	c3			
1	150	4				
2	250	9				
3	500	31				
4	500	20				

r1c1=150
MAIN RAD AUTO FUNC

6. Presente el cuadro de diálogo **Calculate**.

Ajuste:

Calculation Type = MedMed

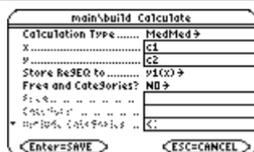
x = C1

y = C2

Store RegEQ to = y1(x)

[F5] 7 C 1 C2

[F5] 7 C1 C2



7. Realice la operación para mostrar la ecuación de regresión MedMed.

Según lo especificado en el cuadro de diálogo **Calculate**, esta ecuación se almacena en **y1(x)**.

Pulse



8. Cierre la pantalla **STAT VARS**. Aparece el **Data/Matrix Editor**.

Pulse

Pasos y pulsaciones

9. Presente el cuadro de diálogo **Calculate**.

Establezca:

Calculation Type = LinReg

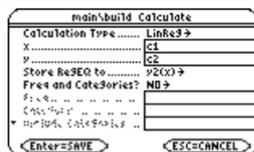
x = C1

y = C2

Store RegEQ to = y2(x)

Pulse **F5** \rightarrow 5 \rightarrow \rightarrow \rightarrow \rightarrow \rightarrow **ENTER**

Visualización



10. Realice la operación para presentar la ecuación de regresión LinReg.

Esta ecuación se almacena en **y2(x)**.

Pulse **ENTER**



11. Cierre la pantalla **STAT VARS**. Aparece el **Data/Matrix Editor**.

Pulse **ENTER**

12. Presente la pantalla Plot Setup.

Plot 1 se resalta de forma predeterminada.

F3 permite eliminar valores seleccionados de Plot.

Pulse **F2**



13. Defina **Plot 1** como:

Plot Type = Scatter

Mark = Box

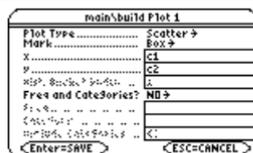
x = C1

y = C2

Observe las similitudes entre este cuadro de diálogo y el de **Calculate**.

F1 \rightarrow 1 \rightarrow 1 \rightarrow C **alpha** 1 \rightarrow **alpha** C2

F1 \rightarrow 1 \rightarrow 1 \rightarrow C1 \rightarrow C2



14. Guarde la definición del gráfico y regrese a la pantalla Plot Setup.

Observe la notación resumida para la definición de **Plot 1**.

Pulse **ENTER** **ENTER**



15. Presente **Y= Editor**. Para **y1(x)**, que es la ecuación de regresión MedMed, ajuste el estilo de visualización en **Dot**.

Nota: Dependiendo del contenido anterior de **Y= Editor**, puede necesitar mover el cursor hasta **y1**.



Cuando **PLOTS 1** aparece en la parte superior de la pantalla, indica que se ha seleccionado **Plot 1**.

Tenga en cuenta que **y1(x)** e **y2(x)** fueron seleccionadas cuando se almacenaron las ecuaciones de regresión.

 \blacklozenge [Y=] [2nd] [F6] 2

 \blacklozenge [Y=] [F6] 2

16. Desplácese hacia arriba para resaltar **Plot 1**.

La definición resumida que aparece es igual a la de la pantalla Plot Setup.

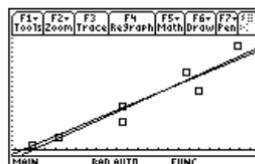
Pulse \odot



17. Utilice **ZoomData** para representar **Plot 1** y las ecuaciones de regresión **y1(x)** e **y2(x)**.

ZoomData examina los datos de los gráficos estadísticos seleccionados y ajusta la ventana de visualización para que incluya todos los puntos.

Pulse [F2] 9



18. Regrese a la sesión actual de **Data/Matrix Editor**.

Pulse [APPS] \odot [ENTER] [ENTER]

19. Introduzca un título para la columna 3.
Defina la cabecera de la columna 3 de acuerdo con los valores estimados por la recta MedMed.

Para introducir un título, el cursor debe resaltar la celda de título situada en el extremo superior de la columna.

$\boxed{F4}$ permite definir la cabecera desde cualquier parte de la columna. Si el cursor se encuentra en la celda de cabecera, no es necesario pulsar $\boxed{F4}$.

$\boxed{\text{F4}}$ $\boxed{\text{Y1}}$ $\boxed{\alpha}$ $\boxed{C1}$ $\boxed{\text{ENTER}}$

$\boxed{\text{F4}}$ $\boxed{\text{Y1}}$ $\boxed{\alpha}$ $\boxed{C1}$ $\boxed{\text{ENTER}}$

20. Introduzca un título para la columna 4.
Defina la cabecera de la columna 4 para los residuos (diferencia entre los valores presentados y los estimados) de MedMed.

$\boxed{\text{F4}}$ $\boxed{\text{Y1}}$ $\boxed{\alpha}$ $\boxed{C1}$ $\boxed{\text{ENTER}}$ $\boxed{\alpha}$ $\boxed{C2}$ $\boxed{-}$ $\boxed{\alpha}$ $\boxed{C3}$ $\boxed{\text{ENTER}}$

$\boxed{\text{F4}}$ $\boxed{\text{Y1}}$ $\boxed{\alpha}$ $\boxed{C1}$ $\boxed{\text{ENTER}}$ $\boxed{\alpha}$ $\boxed{C2}$ $\boxed{-}$ $\boxed{\alpha}$ $\boxed{C3}$ $\boxed{\text{ENTER}}$

F1- Tools	F2 Plot Setup	F3 c1	F4 Header	F5 Calc	F6 Util	F7 Stat
DATA		med	resid			
		c2	c3	c4		
1	4	3.3333	.66667			
2	9	10.889	-1.889			
3	31	29.778	1.2222			
4	20	29.778	-9.778			
c4=c2-c3						
MAIN RAD AUTO FUNC						

21. Introduzca un título para la columna 5.
Defina la cabecera de la columna 5 como valores estimados mediante la recta LinReg.

$\boxed{\text{F4}}$ $\boxed{\text{Y1}}$ $\boxed{\alpha}$ $\boxed{C1}$ $\boxed{\text{ENTER}}$ $\boxed{\alpha}$ $\boxed{C2}$ $\boxed{-}$ $\boxed{\alpha}$ $\boxed{C3}$ $\boxed{\text{ENTER}}$

$\boxed{\text{F4}}$ $\boxed{\text{Y1}}$ $\boxed{\alpha}$ $\boxed{C1}$ $\boxed{\text{ENTER}}$ $\boxed{\alpha}$ $\boxed{C2}$ $\boxed{-}$ $\boxed{\alpha}$ $\boxed{C3}$ $\boxed{\text{ENTER}}$

22. Introduzca un título para la columna 6.
Defina la cabecera de la columna 6 como residuos de LinReg.

$\boxed{\text{F4}}$ $\boxed{\text{Y1}}$ $\boxed{\alpha}$ $\boxed{C1}$ $\boxed{\text{ENTER}}$ $\boxed{\alpha}$ $\boxed{C2}$ $\boxed{-}$ $\boxed{\alpha}$ $\boxed{C3}$ $\boxed{\text{ENTER}}$

$\boxed{\text{F4}}$ $\boxed{\text{Y1}}$ $\boxed{\alpha}$ $\boxed{C1}$ $\boxed{\text{ENTER}}$ $\boxed{\alpha}$ $\boxed{C2}$ $\boxed{-}$ $\boxed{\alpha}$ $\boxed{C3}$ $\boxed{\text{ENTER}}$

F1- Tools	F2 Plot Setup	F3 c1	F4 Header	F5 Calc	F6 Util	F7 Stat
DATA		resid	lin	resid		
		c4	c5	c6		
1	.66667	.22169	3.7783			
2	-1.889	8.3778	.62224			
3	1.2222	28.768	2.232			
4	-9.778	28.768	-8.768			
c6=c2-c5						
MAIN RAD AUTO FUNC						

23. Presente la pantalla Plot Setup y anule **Plot 1**.

Pulse **F2** **F4**

24. Resalte **Plot 2** y defínalo como:

Plot Type = Scatter

Mark = Box

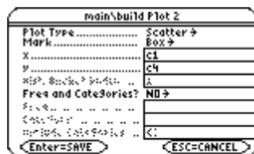
x = C1

y = C4 (MedMed residuals)

F1 **C** **alpha** **1** **alpha** **C4** **ENTER**

ENTER

F1 **C1** **C4** **ENTER** **ENTER**



25. Resalte **Plot 3** y defínalo como:

Plot Type = Scatter

Mark = Plus

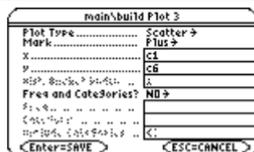
x = C1

y = C6 (LinReg residuals)

F1 **3** **C** **alpha** **1** **alpha** **C6**

ENTER **ENTER**

F1 **3** **C1** **C6** **ENTER** **ENTER**



26. Presente **Y= Editor** y desactive todas las funciones **y(x)**.

En **F5**, seleccione **3:Functions Off**, en lugar de **1:All Off**.

Los gráficos 2 y 3 continúan estando seleccionados.

Pulse **Y=** **F5** **3**

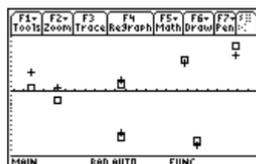


27. Utilice **ZoomData** para representar gráficamente los residuos.

indica los residuos de MedMed;

+ indica los residuos de LinReg.

Pulse **F2** **9**



28. Presente la pantalla Home.

HOME

[CALC HOME]

4. Escriba las instrucciones siguientes.

Request "Enter an integer",n

Presenta un cuadro de diálogo que solicita "Enter an integer", espera a que el usuario introduzca un valor y lo almacena (como una cadena) en la variable n.

expr(n)→n

Convierte la cadena en una expresión numérica.

0→temp

Crea una variable denominada temp asignándole el valor 0.

For i,1,n,1

Inicia un bucle For basado en la variable i. La primera vez que se recorre el bucle, i = 1. Al final del bucle, i se incrementa en 1. El bucle continúa hasta que i > n.

temp+i→temp

Añade el valor actual de i a temp.

EndFor

Marca el final del bucle For.

Disp temp

Presenta el valor final de temp.

Escriba las instrucciones según se indica.

Pulse **[ENTER]** al final de cada línea.

F1-	F2-	F3/FN+	F5	F6-
Tools	Control	/ / /	Find...	Mode

```

i→n
:expr(n)→n
:0→temp
:For i,1,n,1
:  temp+i→temp
:EndFor
:Disp temp
:
:EndPrgm
  
```

MAIN	RAD AUTO	FUNC
------	----------	------

5. Vaya a la pantalla Home e introduzca el nombre del programa seguido de paréntesis.

prog1()

Debe incluir () aun cuando no existan argumentos para el programa.

El programa presenta un cuadro de diálogo con el mensaje que se introdujo en la instrucción correspondiente.

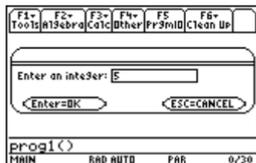
 **[HOME]** **[2nd]** **[a-lock]** **PROG** **[alpha]** **1** **[(]** **[)]**
[ENTER]

 **[♦]** **[CALC HOME]** **PROG1** **[(]** **[)]** **[ENTER]**

Pasos y pulsaciones

6. Escriba 5 en el cuadro de diálogo mostrado.
Pulse 5

Visualización

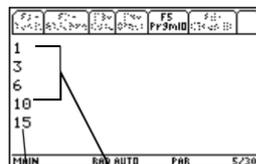


7. Continúe con el programa. La orden **Disp** presenta el resultado en la pantalla Program E/S.

El resultado es la suma de los enteros de 1 a 5.

Aunque la pantalla Program E/S es similar a la pantalla Home, se emplea exclusivamente para las entradas y salidas del programa. En la pantalla Program E/S no pueden realizarse operaciones.

Pulse **[ENTER]** **[ENTER]**



Los resultados de otras ejecuciones pueden aparecer en la pantalla.

Resultado para el entero 5.

8. Abandone la pantalla Program E/S y regrese a Home.

También puede pulsar **[ESC]**, **[2nd]** **[QUIT]**, o bien



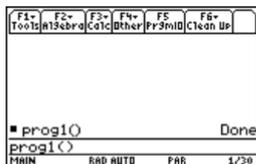
[HOME]



[CALC HOME]

para regresar a la pantalla Home.

Pulse **[F5]**



Text Editor

Inicie una sesión nueva de **Text Editor**. Después, practique utilizando **Text Editor** y escribiendo el texto que desee. A medida que escriba, practique moviendo el cursor de texto y corrigiendo los errores que vaya cometiendo.

Pasos y pulsaciones

1. Empiece una sesión nueva de **Text Editor**.
Pulse 3

Visualización



2. Cree una variable de texto con el nombre **TEST**, donde se almacenará automáticamente el texto que introduzca en esta sesión.



Utilice la carpeta **MAIN**, mostrada como carpeta predeterminada en el recuadro de diálogo **NEW**.

Después de escribir en un cuadro de entrada como **Variable**, debe pulsar **[ENTER]** dos veces.

Pulse **[<] TEST [ENTER] [ENTER]**

3. Escriba un texto de ejemplo.

- Para escribir una letra mayúscula, pulse **[↑]** y la letra.

Solo para la TI-89 Titanium:

- Para escribir un espacio, pulse **[alpha] [⌵]** (función alfa de la tecla **[⌵]**).
- Para escribir un punto, pulse **[alpha]** para quitar el bloqueo alfabético, pulse **[.]**, y por último pulse **[2nd] [a-lock]** para volver a activar el bloqueo.



Practique editando el texto con:

- La tecla del cursor para mover el cursor de texto.
- **[←]** o **[→]** **[DEL]** para borrar el carácter a la izquierda o la derecha del cursor, respectivamente.

 **[2nd] [a-lock]** escriba lo que desee

 escriba lo que desee

4. Salga de **Text Editor** y presente la pantalla Home.

El texto se almacena automáticamente a medida que escribe. Por ello, no es necesario que lo guarde manualmente antes de salir de **Text Editor**.



[HOME]



[CALC HOME]

5. Vuelva a la sesión actual de **Text Editor**.
Observe que el contenido aparece exactamente como lo dejó.

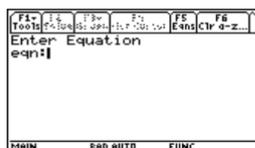
Pulse [2nd] [⇐]

Analizador numérico

Consideremos la ecuación $a = (m_2 - m_1) / (m_2 + m_1) * g$, donde los valores conocidos son $m_2 = 10$ y $g = 9.8$. Si suponemos que $a = 1/3 g$, halle el valor de m_1 .

1. Presente el **Numeric Solver**.

Pulse [APPS]



2. Introduzca la ecuación.

Al pulsar [ENTER] o ⏏, la pantalla presenta una lista de las variables utilizadas en la ecuación.



[alpha] A [=] [(] [alpha] M2 [-] [alpha] M1 [)] [÷]

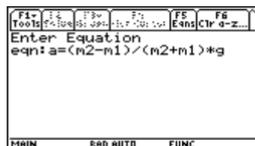
[(] [alpha] M2 [+] [alpha] M1 [)] [×] [alpha] G

[ENTER]



A [=] [(] M2 [-] M1 [)] [÷] [(] M2 [+] M1

[)] [×] G [ENTER]



3. Introduzca valores para cada variable, excepto la incógnita m_1 .

Defina primero m_2 y g . A continuación defina a (debe definir g antes de poder definir a en función de g .) Acepte el valor predeterminado para $bound$. Si una variable se ha definido previamente, su valor se muestra como valor predeterminado.

 \odot 10 \odot \odot 9.8 \odot \odot \odot α G \odot 3
 \odot 10 \odot \odot 9.8 \odot \odot \odot G \odot 3

```

F1- F2- F3- F4- F5- F6-
Tool|Sol|Graph|Get Cursor|EAns|Clv|a-z.
a=(m2-m1)/(m2+m1)*g
a=g/3
m2=10.
m1=
g=9.8
bound=C-1. e14, 1. e14}
MAIN RAD AUTO FUNC
  
```

4. Desplace el cursor hasta la incógnita m_1 .

De forma opcional, puede introducir una aproximación inicial para m_1 . Incluso en el caso de introducir un valor para todas las variables, el analizador numérico halla la solución de las variables marcadas por el cursor.

Pulse \odot \odot

```

F1- F2- F3- F4- F5- F6-
Tool|Sol|Graph|Get Cursor|EAns|Clv|a-z.
a=(m2-m1)/(m2+m1)*g
a=3.266666666666667
m2=10.
m1=
g=9.8
bound=C-1. e14, 1. e14}
MAIN RAD AUTO FUNC
  
```

$g/3$ se calcula al moverse el cursor fuera de la línea.

5. Halle el valor de la incógnita.

Para comprobar la precisión de la solución, los lados izquierdo y derecho de la ecuación se calculan por separado. La diferencia se muestra como $left-rt=0$. Si la solución es precisa, $left-rt=0$.

Pulse $\boxed{F2}$

```

F1- F2- F3- F4- F5- F6-
Tool|Sol|Graph|Get Cursor|EAns|Clv|a-z.
a=(m2-m1)/(m2+m1)*g
a=3.266666666666667
m2=10.
m1=5.7
g=9.8
bound=C-1. e14, 1. e14}
left-rt=0.
MAIN RAD AUTO FUNC
  
```

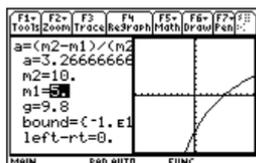
■ marca los valores calculados.

6. Represente gráficamente la solución mediante una ventana de visualización **ZoomStd**.

La gráfica se presenta en una pantalla dividida. Puede explorarla desplazándose por ella, utilizando Zoom, etc.

La variable marcada por el cursor (incógnita m_1) se representa en el eje x ; $left-rt$ se representa en el eje y .

Pulse $\boxed{F3}$ 3



7. Vuelva al **Numeric Solver** y salga de la pantalla dividida.

Puede pulsar $\boxed{\text{ENTER}}$ o \odot para presentar de nuevo la lista de variables.

Pulse $\boxed{2\text{nd}} \boxed{[+]} \boxed{\text{F3}} \boxed{2}$

Bases de numeración

Calcule 10 binario (base 2) + F hexadecimal (base 16) + 10 decimal (base 10). A continuación, utilice el operador \blacktriangleright para convertir un número entero de una base a otra. Para terminar, observe cómo la modificación del modo Base afecta a los resultados presentados.

1. Presente el recuadro de diálogo **MODE**, página 2. En modo **Base**, seleccione **DEC** como base de numeración predeterminada.

Los resultados en números enteros se presentan de acuerdo con el estado del modo **Base**. Los resultados fraccionarios y de coma flotante siempre se presentan en forma decimal.

Pulse $\boxed{\text{MODE}} \boxed{\text{F2}}$ (utilice \odot para pasar al modo **Base**) $\blacktriangleright \boxed{1} \boxed{\text{ENTER}}$



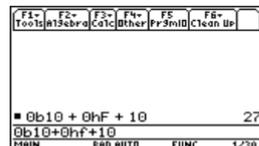
2. Calcule $0b10+0hf+10$.

Para introducir un número binario o hexadecimal, debe utilizar el prefijo $0b$ o $0h$ (cero y la letra B o H). De no ser así, la entrada se considera número decimal.

Nota: El prefijo $0b$ o $0h$ es un cero, no la letra O, seguido de B o H.

$\boxed{0} \boxed{\alpha} \boxed{B} \boxed{10} \boxed{+} \boxed{0} \boxed{2\text{nd}} \boxed{[a\text{-lock}]} \boxed{HF} \boxed{\alpha} \boxed{+} \boxed{10} \boxed{\text{ENTER}}$

$\boxed{0} \boxed{B} \boxed{10} \boxed{+} \boxed{0} \boxed{HF} \boxed{+} \boxed{10} \boxed{\text{ENTER}}$



3. Suma 1 al resultado y conviértalo a binario.

$\boxed{2nd} \boxed{\blacktriangleright}$ presenta el operador de conversión \blacktriangleright .

$\boxed{\text{☎}} \boxed{+} \boxed{1} \boxed{2nd} \boxed{\blacktriangleright} \boxed{2nd} \boxed{[a-lock]} \boxed{BIN} \boxed{\alpha} \boxed{ENTER}$

$\boxed{\text{☎}} \boxed{+} \boxed{1} \boxed{2nd} \boxed{\blacktriangleright} \boxed{BIN} \boxed{ENTER}$

4. Suma 1 al resultado y conviértalo a hexadecimal.

$\boxed{\text{☎}} \boxed{+} \boxed{1} \boxed{2nd} \boxed{\blacktriangleright} \boxed{2nd} \boxed{[a-lock]} \boxed{HEX} \boxed{\alpha} \boxed{ENTER}$

$\boxed{\text{☎}} \boxed{+} \boxed{1} \boxed{2nd} \boxed{\blacktriangleright} \boxed{HEX} \boxed{ENTER}$

5. Suma 1 al resultado y déjelo en la base de numeración decimal por omisión.

Los resultados utilizan el prefijo 0b o 0h para identificar la base.

Pulse $\boxed{+} \boxed{1} \boxed{ENTER}$

F1+	F2+	F3+	F4+	F5	F6+
Tools	13cbrj	C1c	Other	Pr3mD	Clean Up
■	0b10 + 0hF + 10				27
■	(27 + 1)►Bin				0b11100
■	(0b11100 + 1)►Hex				0h1D
■	0h1D + 1				30
ans(1)+1					
MAIN		RAD AUTO		FUNC 4/30	

6. Cambie el modo **Base** a **HEX**.

Cuando **Base = HEX** o **BIN**, la magnitud de un resultado está restringida a determinadas limitaciones de tamaño.

Pulse $\boxed{MODE} \boxed{F2}$ (utilice \blacktriangledown para pasar al modo **Base**) $\blacktriangledown \boxed{2} \boxed{ENTER}$

7. Calcule $0b10+0hF+10$.

$\boxed{\text{☎}} \boxed{0} \boxed{\alpha} \boxed{B} \boxed{10} \boxed{+} \boxed{0} \boxed{2nd} \boxed{[a-lock]} \boxed{HF} \boxed{\alpha} \boxed{+} \boxed{10} \boxed{ENTER}$

$\boxed{\text{☎}} \boxed{0} \boxed{B} \boxed{10} \boxed{+} \boxed{0} \boxed{HF} \boxed{+} \boxed{10} \boxed{ENTER}$

F1+	F2+	F3+	F4+	F5	F6+
Tools	13cbrj	C1c	Other	Pr3mD	Clean Up
■	0b10 + 0hF + 10				27
■	(27 + 1)►Bin				0b11100
■	(0b11100 + 1)►Hex				0h1D
■	0h1D + 1				30
■	0b10 + 0hF + 10				0h1B
0b10+0hf+10					
MAIN		RAD AUTO		FUNC 5/30	

8. Cambie el modo **Base** a **BIN**.

Pulse $\boxed{MODE} \boxed{F2}$ (utilice \blacktriangledown para pasar al modo **Base**) $\blacktriangledown \boxed{3} \boxed{ENTER}$

9. Vuelva a introducir $0b10+0hF+10$.

Pulse \boxed{ENTER}

F1+	F2+	F3+	F4+	F5	F6+
Tools	13cbrj	C1c	Other	Pr3mD	Clean Up
■	(27 + 1)►Bin				0b11100
■	(0b11100 + 1)►Hex				0h1D
■	0h1D + 1				30
■	0b10 + 0hF + 10				0h1B
■	0b10 + 0hF + 10				0b11011
0b10+0hf+10					
MAIN		RAD AUTO		FUNC 6/30	

Gestión de la memoria y de las variables

Asigne valores a diversos tipos de datos de variable. Utilice la pantalla **VAR-LINK** para ver una lista de las variables definidas. Desplace una variable a la memoria del archivo de datos del usuario y explore de qué forma puede o no acceder a una variable archivada (las variables archivadas se bloquean automáticamente). Por último, extraiga la variable del archivo y elimine las variables no usadas para que no ocupen espacio en memoria.

Pasos y pulsaciones

Visualización

1. En la pantalla Home, asigne variables de los siguientes tipos.

Expresión: $5 \rightarrow x1$

Función: $x^2+4 \rightarrow f(x)$

Lista: $\{5, 10\} \rightarrow L1$

Matriz: $[30,25] \rightarrow m1$

F1- Tools	F2- 1/2cbrcj	F3- CclC	F4- Other	F5- Pr2mID	F6- Clean UP
■ $5 \rightarrow x1$ 5					
■ $x^2 + 4 \rightarrow f(x)$ Done					
■ $\{5 \ 10\} \rightarrow L1$ $\{5 \ 10\}$					
■ $[30 \ 25] \rightarrow m1$ $[30 \ 25]$					
[30, 25] → m1					
MAIN		RND AUTO		FUNC	4/20

[HOME] [CLEAR] 5 [STO] X1 [ENTER] X \wedge 2 +
4 [STO] [alpha] F [X] [ENTER] [2nd] [i] 5 []
10 [2nd] [i] [STO] [alpha] L1 [ENTER] [2nd] [i] 30
[] 25 [2nd] [i] [STO] [alpha] M1 [ENTER]

[♦] [CALC HOME] [CLEAR] 5 [STO] X1 [ENTER] X
2 + 4 [STO] F [X] [ENTER] [2nd] [i] 5
[] 10 [2nd] [i] [STO] L1 [ENTER] [2nd] [i] 30 []
25 [2nd] [i] [STO] M1 [ENTER]

2. Supongamos que empieza a realizar una operación con una variable de función, pero no recuerda su nombre.

5*:

Pulse 5 [x]

3. Presente la pantalla **VAR-LINK**.

En este ejemplo, se presupone que las variables asignadas anteriormente son las únicas que están definidas.

F1- Mand34	F2- View	F3- Link	F4- All	F5- Contents	F6- FlashAPP	F7
VAR-LINK [ENT]						
VAR-LINK			FUNC 19			
f			MAT 12			
m1			MAT 12			
x1			EXPR 5			

Pulse [2nd] [VAR-LINK]

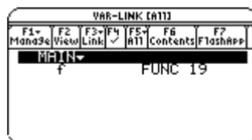
Pasos y pulsaciones

Visualización

4. Cambie la visualización de la pantalla para mostrar únicamente las variables de función.

Aunque esto no parece muy útil en un ejemplo con cuatro variables, tenga en cuenta lo cómodo que resultaría si tuviera muchas variables de distintos tipos.

Pulse $\boxed{F2}$ \odot \odot \odot 5 \boxed{ENTER}

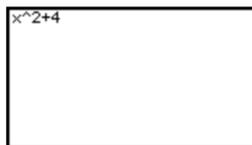


5. Resalte la variable de función **f**, y vea su contenido.

Observe que la función se definió con **f(x)** pero aparece como **f** en la pantalla.

$\boxed{\text{Calculator icon}}$ \odot $\boxed{2nd}$ $\boxed{F6}$

$\boxed{\text{Calculator icon}}$ \odot $\boxed{F6}$



6. Cierre la ventana.

Pulse \boxed{ESC}

7. Con la variable **f** todavía resaltada, cierre **VAR-LINK** para pegar el contenido de la variable en la línea de entrada. Observe que se ha pegado un paréntesis de apertura "(

Pulse \boxed{ENTER}

5*f(

8. Complete la operación.

Pulse 2 $\boxed{)}$ \boxed{ENTER}

5*f(2)

Archivo de una variable

Pasos y pulsaciones

1. Utilice la pantalla **VAR-LINK** para ver una lista de variables. A continuación, resalte la variable apropiada.

El cambio anterior en la visualización deja de tener efecto. La pantalla indica todas las variables definidas.

Pulse **2nd** [VAR-LINK] (utilice \odot para resaltar **x1**)

Visualización

F1- F2	F3- F4	F5- F6	F7
Menu34	ViewLink	RTI	ContentsFlashApp
VAR-LINK (ATT)			
MAIN \blacktriangleright			
f	FUNC	19	
11	LIST	10	
m1	MAT	12	
t1	FIG	26	
x1	EXPR	5	

2. Utilice el menú **F1 Manage** de la barra de herramientas para guardar la variable.

\times indica que la variable está archivada.

Pulse **F1** 8

F1-
2:Copy
3:Rename
4:Move
5:Create Folder
6:Lock
7:Unlock
8:Archive Variable
9:Unarchive Variable

F1- F2	F3- F4	F5- F6	F7
Menu34	ViewLink	RTI	ContentsFlashApp
VAR-LINK (ATT)			
MAIN \blacktriangleright			
f	FUNC	19	
11	MAT	12	
m1	MAT	12	
x x1	EXPR	5	

3. Vuelva a la pantalla Home y utilice la variable guardada para realizar una operación.

HOME 6 \times X1 **ENTER**

\blacktriangleright [CALC HOME] 6 \times X1 **ENTER**

F1-	F2-	F3-	F4-	F5-	F6-
Tools	RT3	Rebro	ColC	Other	Pr3
■ $x^2 + 4 + f(x)$ Done					
■ $\langle 5 \ 10 \rangle + 11$ $\langle 5 \ 10 \rangle$					
■ $\langle 30 \ 25 \rangle + m1$ $\langle 30 \ 25 \rangle$					
■ $5 \cdot f(2)$ 40					
■ $6 \cdot x1$ 30					
6 \times x1					
MAIN BAD AUTO FUNC 6/20					

4. Intente almacenar un valor diferente en la variable guardada.

Pulse 10 **STO** \blacktriangleright X1 **ENTER**

ERROR
Variable or Flash application is locked, protected, or archived
ESC=CANCEL

5. Cancele el mensaje de error.

Pulse **ESC**

6. Utilice **VAR-LINK** para desarchivar la variable.

Pulse **2nd** [VAR-LINK] (utilice \odot para resaltar **x1**) **F1** 9

Pasos y pulsaciones

Visualización

- Vuelva a la pantalla Home y guarde un valor distinto en la variable no archivada.



F1+	F2+	F3+	F4+	F5+	F6+
Tools	Math	Calc	Other	PrsWid	Clean Up
■ <5 10> → 11				<5 10>	
■ <30 25> → m1				[30 25]	
■ 5 · f(2)					40
■ 6 · x1					30
■ 10 → x1					10
10 → x1					
MAIN RAD AUTO FUNC 5/30					

Borrar una variable

Pasos y pulsaciones

Visualización

- Utilice [F5] del menú **VAR-LINK**, de la barra de herramientas para seleccionar todas las variables.

La marca ✓ indica los elementos seleccionados. Observe que también se selecciona la carpeta **MAIN**.

Nota: En vez de pulsar [F5] (si no quiere borrar todas las variables), puede seleccionarlas una por una. Resalte cada elemento que va a borrar y pulse [F4].

Pulse [F5] 1



VAR-LINK (All)						
F1+	F2	F3+	F4	F5+	F6	F7
Mon&9	View	Link	✓ All	Contents	FlashApp	
✓ MAIN						
✓ f				FUNC	10	
✓ 11				MAT	12	
✓ m1				MAT	12	
✓ x1				EXPR	5	

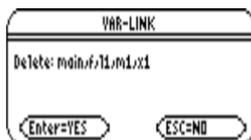
- Utilice [F1] para borrar.

Nota: Puede pulsar [←] (en vez de [F1] 1) para borrar las variables marcadas.

Pulse [F1] 1

- Confirme el borrado.

Pulse [ENTER]



- Con [F5] 1 también se selecciona la carpeta **MAIN**, por lo que aparece un mensaje de error indicando que no puede borrarla. Confirme el mensaje.

Cuando vuelva a presentar **VAR-LINK**, las variables borradas no aparecerán.

Pulse [ENTER]

VAR-LINK (All)						
F1+	F2	F3+	F4	F5+	F6	F7
Mon&9	View	Link	✓ All	Contents	FlashApp	
View						

5. Cierre la pantalla **VAR-LINK** y vuelva a la aplicación en que estuviera (la pantalla Home, en este ejemplo).

Si utiliza **[ESC]** (en vez de **[ENTER]**) para cerrar **VAR-LINK**, el nombre resaltado no se pega en la línea de entrada.

Pulse **[ESC]**

Actividades

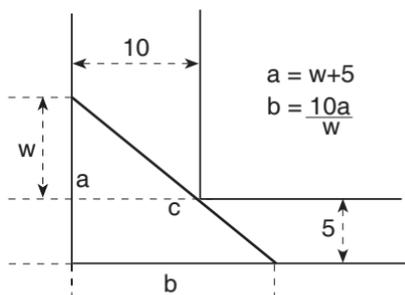
Análisis del problema poste-esquina

En un edificio, un pasillo de diez pies de ancho se une en una de las esquinas con otro pasillo de cinco pies de ancho. Halle la longitud máxima de un poste que pueda hacerse pasar por la esquina sin inclinarlo.

Longitud máxima del poste en el pasillo

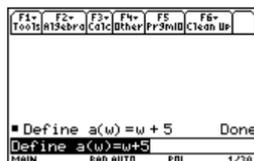
La longitud máxima del poste c es el segmento rectilíneo de menor tamaño que toca la esquina interior y los lados opuestos de los dos pasillos, según se muestra en el siguiente diagrama.

Utilice lados proporcionales y el teorema de Pitágoras para hallar la longitud de c respecto de w . A continuación, halle la raíz de la primera derivada de $c(w)$. El valor mínimo de $c(w)$ es la longitud máxima del poste.



1. Defina la expresión del lado a en función de w y almacénela en $a(w)$.

Nota: Para definir una función, utilice nombres de varios caracteres.



2. Defina la expresión del lado b en función de w y almacénela en $\mathbf{b(w)}$.

F1+	F2+	F3+	F4+	F5	F6+
Tools	R13	Eq	Calc	Other	Pr3
■ Define $a(w) = w + 5$ Done ■ Define $b(w) = \frac{10 \cdot a(w)}{w}$ Done Define $b(w) = 10 \cdot a(w) / w$ Done					
MAIN	RAD	AUTO	PBL	2/30	

3. Defina la expresión del lado c en función de w y almacénela en $\mathbf{c(w)}$.

Introduzca:

Define $c(w) = \sqrt{(a(w)^2 + b(w)^2)}$

F1+	F2+	F3+	F4+	F5	F6+
Tools	R13	Eq	Calc	Other	Pr3
■ Define $b(w) = \frac{10 \cdot a(w)}{w}$ Done ■ Define $c(w) = \sqrt{(a(w))^2 + (b(w))^2}$ Done ne $c(w) = \sqrt{(a(w))^2 + (b(w))^2}$ Done					
MAIN	RAD	AUTO	FUNC	3/30	

4. Utilice la orden **zeros()** para calcular la raíz de la primera derivada de $\mathbf{c(w)}$ y hallar así el valor mínimo de $\mathbf{c(w)}$.

Nota: La longitud máxima del poste es el valor mínimo de $\mathbf{c(w)}$.

F1+	F2+	F3+	F4+	F5	F6+
Tools	R13	Eq	Calc	Other	Pr3
$(5 \cdot 2^{2/3})$ ■ $c(5 \cdot 2^{2/3})$ $\{5 \cdot (2^{2/3} + 1)^{3/2}\}$ ■ $c(5 \cdot 2^{2/3})$ (20.8097) $c(\sqrt{(5 \cdot 2^{2/3})^2 + (20.8097)^2})$					
MAIN	RAD	AUTO	FUNC	6/30	

5. Calcule la longitud máxima exacta del poste.

Introduzca: \mathbf{c} (2nd) [ANS]

F1+	F2+	F3+	F4+	F5	F6+
Tools	R13	Eq	Calc	Other	Pr3
■ zeros($\frac{d}{dw}(c(w), w)$) $(5 \cdot 2^{2/3})$ ■ $c(5 \cdot 2^{2/3})$ $\{5 \cdot (2^{2/3} + 1)^{3/2}\}$					
$c(\text{ans}(1))$					
MAIN	RAD	AUTO	FUNC	5/30	

6. Calcule la longitud máxima aproximada del poste.

Resultado: Aproximadamente 20,8097 pies.

Nota: Copie y pegue el resultado del paso 4 en la línea de entrada, entre los paréntesis de $\mathbf{c()}$, y pulse \square [ENTER].

F1+	F2+	F3+	F4+	F5	F6+
Tools	R13	Eq	Calc	Other	Pr3
$(5 \cdot 2^{2/3})$ ■ $c(5 \cdot 2^{2/3})$ $\{5 \cdot (2^{2/3} + 1)^{3/2}\}$ ■ $c(5 \cdot 2^{2/3})$ (20.8097) $c(\sqrt{(5 \cdot 2^{2/3})^2 + (20.8097)^2})$					
MAIN	RAD	AUTO	FUNC	6/30	

Obtención de la solución de una ecuación de segundo grado

Esta aplicación muestra cómo calcular la solución de una ecuación de segundo grado:

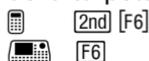
$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Para obtener más información sobre cómo utilizar las órdenes de este ejemplo, consulte *Cálculo simbólico*.

Operaciones para hallar la solución de una ecuación de segundo grado

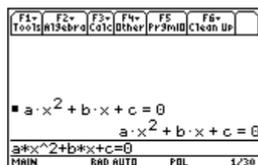
Para hallar la solución de una ecuación de segundo grado, se va a completar el cuadrado del binomio que represente dicha ecuación.

1. Borre todas las variables de un solo carácter de la carpeta seleccionada.



Elija **1:Clear a-z** y pulse **[ENTER]** para confirmar.

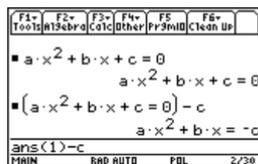
2. En la pantalla Home, introduzca la ecuación general de segundo grado: $ax^2+bx+c=0$.



3. Reste c de ambos lados de la ecuación.

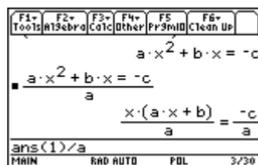


Nota: En este ejemplo se emplea el resultado de la última respuesta para realizar operaciones en la TI-89 Titanium / Voyage™ 200. Con ello, se reducen las pulsaciones y el riesgo de cometer errores.

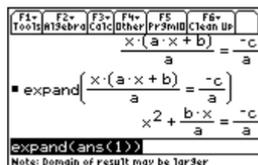


4. Divida ambos lados de la ecuación entre el coeficiente principal a.

Nota: Continúe utilizando la última respuesta (**2nd** [ANS]), al igual que en el paso 3, en los pasos 4 a 9.



5. Utilice la función **expand()** para desarrollar el resultado de la última respuesta.



6. Complete el cuadrado añadiendo $\left(\frac{b}{a}\right)^2$ a ambos lados de la ecuación.

F1-	F2-	F3-	F4+	F5	F6+
Tools	R13	Eq	Col1	Other	Pr3

$$\left(x^2 + \frac{b \cdot x}{a} = \frac{-c}{a}\right) + \left(\frac{b}{a}\right)^2$$

$$x^2 + \frac{b \cdot x}{a} + \frac{b^2}{4 \cdot a^2} = \frac{b^2}{4 \cdot a^2} - \frac{c}{a}$$

ans(1)+((b/a)/2)^2

MMIN ERD AUTO PDL 5/20

7. Factorice el resultado utilizando la función **factor()**.

F1-	F2-	F3-	F4+	F5	F6+
Tools	R13	Eq	Col1	Other	Pr3

$$\text{factor}\left(x^2 + \frac{b \cdot x}{a} + \frac{b^2}{4 \cdot a^2} = \rightarrow\right)$$

$$\frac{(2 \cdot a \cdot x + b)^2}{4 \cdot a^2} = -\frac{(4 \cdot a \cdot c - b^2)}{4 \cdot a^2}$$

factor(ans(1))

MMIN ERD AUTO PDL 6/20

8. Multiplique ambos lados de la ecuación por $4a^2$.

F1-	F2-	F3-	F4+	F5	F6+
Tools	R13	Eq	Col1	Other	Pr3

$$4 \cdot a^2 \cdot \left(x^2 + \frac{b \cdot x}{a} + \frac{b^2}{4 \cdot a^2} = \rightarrow\right)$$

$$4 \cdot a^2 \cdot \left(\frac{(2 \cdot a \cdot x + b)^2}{4 \cdot a^2} = -\frac{(4 \cdot a \cdot c - b^2)}{4}\right)$$

$$(2 \cdot a \cdot x + b)^2 = -(4 \cdot a \cdot c - b^2)$$

4a^2*ans(1)

Note: Domain of result may be larger

9. Obtenga la raíz cuadrada de los dos lados de la ecuación, aplicando las condiciones $a > 0$ y $b > 0$ y $x > 0$.

F1-	F2-	F3-	F4+	F5	F6+
Tools	R13	Eq	Col1	Other	Pr3

$$\sqrt{(2 \cdot a \cdot x + b)^2} = \sqrt{-(4 \cdot a \cdot c - b^2)}$$

$$2 \cdot a \cdot x + b = \sqrt{b^2 - 4 \cdot a \cdot c}$$

(1) | a>0 and b>0 and x>0

MMIN ERD AUTO PDL 8/20

10. Halle el valor de x restando b a ambos lados y dividiendo entre $2a$.

F1-	F2-	F3-	F4+	F5	F6+
Tools	R13	Eq	Col1	Other	Pr3

$$\frac{2 \cdot a \cdot x + b}{2 \cdot a} = \frac{\sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a}$$

$$\frac{2 \cdot a \cdot x - (2 \cdot a - 1) \cdot b}{2 \cdot a} = \frac{\sqrt{b^2 - 4 \cdot a \cdot c}}{2}$$

ans(1)-b

Note: Domain of result may be larger

Nota: Dadas las condiciones impuestas en el paso 9, ésta es sólo una de las dos soluciones que se pueden obtener.

F1-	F2-	F3-	F4+	F5	F6+
Tools	R13	Eq	Col1	Other	Pr3

$$2 \cdot a \cdot x = \sqrt{b^2 - 4 \cdot a \cdot c} - b$$

$$2 \cdot a \cdot x = \sqrt{b^2 - 4 \cdot a \cdot c} - b$$

$$x = \frac{\sqrt{b^2 - 4 \cdot a \cdot c} - b}{2 \cdot a}$$

ans(1)/(2a)

Note: Domain of result may be larger

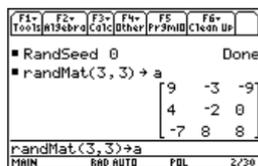
Estudio de una matriz

En esta actividad se muestra cómo realizar varias operaciones con matrices.

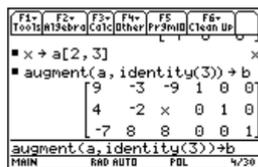
Estudio de una matriz 3x3

Realice estos pasos para generar una matriz aleatoria, hallar y ampliar la matriz identidad y , a continuación, obtener un valor no válido para la matriz inversa.

1. En la pantalla Home, utilice **RandSeed** para establecer el inicio del generador de números aleatorios en el valor predeterminado y, a continuación, emplee **randMat()** para crear una matriz aleatoria 3x3 y almacenarla en a .

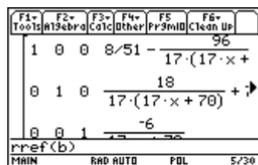


2. Sustituya el elemento $[2,3]$ de la matriz por la variable x , y a continuación, utilice la función **augment()** para ampliar la matriz a con la matriz identidad 3x3 y almacenar el resultado en b .



3. Utilice **rref()** para "reducir las filas" de la matriz b :

En el resultado, la matriz identidad aparecerá en las tres primeras columnas y a^{-1} en las tres últimas.



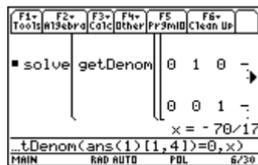
Nota: Para desplazarse por el resultado, utilice el cursor en el área de historia.

4. Halle el valor de x que hará que la matriz inversa no sea válida.

Introduzca:

solve(getDenom([2nd] [ANS] [1,4])=0,x)

Resultado: **$x = -70/17$**



Nota: Para desplazarse por el resultado, utilice el cursor en el área de historia.

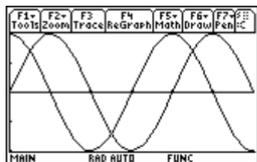
Estudio de $\cos(x) = \sin(x)$

En esta actividad se emplean dos métodos para hallar los puntos en que $\cos(x) = \sin(x)$ para los valores de x comprendidos entre 0 y 3π .

Método 1: Gráfica

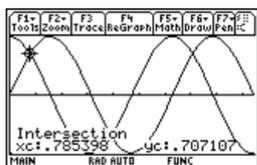
Realice los pasos siguientes para averiguar dónde se cortan las gráficas de las funciones $y_1(x)=\cos(x)$ e $y_2(x)=\sin(x)$.

1. En **Y= Editor**, escriba $y_1(x)=\cos(x)$ e $y_2(x)=\sin(x)$.
2. En **Window Editor**, establezca $x_{\min}=0$ y $x_{\max}=3\pi$.
3. Pulse $\boxed{F2}$ y seleccione **A:ZoomFit**
4. Halle los puntos de intersección de las dos funciones.



Nota: Pulse $\boxed{F5}$ y seleccione **5:Intersection**. Cuando la pantalla solicite la selección de las dos curvas y de los extremos inferior y superior de la intersección A, introduzca la información correspondiente.

5. Anote las coordenadas de x e y. Para hallar otras intersecciones, repita los pasos 4 y 5.

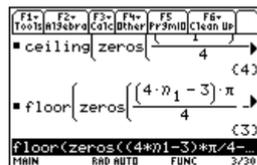
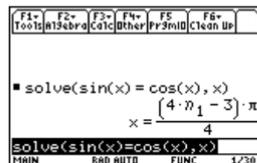


Método 2: Cálculo simbólico

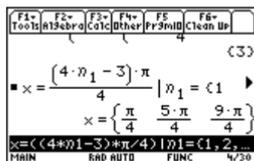
Realice los pasos siguientes para resolver la ecuación $\sin(x)=\cos(x)$ respecto de x.

1. En la pantalla Home, introduzca **solve(sin(x)=cos(x),x)**.
Dando valores enteros a @n1 encontramos los valores de x.
2. Utilizando las funciones **ceiling()** y **floor()**, halle los valores enteros máximo y mínimo que verifican las condiciones.

Nota: Para resaltar la última respuesta, desplace el cursor hasta el área de historia. Pulse $\boxed{\text{ENTER}}$ para copiar el resultado de la solución general.



3. Introduzca la solución general de x y aplique las restricciones para $@n1$, según se indica.



Compare el resultado con el del método 1.

Nota: Para obtener el operador "with":

[1].

Cálculo del área mínima de un paralelepípedo

En esta actividad se explica cómo hallar el área mínima de un paralelepípedo que tiene un volumen constante V . Para obtener información detallada sobre los pasos empleados en este ejemplo, consulte *Cálculo simbólico y Representación gráfica en 3D*.

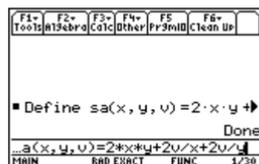
Estudio en una gráfica en 3D del área de un paralelepípedo

Realice los pasos siguientes para definir una función que proporcione el área de un paralelepípedo, dibujar una gráfica en 3D y utilizar la herramienta **Trace** con el fin de hallar un punto próximo al valor mínimo.

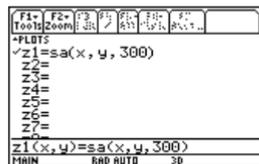
1. En la pantalla Home, defina la función **sa(x,y,v)** para el área del paralelepípedo.

Introduzca:

define sa(x,y,v)=2*x*y+2v/x+2v/y



2. Seleccione el modo 3D Graph. A continuación, introduzca la función **z1(x,y)**, como se indica en el ejemplo, con volumen $v=300$.



3. Ajuste las variables de ventana en:

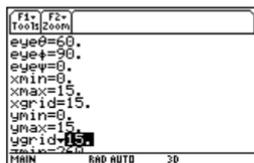
eye= [60,90,0]

x= [0,15,15]

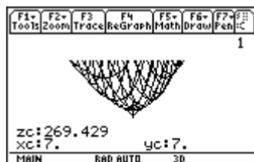
y= [0,15,15]

z= [260,300]

ncontour= [5]



4. Represente la gráfica de la función y utilice **Trace** para desplazarse hasta el punto más cercano al valor mínimo de la función superficial.

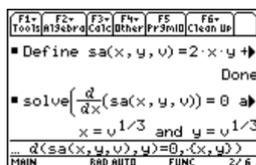


Cálculo analítico del área mínima

Realice los pasos siguientes para resolver el problema de forma analítica en la pantalla Home.

1. Halle el resultado de x e y en función de v .

Introduzca: **solve(d(sa(x,y,v),1x)=0**
(d(sa(x,y,v),y)=0, {x,y}

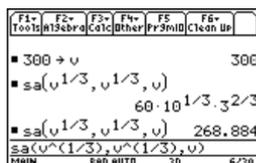


2. Halle el área mínima cuando el valor de v es igual a 300.

Introduzca: **300**→**v**

Introduzca: **sa(v^(1/3), v^(1/3), v)**

Nota: Pulse **ENTER** para obtener el resultado exacto en forma simbólica. Pulse **◀** **ENTER** para obtener el resultado aproximado en forma decimal.



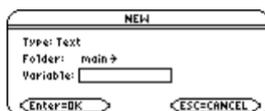
Ejecución de un documento mediante Text Editor

En esta actividad se explica cómo usar **Text Editor** para ejecutar un documento. Para obtener más información sobre las operaciones con texto, consulte *Text Editor*.

Ejecución de un documento

Realice los pasos siguientes para escribir un documento utilizando **Text Editor**, comprobar cada línea y verificar los resultados en el área de historia de la pantalla Home.

1. Abra **Text Editor** y cree una nueva variable denominada **demo1**.



Nota: Para acceder al símbolo de orden "C", utilice el menú **F2** **1:Command** de la barra de herramientas.

2. En **Text Editor**, escriba las líneas siguientes.

- : Calcule el valor máximo de f en el intervalo cerrado $[a,b]$
- : considere que f es diferenciable en $[a,b]$
- : defina $f(x)=x^3-2x^2+x-7$
- C : $1 \rightarrow a:3.22 \rightarrow b$
- C : $d(f(x),x) \rightarrow df(x)$
- C : $\text{zeros}(df(x),x)$
- C : $f(\text{ans}(1))$
- C : $f(\{a,b\})$
- C : El número más alto de las dos órdenes anteriores es el valor máximo de la función. El número más bajo es el valor mínimo.

3. Pulse **[F3]** y seleccione **1:Script view** para que **Text Editor** y la pantalla Home aparezcan en una pantalla dividida. Desplace el cursor hasta la primera línea de **Text Editor**.

4. Pulse **[F4]** repetidamente para ejecutar las líneas del documento de una en una.

Nota: Pulse **[F3]** y seleccione **2:Clear split** para que **Text Editor** vuelva a aparecer en la pantalla completa.

5. Para mostrar los resultados del documento en la pantalla completa, vaya a la pantalla Home.

Nota: Pulse **[2nd] [QUIT]** dos veces para mostrar la pantalla Home.

Descomposición de una función racional

En esta actividad se examinan los resultados obtenidos al descomponer una función racional en cociente y resto. Para obtener información detallada sobre los pasos empleados en este ejemplo, consulte *Representación gráfica básica de funciones* y *Cálculo simbólico*.

Descomposición de una función racional

Para examinar la descomposición de la función racional

$f(x) = (x^3 - 10x^2 - x + 50)/(x - 2)$ mediante una gráfica:

1. En la pantalla Home, introduzca la función racional, según se indica, y almacénela en $f(x)$.

Introduzca: $(x^3 - 10x^2 - x + 50)/(x - 2) \rightarrow f(x)$

Nota: En las pantallas del ejemplo, las entradas actuales se muestran en el modo inverso.

Calculator screen showing the input of the rational function $f(x) = \frac{x^3 - 10x^2 - x + 50}{x - 2}$. The screen displays the expression and the result $f(x)$.

2. Utilice la función de fracción propia (**propFrac**) para descomponer la función en cociente y resto.

Calculator screen showing the use of the **propFrac** function to decompose the rational function. The screen displays the expression $\text{propFrac}(f(x))$ and the result $\frac{16}{x-2} + x^2 - 8x - 17$.

3. Copie la última respuesta en la línea de entrada.
— o bien —

Introduzca: $16/(x-2) + x^2 - 8x - 17$

Nota: Desplace el cursor hasta el área de historia para resaltar la última respuesta. Pulse **ENTER** para copiarla en la línea de entrada.

Calculator screen showing the copying of the last result to the input line. The screen displays the expression $\text{propFrac}(f(x))$ and the result $\frac{16}{x-2} + x^2 - 8x - 17$.

4. Edite la última respuesta en la línea de entrada. Almacene el resto en $y1(x)$ y el cociente en $y2(x)$, según se muestra.

Introduzca:

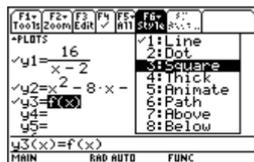
$16/(x-2) \rightarrow y1(x); x^2 - 8x - 17 \rightarrow y2(x)$

Calculator screen showing the editing of the last result to store the remainder in $y1(x)$ and the quotient in $y2(x)$. The screen displays the expression $\text{propFrac}(f(x))$ and the result $\frac{16}{x-2} + x^2 - 8x - 17$.

5. En **Y= Editor**, seleccione el estilo de representación continua para **y2(x)**.



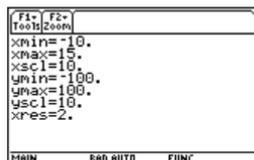
6. Añada la función inicial **f(x)** a **y3(x)** y seleccione el estilo de representación en recuadros.



7. En **Window Editor**, ajuste las variables de ventana en:

$$x = [-10, 15, 10]$$

$$y = [-100, 100, 10]$$

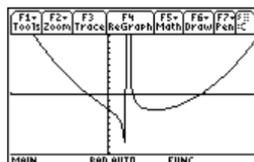
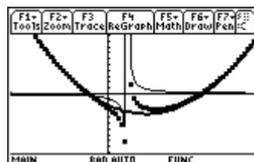


8. Dibuje la gráfica.

Nota: Verifique que el modo Graph está establecido en Function.

Observe que el comportamiento general de la función **f(x)** está basado en el cociente de segundo grado **y2(x)**. La expresión racional es, en esencia, una función de segundo grado para valores muy grandes en valor absoluto de x .

En la gráfica inferior, se representa por separado **y3(x)=f(x)** utilizando el estilo de línea.



Estudio de estadísticas: Filtrado de datos por categorías

Esta actividad proporciona un estudio estadístico del peso de los estudiantes de enseñanza secundaria obligatoria. Se han utilizado categorías para filtrar los datos.

Filtrado de datos por categorías

Cada alumno se incluye dentro de una de las ocho categorías dependiendo de su sexo y curso académico (primer curso, segundo curso, tercer curso o cuarto curso). Los datos (peso en libras) y categorías correspondientes se introducen en **Data/Matrix Editor**.

Tabla 1: Categoría frente a descripción

Categoría (C2)	Año académico y sexo
1	Alumnos de primer curso
2	Alumnas de primer curso
3	Alumnos de segundo curso
4	Alumnas de segundo curso
5	Alumnos de tercer curso
6	Alumnas de tercer curso
7	Alumnos de cuarto curso
8	Alumnas de cuarto curso

Tabla 2: C1 (peso de cada alumno en libras) frente a C2 (categoría)

C1	C2	C1	C2	C1	C2	C1	C2
110	1	115	3	130	5	145	7
125	1	135	3	145	5	160	7
105	1	110	3	140	5	165	7
120	1	130	3	145	5	170	7
140	1	150	3	165	5	190	7
85	2	90	4	100	6	110	8
80	2	95	4	105	6	115	8
90	2	85	4	115	6	125	8
80	2	100	4	110	6	120	8
95	2	95	4	120	6	125	8

Realice los pasos siguientes para comparar el peso de los estudiantes de secundaria según el curso.

1. Inicie **Data/Matrix Editor** y cree una nueva variable de datos denominada **students**.



2. Introduzca los datos y categorías de la tabla 2 en las columnas **c1** y **c2**, respectivamente.

	F1	F2	F3	F4	F5	F6	F7
	Plot Setup	Col1	Header	Col2	Col3	Plot	
DATA							
		c1	c2	c3			
4		120	1				
5		140	1				
6		85	2				
7		80	2				

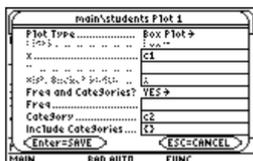
F7C2=2

3. Abra el menú **[F2] Plot Setup** de la barra de herramientas.

Nota: Defina varios gráficos de recuadros para comparar los distintos subconjuntos de datos.



4. Defina el gráfico y los parámetros de filtrado para **Plot 1**, según se muestra en la pantalla.



5. Copie **Plot 1** en **Plot 2**.



6. Repita el paso 5 y copie **Plot 1** en **Plot 3**, **Plot 4** y **Plot 5**.



7. Pulse **F1** y modifique el elemento **Include Categories** desde **Plot 2** hasta **Plot 5**, de acuerdo con lo siguiente:

Plot 2: {1,2}

(alumnos, alumnas de primer curso)

Plot 3: {7,8}

(alumnos, alumnas de cuarto curso)

Plot 4: {1,3,5,7}

(todos los alumnos)

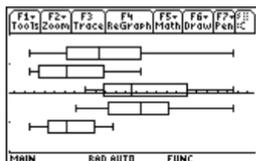
Plot 5: {2,4,6,8}

(todas las alumnas)

8. En **Y= Editor**, anule las funciones que puedan haberse seleccionado con cualquier actividad anterior.

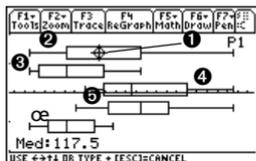
Nota: Sólo debe seleccionarse desde **Plot 1** hasta **Plot 5**.

9. Muestre los gráficos pulsando **F2** y seleccionando **9:Zoomdata**.



10. Utilice la herramienta **Trace** para comparar el peso medio de los alumnos en los distintos subconjuntos.

- 1 media, todos los alumnos
- 2 todos los alumnos
- 3 todos los de primer curso
- 4 todos los de cuarto curso
- 5 todos los alumnos
- 6 todas las alumnas



Programa del CBL 2™/CBL™ para la TI-89 Titanium / Voyage™ 200

Esta actividad incluye un programa que podrá utilizar cuando la TI-89 Titanium / Voyage™ 200 esté conectada a una unidad Calculator-Based Laboratory™ (CBL 2™). El programa permite realizar un experimento sobre la "Ley de enfriamiento de Newton". Introduzca el texto largo con el teclado del ordenador y envíelo a la calculadora por medio del software de TI Connect™. Si desea ver una lista de los programas disponibles para la unidad CBL 2™ visite el sitio Web de TI, en la dirección educaton.ti.com.

Instrucción del programa	Descripción
:cooltemp()	Nombre del programa
:Prgm	
:Local i	Establece una variable local que sólo existe durante el tiempo de ejecución.
:setMode("Graph", "FUNCTION")	Define la TI-89 Titanium / Voyage™ 200 para gráficas de funciones.
:PlotsOff	Desactiva gráficas anteriores.
:FnOff	Desactiva funciones anteriores.
:ClrDraw	Borra elementos anteriormente dibujados en las pantallas gráficas.
:ClrGraph	Borra gráficas anteriores.
:ClrIO	Limpia la pantalla Program ES (entrada/salida) de la TI-89 Titanium / Voyage™ 200.
:-10>xmin:99>xmax:10>xscl	Define las variables de ventana.
:-20>ymin:100>ymax:10>yscl	
:{0}>data	Crea y/o borra una lista denominada data.
:{0}>time	Crea y/o borra una lista denominada time.
:Send{1,0}	Envía una orden para vaciar la unidad.

Instrucción del programa	Descripción
:Send{1,2,1}	Define el canal 2 del CBL 2™/CBL™ en AutoLD para registrar una temperatura.
:Disp "Press ENTER to start"	Solicita al usuario que pulse ENTER .
:Disp "graphingTemperature."	
:Pause	Espera a que el usuario esté listo para empezar.
:PtText "TEMP(C)",2,99	Asigna una etiqueta al eje y de la gráfica.
:PtText "T(S)",80,-5	Asigna una etiqueta al eje x de la gráfica.
:Send{3,1,-1,0}	Envía la orden Trigger al CBL 2™/CBL™; recopila datos en tiempo real.
:For i,1,99	Repite las dos instrucciones siguientes para 99 lecturas de temperatura.
:Get data[i]	Obtiene una temperatura del CBL 2™/CBL™ y la almacena en una lista.
:PtOn i,data[i]	Representa los datos de temperatura en una gráfica.
:EndFor	
:seq(i,i,1,99,1)→time	Crea una lista para representar los números del muestreo time o data.
:NewPlot 1,1,time,data,,,,4	Representa time y data mediante NewPlot y la herramienta Trace.
:DispG	Presenta la gráfica.
:PtText "TEMP(C)",2,99	Vuelve a asignar una etiqueta a los ejes.
:PtText "T(S)",80,-5	
:EndPrgm	Detiene el programa.

También puede usar el Calculator-Based Ranger™ (CBR™) para estudiar las relaciones matemáticas y científicas entre distancia, velocidad y tiempo mediante la utilización de los datos recogidos al realizar una serie de actividades.

Estudio de la trayectoria de una pelota de béisbol

Esta actividad utiliza la división de pantalla para mostrar de forma simultánea una gráfica en paramétricas y la tabla correspondiente a fin de estudiar la trayectoria de una pelota de béisbol.

Definición de una gráfica en paramétricas y su tabla

Realice los pasos siguientes para estudiar la trayectoria de la pelota, que parte con una velocidad inicial de 95 pies por segundo y un ángulo de 32 grados.

1. Defina los modos de la **Page 1** según se muestra en la pantalla.



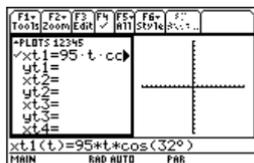
2. Defina los modos de la **Page 2** según se muestra en la pantalla.



3. En **Y= Editor**, situado a la izquierda, introduzca la ecuación **xt1(t)** de distancia de la pelota en función del tiempo **t**.

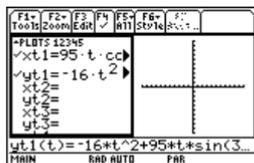
$$xt1(t)=95*t*\cos(32^\circ)$$

Nota: Pulse $\boxed{2nd} \boxed{[^\circ]}$ para obtener el símbolo de grados.



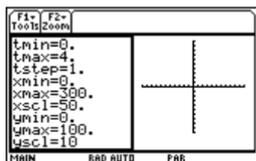
4. En **Y= Editor**, introduzca la ecuación **yt1(t)** de altura de la pelota en función del tiempo **t**.

$$yt1(t)=-16*t^2+95*t*\sin(32^\circ)$$



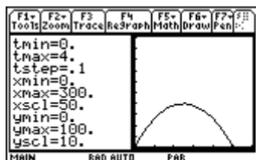
5. Ajuste las variables de ventana en:

t values= [0,4,.1]
x values= [0,300,50]
y values= [0,100,10]



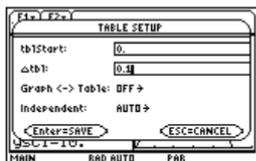
6. Cambie a la parte derecha de la pantalla y presente la gráfica.

Nota: Pulse $\boxed{2nd}$ $\boxed{[]}$.



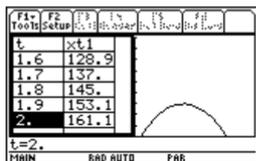
7. Muestre el cuadro de diálogo **TABLE SETUP** y cambie **tblStart** a **0** y Δ **tbl** a **0.1**.

Nota: Pulse $\boxed{\blacklozenge}$ $\boxed{[TBLSET]}$.



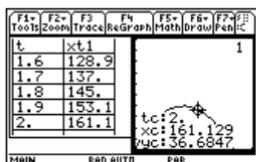
8. Presente la tabla en la parte izquierda y pulse $\boxed{\blacktriangledown}$ para resaltar $t=2$.

Nota: Pulse $\boxed{\blacklozenge}$ $\boxed{[TABLE]}$.



9. Pase a la parte derecha. Pulse $\boxed{F3}$ y desplácese a lo largo de la gráfica para mostrar los valores de **xc** y **yc** cuando $tc=2$.

Nota: Al desplazar el cursor Traza desde $tc=0.0$ hasta $tc=3.1$, verá la posición de la pelota en el tiempo tc .



Ejercicio opcional

Suponiendo que la velocidad inicial es la misma, 95 pies por segundo, halle el ángulo con que debe golpearse la pelota para alcanzar la distancia máxima.

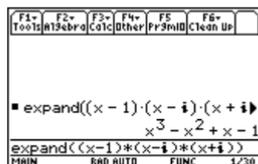
Visualización de raíces complejas de un polinomio de tercer grado

En esta actividad se describe la representación gráfica de las raíces complejas de un polinomio de tercer grado.

Visualización de raíces complejas

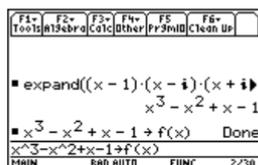
Realice los pasos siguientes para desarrollar el polinomio de tercer grado $(x-1)(x-i)(x+i)$, hallar el valor absoluto de la función, representar gráficamente la función módulo y utilizar la herramienta **Trace** para estudiar dicha función módulo.

1. En la pantalla Home, utilice la función **expand** para desarrollar la expresión $(x-1)(x-i)(x+i)$ y ver el polinomio correspondiente.



2. Copie y pegue la última respuesta en la línea de entrada y almacénela en la función **f(x)**.

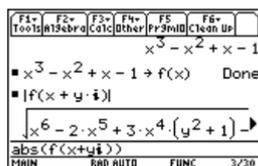
Nota: Desplace el cursor hasta el área de historia para resaltar la última respuesta y pulse **ENTER** para copiarla en la línea de entrada



3. Utilice la función **abs()** para hallar el valor absoluto de **f(x+yj)**.

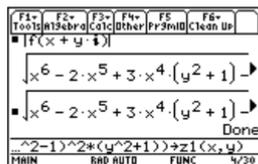
(Esta operación puede tardar cerca de 2 minutos).

Nota: El valor absoluto de la función produce un efecto visual según el cual las raíces son tangentes al eje x , en lugar de cortarlo. De la misma forma, el valor absoluto de una función de dos variables hará que las raíces sólo se apoyen en el plano xy .



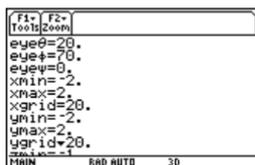
4. Copie y pegue la última respuesta en la línea de entrada y almacénela en la función **z1(x,y)**.

Nota: La gráfica **z1(x,y)** representará la función módulo.



5. Ajuste la unidad en el modo gráfico en 3D, active los ejes para el formato gráfico y ajuste las variables de ventana en:

eye= [20,70,0]
x= [-2,2,20]
y= [-2,2,20]
z= [-1,2]
ncntour= [5]

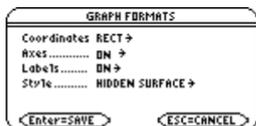


6. En **Y=Editor**, pulse:



y defina las variables de Graph Format con los valores siguientes:

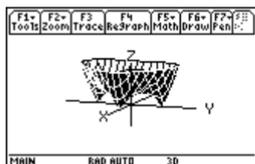
Axes= ON
Labels= ON
Style= HIDDEN SURFACE



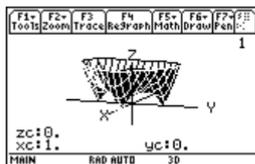
Nota: El cálculo y dibujo de las gráficas tarda aproximadamente tres minutos.

7. Represente la función módulo.

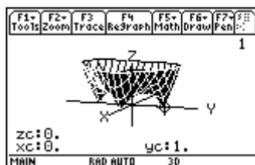
La gráfica en 3D se utiliza para mostrar la imagen de los puntos en los que la superficie se apoya en el plano xy .



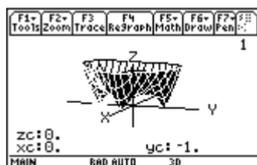
8. Utilice la herramienta Trace para examinar los valores de la función en $x=1$ e $y=0$.



9. Utilice la herramienta Trace para examinar los valores de la función en $x=0$ e $y=1$.



10. Utilice la herramienta Trace para examinar los valores de la función si $x=0$ e $y=-1$.



Resumen

Observe que el valor de z_c es cero para los valores de la función en los pasos 7 a 9. Por tanto, las raíces complejas $1, -i, i$ del polinomio $x^3 - x^2 + x - 1$ aparecen en los tres puntos en los que la gráfica de la función módulo se apoya en el plano xy .

Solución de un problema de interés compuesto

Esta actividad puede utilizarse para hallar la tasa de interés, el capital principal, el número de períodos de liquidación y el capital final de una renta anual.

Cálculo del tipo de interés de una renta anual

Realice los pasos siguientes para hallar el tipo de interés (i) de una renta anual en la que el capital inicial (p) es \$1,000, el número de períodos de liquidación (n) es 6 y el capital final (s) es \$2,000.

1. En la pantalla Home, introduzca la ecuación para resolver p .

2. Introduzca la ecuación para resolver n .

3. Introduzca la ecuación para resolver i utilizando el operador "with".

solve(s=p*(1+i)^n,i) | s=2000 y p=1000 y n=6

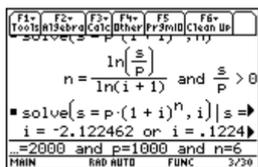
Resultado: El tipo de interés es 12.246%.

Nota:

- Para introducir el operador "with" (|):



- Pulse **ENTER** para obtener un resultado de coma flotante.



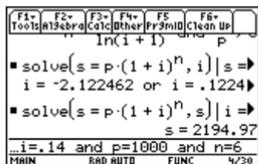
Cálculo del capital final

Para hallar el capital final, utilice los valores del ejemplo anterior, para un tipo de interés del 14%.

Introduzca la ecuación para resolver s .

solve(s=p*(1+i)^n,s) | i=.14 y p=1000 y n=6

Resultado: El capital final a un interés del 14% es \$2,194.97.



Cálculo de amortizaciones

Esta actividad crea una función que puede utilizarse para calcular el coste de un vehículo comprado a plazos. Para obtener información detallada sobre los pasos empleados en este ejemplo, consulte *Programación*, disponible en el sitio Web de TI en education.ti.com y en el CD incluido en el paquete.

Función Time-Value-of- Money

En Program Editor, defina la función Time-Value-of-Money (**tvm**) donde **temp1** = número de cuotas, **temp2** = tipo de interés anual, **temp3** = valor actual, **temp4** = cuota mensual, **temp5** = capital final y **temp6** = principio o fin del periodo de pago (1 = principio de mes, 0 = fin de mes).

```
:tvm(temp1,temp2,temp3,temp4,temp5,temp6)
:Func
:Local temp1,tempfunc,tempstr1
:-temp3+(1+temp2/1200*temp6)*temp4*((1-(1+temp2/1200)^
(-temp1))/(temp2/1200))-temp5*(1+temp2/1200)^(-temp1)
  >tempfunc
:For tempi,1,5,1
:"temp" &exact(string(tempi))>tempstr1
:If when(#tempstr1=0,false,false,true) Then
:If tempi=2
:Return approx(nsolve(tempfunc=0,#tempstr1) | #tempstr1>0
  and #tempstr1<100)
:Return approx(nsolve(tempfunc=0,#tempstr1))
:EndIf
:EndFor
:Return "parameter error"
:EndFunc
```

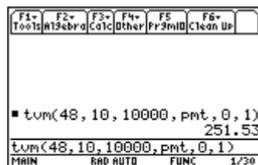
Nota: Puede usar el teclado de su ordenador para escribir textos largos y después emplear el software de TI Connect™ para enviarlos a la TI-89 Titanium / Voyage™ 200.

Cálculo de la cuota mensual

Calcule la cuota mensual a pagar por la adquisición de un vehículo de \$10,000 si desea realizar 48 pagos a un interés anual del 10%.

En la pantalla Home, introduzca los valores **tvm** para hallar *pmt*.

Resultado: La cuota mensual es 251.53.



Cálculo del número de cuotas

Calcule el número de cuotas necesarias para liquidar el préstamo del vehículo si realiza pagos mensuales de \$300.

En la pantalla Home, introduzca los valores **tvM** para hallar n .

Resultado: El número de cuotas es 38.8308.

F1+	F2+	F3+	F4+	F5	F6+	
Tools	R13eBr9	ColC	Other	Pr3mID	Clean Up	
■ tvM(n, 10, 10000, 300, 0, 1)						
38.8308						
tvM(n, 10, 10000, 300, 0, 1)						
MAIN	END AUTO	FUNC	2/30			

Cálculo de factores racionales, reales y complejos

Esta actividad muestra cómo hallar los factores racionales, reales y complejos de expresiones. Para obtener información detallada sobre los pasos utilizados en este ejemplo, consulte *Cálculo simbólico*.

Cálculo de factores

Introduzca en la pantalla Home las expresiones abajo indicadas.

1. **factor(x^3-5x)** **[ENTER]** presenta un resultado racional.

F1+	F2+	F3+	F4+	F5	F6+	
Tools	R13eBr9	ColC	Other	Pr3mID	Clean Up	
■ factor($x^3-5 \cdot x$)						
$x \cdot (x^2-5)$						
factor(x^3-5x)						
MAIN	END AUTO	FUNC	1/30			

2. **factor(x^3+5x)** **[ENTER]** presenta un resultado racional.

F1+	F2+	F3+	F4+	F5	F6+	
Tools	R13eBr9	ColC	Other	Pr3mID	Clean Up	
■ factor($x^3+5 \cdot x$)						
$x \cdot (x^2+5)$						
factor(x^3+5x)						
MAIN	END AUTO	FUNC	1/30			

3. **factor(x^3-5x, x)** **[ENTER]** presenta un resultado real.

F1+	F2+	F3+	F4+	F5	F6+	
Tools	R13eBr9	ColC	Other	Pr3mID	Clean Up	
■ factor($x^3-5 \cdot x, x$)						
$x \cdot (x + \sqrt{5}) \cdot (x - \sqrt{5})$						
factor(x^3-5x, x)						
MAIN	END AUTO	FUNC	1/30			

4. **cfactor(x^3+5x, x)** **[ENTER]** presenta un resultado complejo.

F1+	F2+	F3+	F4+	F5	F6+	
Tools	R13eBr9	ColC	Other	Pr3mID	Clean Up	
■ cFactor($x^3+5 \cdot x, x$)						
$x \cdot (x + \sqrt{5} \cdot i) \cdot (x - \sqrt{5} \cdot i)$						
cfactor(x^3+5x, x)						
MAIN	END AUTO	FUNC	1/30			

Simulación de una extracción sin reemplazamiento

Esta actividad simula el resultado obtenido al extraer, sin reemplazamiento, bolas de diferentes colores de una urna. Para obtener información detallada sobre los pasos utilizados en este ejemplo, consulte *Programación*.

Función de extracción sin reemplazamiento

En **Program Editor**, defina **drawball()** como una función que depende de dos parámetros. El primer parámetro es una lista en la que cada elemento corresponde al número de bolas de un determinado color. El segundo parámetro es el número de bolas que se va a seleccionar. La función devuelve una lista en la que cada elemento corresponde al número de bolas seleccionadas de cada color.

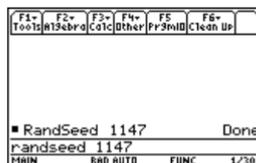
```
:drawball(urnlist,drawnum)      :For j,1,colordim,1
:Func                          :cumSum(templist)→urncum
:Local templist,drawlist,colordim, :If pick ≤ urncum[j] Then
  numballs,i,pick,urncum,j      :drawlist[j]+1→drawlist[j]
:If drawnum>sum(urnlist)       :templist[j]-1→templist[j]
:Return "too few balls"        :Exit
:dim(urnlist)→colordim        :EndIf
:urnlist→templist             :EndFor
:newlist(colordim)→drawlist    :EndFor
:For i,1,drawnum,1             :Return drawlist
:sum(templist)→numballs       :EndFunc
:rand(numballs)→pick
```

(continúa en la columna siguiente)

Extracción sin reemplazamiento

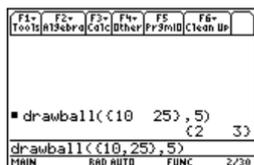
Supongamos que una urna contiene n_1 bolas de un color, n_2 bolas de un segundo color, n_3 bolas de un tercer color, etc. Simulemos que efectuamos una extracción sin reemplazamiento.

1. Introduzca un número aleatorio utilizando el orden **RandSeed**.



2. Suponiendo que la urna contiene 10 bolas rojas y 25 blancas, simule la retirada de 5 bolas de la urna escogidas al azar, sin reponerlas. Introduzca **drawball({10,25},5)**.

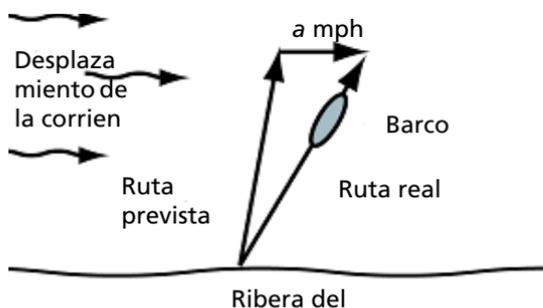
Resultado: 2 bolas rojas y 3 blancas.



Uso de vectores para determinar la velocidad

Un pequeño barco pesquero zarpa de la ribera sur del río Allegheny y navega describiendo un ángulo de 80° a una velocidad de 20 nudos. Sin embargo, la fuerza hacia el este de la corriente desplaza la trayectoria del barco y su ángulo real en relación al margen del río es de 60° .

¿Cuál es la velocidad de la corriente y cuál es la velocidad real a la que navega el barco?



1. Defina los modos de la **Page 1** como se muestra en la pantalla. (Los ángulos expresados en grados y todos los dígitos con coma decimal flotante.)

Pulse: **[MODE]** \downarrow \downarrow \downarrow . En la opción Angle, seleccione **2:DEGREE**. En la opción Display Digits, seleccione **E:FLOAT**.



2. Defina los modos de la **Page 2** como se muestra en la pantalla. (Las respuestas aparecen en formato decimal.)

Pulse: **[MODE]** **[F2]** \downarrow \downarrow . En la opción Exact/Approx, seleccione **3:APPROXIMATE**.



3. Introduzca los vectores que describen la ruta inicial del barco, la corriente del agua y la ruta resultante del barco.

Guarde estos vectores como **i**, **c** y **r**. Utilice el valor **a** para la velocidad conocida de la corriente. Utilice el valor **b** para la velocidad del barco.

Introduzca:

[20,80°]→i

[a,0°]→c

[b,60°]→r

F1	F2	F3	F4	F5	F6	
Tools	134brj	Calc	Other	Pr3mID	Clean Up	
■ [20 80°] → i [20. 80.] ■ [a 0°] → c [a 0.] ■ [b 60°] → r [b 60.] [b,60°]→r						
MAIN					DEG APPR DR FUNC	3/20

Los vectores se suelen escribir tanto en forma polar como rectangular, por lo que resulta útil convertir los vectores polares a forma rectangular.

4. Defina la función **p2r**.

Introduzca: **Define**

p2r(x)=[x[1,1]*cos(x[1,2]),
x[1,1]*sin(x[1,2])]

F1	F2	F3	F4	F5	F6	
Tools	134brj	Calc	Other	Pr3mID	Clean Up	
■ Define p2r(x)=[x[1,1].co Done ...2] , x[1,1]*sin(x[1,2])						
MAIN					DEG APPR DR FUNC	1/20

Al convertir los vectores a forma rectangular, la suma de los vectores **i** y **c** es igual al vector resultante **r**.

5. Utilizando la función **p2r**, convierta los vectores **i**, **c** y **r** a forma rectangular.

Introduzca:

p2r(i)→i

p2r(c)→c

p2r(r)→r

F1	F2	F3	F4	F5	F6	
Tools	134brj	Calc	Other	Pr3mID	Clean Up	
■ p2r(i) → i [3.47296355334 19.696155]						
■ p2r(c) → c [a 0.]						
■ p2r(r) → r [.5 b .866025403784 b]						
p2r(r)→r						
MAIN					DEG APPR DR FUNC	3/20

Dado que los vectores son iguales, la coordenada x de **i+c** debe ser igual que la coordenada x del vector resultante **r**. Del mismo modo que la coordenada y de **i+c** deberá ser igual que la coordenada y del vector resultante **r**.

6. Defina dos ecuaciones en las que intervengan los vectores $\mathbf{i}+\mathbf{c}$ y \mathbf{r} .

- La ecuación 1 define que las coordenadas x sean iguales entre sí.
- La ecuación 2 define que las coordenadas y son iguales.

Guarde estas ecuaciones como **eq1** y **eq2**, respectivamente. Introduzca:

$$\mathbf{i}[1,1]+\mathbf{c}[1,1]=\mathbf{r}[1,1]\rightarrow\mathbf{eq1}$$

$$\mathbf{i}[1,2]+\mathbf{c}[1,2]=\mathbf{r}[1,2]\rightarrow\mathbf{eq2}$$

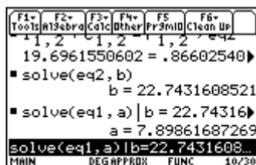
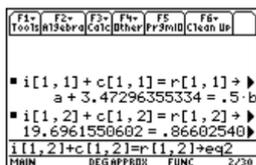
7. Resuelva **eq2** para **b** y calcule la velocidad real del barco.

solve(eq2,b)

8. Sustituya el valor conocido de **b** en la ecuación **eq1**, y resuelva **eq1** para **a** a fin de determinar el valor de **a**, la velocidad de la corriente que se desplaza hacia el este.

solve(eq1,a) | b

El barco navega a una velocidad de 22,7 nudos; la corriente del agua es de 7,9 nudos, aproximadamente.



Conectividad

Conexión de dos unidades

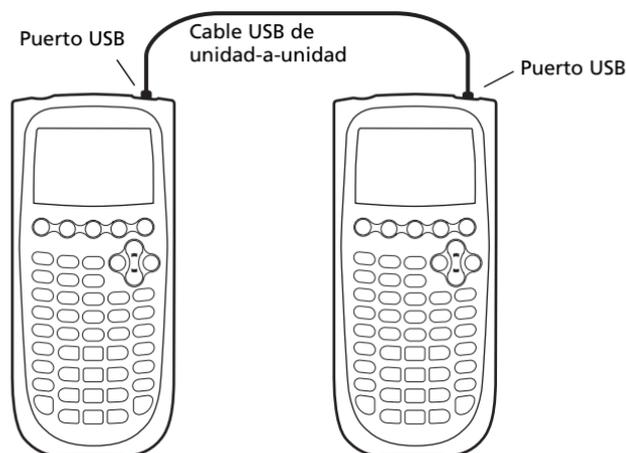
Tanto la TI-89 Titanium como la Voyage™ 200 calculadora gráfica disponen de un cable que permite conectar dos unidades y transmitir información entre ambas. El Cable USB de unidad-a-unidad viene incluido en la TI-89 Titanium; utilice este cable con el puerto USB del dispositivo. El cable estándar de unidad-a-unidad viene incluido en la Voyage™ 200; utilice este cable con el puerto E/S del dispositivo.

Nota: La TI-89 Titanium cuenta con los dos tipos de puertos, USB y E/S, lo que hace posible conectar la calculadora gráfica utilizando cualquier puerto de enlace. No obstante, para poder conectar la calculadora a un ordenador mediante el puerto E/S es necesario disponer de un cable estándar de unidad-a-unidad (en venta por separado) o de un TI Connectivity Cable USB (también en venta por separado).

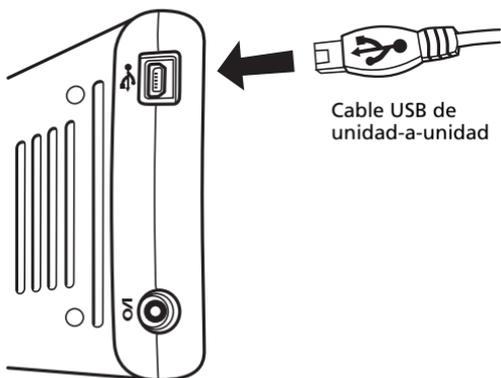
Conexión previa al envío o recepción

Presionando con firmeza, inserte un extremo del cable en el puerto de enlace de cada unidad. Las unidades pueden emitir o recibir datos, dependiendo de cómo se hayan configurado desde la pantalla **VAR-LINK**.

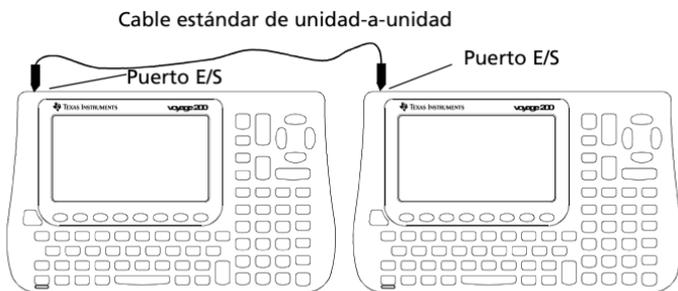
Puede enlazar una TI-89 Titanium o Voyage™ 200 a otra TI-89 Titanium, Voyage™ 200, TI-89 o TI-92 Plus.



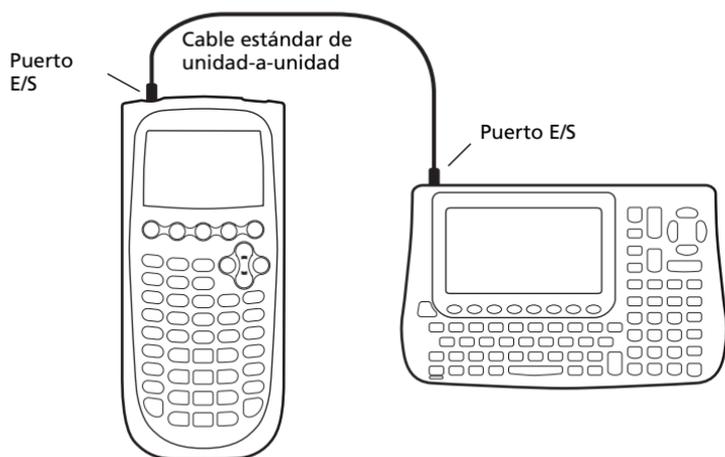
Dos calculadoras TI-89 Titanium conectadas entre sí



Coloque las calculadoras con los puertos USB de ambas encarados, e inserte el conector.



Dos calculadoras Voyage™ 200 conectadas entre sí



Una calculadora TI-89 Titanium y una Voyage™ 200 conectadas entre sí



Una calculadora TI-89 Titanium y una TI-89 conectadas entre sí

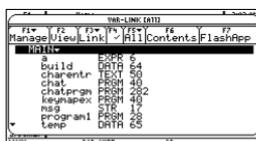
Transmisión de variables, aplicaciones Flash y carpetas

La transmisión de variables es un medio eficaz para compartir cualquier variable, función, programa, etc. incluidos en la lista de la pantalla **VAR-LINK**, además de aplicaciones (Apps) Flash y carpetas.

Preparación de las unidades

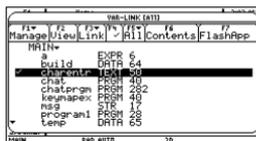
Las aplicaciones Flash sólo se pueden transmitir entre determinados tipos de unidades. Por ejemplo, puede transferir una aplicación (App) de una TI-89 Titanium a otra TI-89 Titanium, o desde una TI-89 Titanium a una TI-89. De igual manera se puede transferir una aplicación desde una Voyage™ 200 a otra Voyage™ 200, o desde una Voyage™ 200 a una TI-92 Plus.

1. Conecte dos calculadoras gráficas por medio del cable apropiado.
2. En la unidad *emisora*, pulse **[2nd]** **[VAR-LINK]** para mostrar la pantalla **VAR-LINK**.

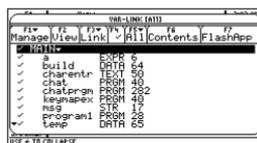


3. En la unidad *emisora*, seleccione las variables, carpetas o aplicaciones Flash que desee enviar.

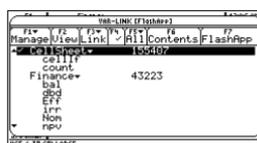
- Si quiere seleccionar sólo una variable, aplicación Flash o carpeta, lleve el cursor hasta ella para resaltarla y pulse **[F4]** para situar una marca de verificación (✓) junto a la selección.



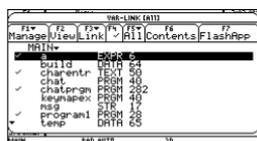
- Si está en la pantalla **VAR-LINK** predeterminada, esta acción seleccionará la carpeta y su contenido. Las carpetas contraídas se expanden al seleccionarlás.



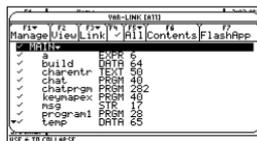
- Si elige una aplicación Flash (desde la ficha F7), se seleccionará la carpeta App y su contenido. La marca de verificación aparece junto a la carpeta, no junto al contenido de la misma. Las carpetas de aplicaciones Flash contraídas se expanden automáticamente.



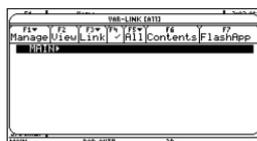
- Para seleccionar múltiples variables, aplicaciones Flash o carpetas, resalte cada una de ellas y pulse **F4** para situar una marca de verificación (✓) junto a la selección. Vuelva a pulsar **F4** para anular la selección de cualquier elemento que no desee transmitir.



- Para seleccionar todas las variables, aplicaciones Flash o carpetas, elija **F5 All 1:Select All.**



4. En la unidad *receptora*, pulse **2nd** [VAR-LINK] para mostrar la pantalla **VAR-LINK**. (La unidad emisora sigue mostrando la pantalla **VAR-LINK**.)



5. En las dos unidades, *receptora* y *emisora*, pulse **F3 Link** para mostrar las opciones de menú.



6. En la unidad *receptora*, seleccione **2:Receive**.

En La línea de estado de la unidad receptora muestra el mensaje **VAR-LINK: WAITING TO RECEIVE** junto con el indicador **BUSY**.

7. En la unidad *emisora*, seleccione **1:Send**.

De este modo se inicia la transmisión.

Durante la transmisión aparece una barra de progreso en la línea de estado de la unidad receptora. Cuando ha finalizado, la pantalla **VAR-LINK** de la unidad receptora se actualiza con el resultado de la transmisión.

Nota: Antes de transferir una App adquirida, puede ser necesario que la unidad receptora tenga el certificado correspondiente. Un certificado es un archivo generado por TI, que no es preciso para las descargas gratuitas y las aplicaciones básicas.

Reglas para la transmisión de variables, aplicaciones Flash y carpetas

Si en las unidades emisora y receptora hay variables no bloqueadas ni archivadas con el mismo nombre, prevalecerán las de la unidad emisora y se sobrescribirán a las que haya en la receptora.

Si en las unidades emisora y receptora hay variables bloqueadas con el mismo nombre, se deberán desbloquear en la unidad receptora antes de poderlas sobrescribir con las de la unidad emisora. Si las unidades emisora y receptora tienen variables archivadas con el mismo nombre, aparecerá un mensaje solicitando que confirme la sobrescritura.

Si selecciona:	El resultado:
Variable no bloqueada	La variable se transmite a la carpeta actual y permanece desbloqueada en la unidad receptora.
Variable bloqueada	La variable se transmite a la carpeta actual y permanece bloqueada en la unidad receptora.
Variable archivada	La variable se transmite a la carpeta actual y permanece archivada en la unidad receptora.
Aplicación Flash desbloqueada	Si la unidad receptora dispone de la certificación adecuada, la aplicación Flash se transmite. Permanece desbloqueada en la unidad receptora.
Aplicación Flash bloqueada	Si la unidad receptora dispone de la certificación adecuada, la aplicación Flash se transmite. Permanece bloqueada en la unidad receptora.

Si selecciona:	El resultado:
Carpeta desbloqueada	Se transmiten la carpeta y el contenido seleccionado. La carpeta permanece desbloqueada en la unidad receptora.
Carpeta bloqueada	Se transmiten la carpeta y el contenido seleccionado. La carpeta se desbloquea en la unidad receptora.

Cancelación de una transmisión

Desde la unidad emisora o la unidad receptora:

1. Pulse **[ON]**.
Aparece un mensaje de error.
2. Pulse **[ESC]** o **[ENTER]**.



Errores comunes y mensajes de notificación

Aparece en:	Mensaje y descripción:
Unidad emisora	

Aparece después de varios segundos si:

- No hay un cable conectado al puerto de enlace de la unidad emisora.
– o bien –
- No hay una unidad receptora conectada al otro extremo del cable.
– o bien –
- La unidad receptora no está configurada para recibir.

Pulse **[ESC]** o **[ENTER]** para cancelar la transmisión.

Nota: Es posible que la unidad emisora no muestre siempre este mensaje. En su lugar, se mantendrá en estado **BUSY** hasta que cancele la transmisión.

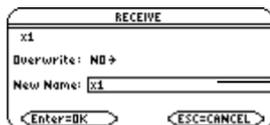
Aparece en: Mensaje y descripción:

Unidad emisora



La unidad receptora no tiene el certificado adecuado para el sistema operativo (OS) o la aplicación Flash que se dispone a enviar.

Unidad receptora

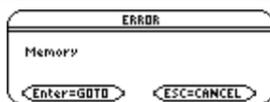


La opción New Name sólo está activa si Overwrite indica NO.

La unidad receptora tiene una variable con el mismo nombre que una determinada variable que se está enviando.

- Para sobrescribir la variable existente, pulse **[ENTER]**. (De forma predeterminada, **Overwrite = YES**.)
- Para guardar la variable con otro nombre, cambie a **Overwrite = NO**. En el cuadro de entrada **New Name**, escriba un nombre de variable que no esté en la unidad receptora. A continuación, pulse **[ENTER]** dos veces.
- Para omitir esta variable y continuar con la siguiente, defina **Overwrite = SKIP** y pulse **[ENTER]**.
- Para cancelar la transmisión, pulse **[ESC]**.

Unidad receptora



La unidad receptora carece de memoria suficiente para el envío en curso. Pulse **[ESC]** o **[ENTER]** para cancelar la transmisión.

Borrado de variables, aplicaciones Flash y carpetas

1. Pulse **[2nd] [VAR-LINK]** para mostrar la pantalla **VAR-LINK**.
2. Seleccione las variables, carpetas o aplicaciones Flash que desee borrar.

- Si quiere seleccionar sólo una variable, aplicación Flash o carpeta, lleve el cursor hasta ella para resaltarla y pulse **[F4]** para situar una marca de verificación (✓) junto a la selección.
 - Si está en la pantalla **VAR-LINK** predeterminada, esta acción seleccionará la carpeta y su contenido. Las carpetas contraídas se expanden al seleccionarlás.
 - Si elige una aplicación Flash (desde la ficha F7), se seleccionará la carpeta App y su contenido. La marca de verificación aparece junto a la carpeta, no junto al contenido de la misma. Las carpetas de aplicaciones Flash contraídas se expanden automáticamente.

Nota: No es posible borrar la carpeta **Main**.

- Para seleccionar múltiples variables, aplicaciones Flash o carpetas, resalte cada una de ellas y pulse **[F4]** para situar una marca de verificación (✓) junto a la selección. Vuelva a pulsar **[F4]** para anular la selección de cualquier elemento que no desee transmitir.
- Para seleccionar todas las variables, aplicaciones Flash o carpetas, elija **[F5] All 1:Select All**.

3. Pulse **[F1]** y elija **1>Delete**.

– o bien –

Pulse **[←]**. Aparecerá un mensaje de confirmación.

4. Pulse **[ENTER]** para confirmar la supresión.

Dónde conseguir aplicaciones (Apps) Flash

Para obtener la información más actualizada sobre aplicaciones Flash para variables, acceda al sitio Web de Texas Instruments, en la dirección education.ti.com o póngase en contacto con Texas Instruments en TI-Cares™.

Hay muchas aplicaciones para las que ya no es necesario el certificado. Si intenta transferir una App de una unidad a otra y aparece un mensaje **Unlicensed OS or Flash application (Aplicación Flash o SO sin licencia)**, vuelva a repetir el intento de descarga de la aplicación desde el sitio Web de Texas Instruments, en la dirección education.ti.com.

Puede descargar una aplicación Flash y/o un certificado del sitio Web de Texas Instruments a un ordenador y utilizar un USB cable o un TI Connectivity Cable USB para instalar la aplicación o el certificado en la TI-89 Titanium / Voyage™ 200 calculadora gráfica.

Para obtener instrucciones de instalación para aplicaciones Flash, consulte education.ti.com/guides.

Transmisión de variables con control de programas

Para transmitir una variable de un dispositivo a otro, puede utilizar un programa que contenga las funciones **GetCalc** y **SendCalc**.

SendCalc envía una variable al puerto de enlace, en el que hay un dispositivo conectado y preparado para recibirla. El dispositivo conectado debe mostrar la pantalla Home (Principal) o estar ejecutando la función **GetCalc** desde un programa.

Si lo desea, puede utilizar parámetros opcionales con la función **SendCalc** o **GetCalc** para especificar el puerto correspondiente, USB o E/S. (Consulte el Apéndice A para obtener más detalles.) Si no incluye estos parámetros, la TI-89 Titanium utilizará el puerto USB para la comunicación.

El programa “Chat”

El programa siguiente utiliza las funciones **GetCalc** y **SendCalc**. El programa establece dos bucles que permiten que los dispositivos conectados se turnen en el envío y la recepción/visualización de una variable llamada **msg**. **InputStr** permite a cada usuario escribir un mensaje en la variable **msg**.

```

:Chat()
:Prgm
:ClrIO
:Disp "On first unit to send,", "
  enter 1;","On first to receive,"
:InputStr " enter 0",msg
:If msg="0" Then
:  While true
① [ :   GetCalc msg
    [ :   Disp msg
② [ :   InputStr msg
    [ :   SendCalc msg
    :   EndWhile
    :Else
④ [ :   InputStr msg
    [ :   SendCalc msg
⑤ [ :   GetCalc msg
    [ :   Disp msg
    :   EndWhile
:EndIf
:EndPrgm

```

Notas:

- ❶ Configura la unidad para recibir y mostrar la variable msg.
- ❷ A continuación, permite al usuario escribir un mensaje en msg y enviarlo.
- ❸ El bucle ejecutado por la unidad que recibe el primer mensaje.
- ❹ Permite a este usuario escribir un mensaje en msg y enviarlo.
- ❺ A continuación, configura esta unidad para que reciba y muestre el msg.
- ❻ El bucle ejecutado por la unidad que envía el primer mensaje.

Para sincronizar **GetCalc** y **SendCalc**, los bucles están organizados de tal forma que la unidad receptora ejecute **GetCalc** mientras la unidad emisora espera que el usuario escriba el mensaje.

Ejecución del programa

En este procedimiento se asume que:

- Los dos dispositivos están enlazados mediante el cable de conexión.
- El programa Chat está cargado en ambos dispositivos.

- Se ha utilizado Program Editor en ambos dispositivos para acceder al programa.
- o bien -
- Se ha accedido al programa en un dispositivo y luego se ha utilizado **VAR-LINK** para transmitir la variable de programa al otro.

Para ejecutar el programa en los dos dispositivos:

1. En la pantalla Home (Principal) de cada dispositivo, escriba **chat()**.
2. Cuando cada dispositivo muestre el indicativo inicial, responda como se indica a continuación.

En la:	Escriba:
Calculadora que va a enviar el primer mensaje.	1 y pulse [ENTER] .
Calculadora que va a recibir el primer mensaje.	0 y pulse [ENTER] .

3. Escriba los mensajes cuando sea su turno y pulse **[ENTER]** para enviar la variable **msg** al otro dispositivo.

Parada del programa

Dado que el programa **Chat** establece un bucle infinito en los dos dispositivos, pulse **[ON]** (en ambas calculadoras) para detener el programa. Si pulsa **[ESC]** para aceptar el mensaje de error, el programa se detendrá en la pantalla Program I/O. Pulse **[F5]** o **[ESC]** para regresar a la pantalla Home.

Actualización del sistema operativo (SO)

Puede utilizar un ordenador para actualizar el sistema operativo de la TI-89 Titanium o la Voyage™ 200. También puede transferir el sistema operativo de una unidad a otra del mismo modelo (por ejemplo, de una TI-89 Titanium a una TI-89 Titanium o de una Voyage™ 200 a una Voyage™ 200).

La instalación del software del sistema operativo restablece toda la memoria del dispositivo a los valores de configuración de fábrica. Esto significa que se borrarán todas las variables definidas por el usuario (tanto en la memoria RAM como en los archivos de datos del usuario), funciones, programas, listas y carpetas (excepto la carpeta principal, Main). También es posible que se borren algunas aplicaciones Flash. Antes de llevar a cabo la actualización del sistema operativo consulte la información importante relacionada con las pilas.

Información importante relacionada con la descarga del sistema operativo

Se aconseja instalar pilas nuevas antes de proceder con la descarga de un sistema operativo.

Si está trabajando con la TI-89 Titanium en un idioma que no sea inglés y se dispone a actualizar el software del sistema operativo, asegúrese de que tiene instalada la aplicación de traducción más reciente. Tenga en cuenta que, de lo contrario, la pantalla no mostrará correctamente las solicitudes del programa, los mensajes de error ni cualquier otra información relacionada con la nueva funcionalidad del sistema operativo.

Durante el modo de descarga de SO, no está disponible la función de apagado automático, Automatic Power Down™ (APD™). Si deja el dispositivo en modo de descarga durante más tiempo del necesario antes de comenzar con el proceso se agotarán las pilas. Por tanto será necesario cambiar las pilas gastadas por otras nuevas antes de comenzar la descarga.

Si la transferencia se interrumpe de forma accidental antes de finalizar completamente, deberá volver a instalar el sistema operativo. De nuevo, recuerde que debe instalar pilas nuevas antes de descargar.

Póngase en contacto con Texas Instruments en TI-Cares™ si observa algún problema.

Copia de seguridad de la unidad previa a la instalación del sistema operativo

Al instalar la actualización del sistema operativo, recuerde que el proceso de instalación:

- Borra todas las variables (tanto en la memoria RAM como en el archivo de datos del usuario), funciones, programas y carpetas definidas por el usuario .
- El borrado podría afectar también a las aplicaciones Flash.
- Restablece todas las variables y modos del sistema a los valores de configuración definidos en fábrica. El resultado es equivalente a utilizar la pantalla **MEMORY** para restablecer toda la memoria del sistema.

Si desea conservar algunas variables o aplicaciones Flash existentes, haga lo siguiente antes de comenzar a instalar la actualización:

- **Importante:** Instale pilas nuevas.
- Transmita las variables o las aplicaciones Flash a otro dispositivo.
– o bien –

- Utilice un USB cable o TI Connectivity Cable USB y el software de TI Connect™ (education.ti.com/downloadticonnect) para enviar las variables y/o las aplicaciones Flash a un ordenador.

Dónde conseguir actualizaciones del sistema operativo

Para obtener la información más reciente sobre actualizaciones del sistema operativo, acceda al sitio Web de Texas Instruments, en la dirección education.ti.com o póngase en contacto con Texas Instruments en TI-Cares™.

Puede descargar una actualización del sistema operativo o aplicación Flash del sitio Web de Texas Instruments en un ordenador y utilizar un USB cable o un TI Connectivity Cable USB para instalar la aplicación o el certificado en la TI-89 Titanium / Voyage™ 200.

Para obtener información más detallada, consulte las instrucciones en la Web.

Transferencia del sistema operativo

El software del sistema operativo sólo se puede transferir de una TI-89 Titanium a una TI-89 Titanium, de una TI-89 a una TI-89, de una Voyage™ 200 a una Voyage™ 200 o desde una TI-92 Plus a otra TI-92 Plus.

Para transferir el sistema operativo (SO) de unidad a unidad:

1. Conecte entre sí dos unidades idénticas, por ejemplo, una TI-89 Titanium con una TI-89 Titanium; o una Voyage™ 200 con una Voyage™ 200.
2. En las unidades receptora y emisora, pulse **[2nd] [VAR-LINK]** para mostrar la pantalla **VAR-LINK**.
3. En las dos unidades, receptora y emisora, pulse **[F3] Link** para mostrar las opciones de menú.
4. En la unidad receptora, seleccione **5:Receive OS**.

Aparecerá un mensaje de advertencia. Pulse **[ESC]** para detener el proceso, o **[ENTER]** para continuar. Al pulsar **[ENTER]** aparece **VAR-LINK: WAITING TO RECEIVE** junto con el indicador **BUSY** en la línea de estado de la unidad receptora.

5. En la unidad emisora, seleccione **4:Send OS**.

Aparecerá un mensaje de advertencia. Pulse **[ESC]** para detener el proceso, o **[ENTER]** para iniciar la transmisión.

Importante:

- En cada unidad receptora, no olvide hacer copia de seguridad de la información, si es necesario, e instalar pilas nuevas.
- Asegúrese de que las dos unidades, receptora y emisora, muestren la pantalla **VAR-LINK**.

Durante la transferencia, la unidad receptora muestra el progreso de la operación. Una vez finalizada la transferencia:

- La unidad emisora regresa a la pantalla **VAR-LINK**.
- La unidad receptora regresa al escritorio de Apps o a la pantalla Home (Principal). Es posible que necesite utilizar   (aclamar) o   (oscurecer) para ajustar el contraste.

No intente cancelar la transferencia de un sistema operativo

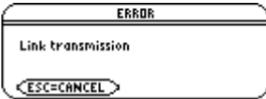
El sistema operativo de la unidad receptora se borra realmente en cuanto se inicia la transferencia. Si ésta se interrumpe antes de haber finalizado, la unidad receptora no podrá funcionar correctamente, y será necesario instalar de nuevo la actualización del sistema operativo.

Si va a actualizar el sistema operativo de varias unidades

Para actualizar el sistema operativo de varias unidades, descargue e instale el SO en cada unidad, y transfiera luego la actualización del sistema operativo de una unidad a otra. Este método resulta más rápido que instalarlo en cada unidad por medio del ordenador. Las actualizaciones del sistema operativo se distribuyen de forma gratuita y no precisan de ningún certificado para su descarga ni instalación.

Mensajes de error

La mayoría de los mensajes de error aparecen en la unidad emisora. Dependiendo del momento del proceso de transferencia en el que se ha producido el error puede aparecer un mensaje en la pantalla de la unidad receptora.

Mensaje de error	Descripción
	Las unidades emisora y receptora no están conectadas correctamente, o la unidad receptora no está preparada para recibir.

Mensaje de error**Descripción**

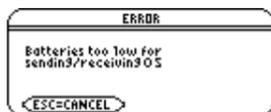


El certificado de la unidad receptora no es válido para el sistema operativo (SO) de la unidad emisora. Debe obtener e instalar un certificado válido.

Si la aplicación ya no requiere el uso de un certificado, puede descargarla de nuevo desde el sitio Web de Texas Instruments, en la dirección education.ti.com, y volver a instalar la aplicación en la calculadora.



Se ha producido un error durante la transferencia. El sistema operativo de la unidad receptora está dañado. Debe reinstalar el software del producto desde un ordenador.



Cambie las pilas de la unidad que muestre este mensaje.

Recopilación y transmisión de listas ID

La pantalla **VAR-LINK** cuenta con la opción de menú **[F3] 6:Send ID List**, que permite recuperar números de identificación (ID) electrónicos de un dispositivo TI-89 Titanium, TI-89, Voyage™ 200 calculadora gráfica o TI-92 Plus.

Listas de ID y certificados de grupo

La función de listas ID es una forma cómoda de recopilar números de identificación para la compra en grupo de aplicaciones comerciales. Una vez obtenidas las listas de ID puede enviarlas a Texas Instruments para conseguir un certificado de grupo; es decir, un certificado que autoriza a distribuir el software adquirido en varias unidades TI-89 Titanium, TI-89, Voyage™ 200 o TI-92 Plus. El software indicado en los certificados de grupo puede cargarse, borrarse y cargarse de nuevo en los dispositivos con la frecuencia necesaria. Puede añadir al certificado de grupo nuevos números ID y/o nuevas aplicaciones comerciales.

Recopilación de listas de ID

Para recopilar los números de identificación, puede utilizar una calculadora y agrupar en ella todos los números, o utilizar varias calculadoras y consolidar las distintas listas de ID en una unidad.

Para enviar un número ID de un dispositivo a otro, debe conectar primero las dos unidades entre sí por medio de un Cable USB de unidad-a-unidad o un Cable estándar de unidad-a-unidad.

Paso:	En la:	Haga lo siguiente:
1.	Unidad recolectora (Unidad receptora)	Muestre la pantalla Home (Principal). Pulse:    
2.	Unidad emisora	a. Pulse [2nd] [VAR-LINK] para mostrar la pantalla VAR-LINK . b. Pulse [F3] Link y seleccione 6:Send ID List .
		
		La unidad emisora envía una copia de su número de identificación único a la lista de ID de la unidad recolectora. La unidad emisora siempre conserva su propio número de ID, que no se puede borrar del dispositivo.
3.	Otras unidades	Repita los pasos 1 y 2 hasta que todas las listas de ID estén en un dispositivo. Dependiendo de la memoria disponible en el dispositivo recolector, es posible recopilar hasta un máximo de 4.000 identificaciones.

Notas:

- No es posible ver la lista de ID de las unidades emisoras ni de la recolectora.
- La lista de ID que se envía con éxito de un dispositivo a otro se borra automáticamente de la unidad emisora.
- Si una unidad recoge dos veces el mismo identificador, el duplicado se borra automáticamente de la lista.

Borrado de la lista de ID

La lista de identificadores permanece en el dispositivo recolector una vez descargada en el ordenador. Puede utilizar el dispositivo para cargar la lista en otros ordenadores.

Para borrar la lista de identificadores de la unidad recolectora:

1. Pulse **[2nd] [VAR-LINK]** para mostrar la pantalla **VAR-LINK**.
2. Pulse **[F1] Manage** y seleccione **A:Clear ID List**.



Compatibilidad entre TI-89 Titanium, Voyage™ 200, TI-89 y TI-92 Plus

En términos generales tanto los datos como los programas de las calculadoras TI-89 Titanium, TI-89, Voyage™ 200 y TI-92 Plus son compatibles entre sí, con algunas excepciones.

La mayoría de las funciones de la TI-89 Titanium son compatibles con las de TI-89, Voyage™ 200 y TI-92 Plus. La TI-89 Titanium y la TI-89 son similares, salvo que la TI-89 Titanium tiene más memoria (más espacio para Apps y archivo del usuario) y la TI-89 Titanium dispone de un puerto USB. La Voyage™ 200 es igual que la TI-92 Plus excepto en que tiene más memoria y, por lo tanto, más espacio para aplicaciones (Apps).

Los datos son compatibles entre los modelos TI-89 Titanium, TI-89, Voyage™ 200 y TI-92 Plus, si bien algunos programas escritos para una calculadora determinada pueden no funcionar en otra a causa de las diferencias en el tamaño de las pantallas, los teclados o el puerto USB de la TI-89 Titanium.

Existen otras incompatibilidades debidas a la diferente versión del sistema operativo de cada calculadora. Para descargar la versión más reciente del sistema operativo, visite el sitio Web de Texas Instruments, en la dirección education.ti.com/downloadticonnect.

Tabla de enlaces para transmisión

Hacia → Desde ↓	TI-89 Titanium	TI-89	Voyage™ 200	TI-92 Plus
TI-89 Titanium	SO Apps Variables	Apps Variables	Variables	Variables
TI-89	Apps Variables	SO Apps Variables	Variables	Variables
Voyage™ 200	Variables	Variables	SO Apps Variables	Apps Variables
TI-92 Plus	Variables	Variables	Apps Variables	SO Apps Variables

Gestión de memoria y variables

Comprobación y reinicio de la memoria

La pantalla **MEMORY** muestra la cantidad de memoria (en bytes) que utiliza cada tipo de variable, y la cantidad de memoria libre. Además, esta pantalla puede utilizarse para restaurar la memoria.

Presentación de la pantalla **MEMORY**

Pulse [2nd] [MEM]. La pantalla siguiente pertenece a una Voyage™ 200 calculadora gráfica. (Los números de su pantalla **MEMORY** pueden ser distintos de los de la ilustración.)

MEMORY			
[F1] RESET			
Expr	61	Text	74
List	80	DOB	0
Matrix	238	Data	0
Function	0	Other	154
Prgm/Asm	269	History	1872
Picture	2241	Systen	126862
String	0	FlashApp	894319
		Archive	219
		RAM free	131312
		Flash ROM free	1923452

Prgm/Asn: Incluye programas escritos para la TI-89 Titanium / Voyage™ 200 así como cualquier programa de lenguaje ensamblador que haya cargado.

History: Tamaño de los pares guardados en el área de historia de la pantalla Home.

FlashApp: Tamaño de las aplicaciones Flash.

RAM free: Espacio libre en la Flash RAM.

Flash ROM free: Espacio libre en la Flash ROM.

Nota: Use la pantalla **VAR-LINK** para visualizar el tamaño de las variables individuales y determinar si pertenecen al archivo de datos del usuario.

Para cerrar la pantalla, pulse [ENTER]. Para restaurar la memoria, lleve a cabo el procedimiento siguiente.

Reinicio de la memoria

En la pantalla **MEMORY**:

1. Pulse **[F1]**.
2. Seleccione el elemento correspondiente.



Elemento	Descripción
RAM	1:All RAM: Al reiniciarse se borran todos los datos y programas de la RAM. 2:Default: Restablece los ajustes de fábrica de todos los modos y las variables del sistema. No afecta a las funciones, carpetas y variables definidas por el usuario.
Flash ROM	1:Archive: Al reiniciarse, se borran todos los datos y programas de la Flash ROM. 2:Flash Apps: Al reiniciarse, se borran todas las aplicaciones Flash de la Flash ROM. 3:Both: Al reiniciarse, se borran todos los datos, programas y aplicaciones Flash de la Flash ROM
All Memory	Al reiniciarse, se borran todos los datos, programas y aplicaciones Flash de la RAM y la Flash ROM.

Importante: Para eliminar variables individuales (en lugar de todas), utilice **VAR-LINK**.

3. Cuando se le pida una confirmación, pulse **[ENTER]**.

La TI-89 Titanium / Voyage™ 200 presenta un mensaje al completarse el reinicio.

Nota: Para cancelar el reinicio, pulse **[ESC]** en vez de **[ENTER]**.

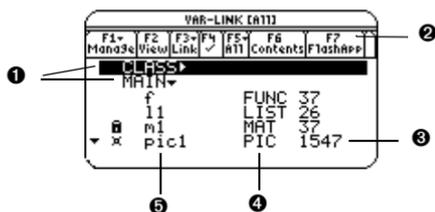
4. Pulse **[ENTER]** para confirmar el mensaje.

Presentación de la pantalla VAR-LINK

La pantalla **VAR-LINK** muestra las variables y carpetas que se encuentran definidas en cada momento. Después de presentar esta pantalla, puede manipular las variables, las carpetas o ambos elementos.

Presentación de la pantalla VAR-LINK

Pulse **[2nd] [VAR-LINK]**. De forma predeterminada, la pantalla **VAR-LINK** muestra todas las variables definidas por el usuario en todas las carpetas, junto con los tipos de datos.



- ❶ Nombres de carpetas (en orden alfabético)
- ❷ Muestra las aplicaciones Flash instaladas
- ❸ Tamaño en bytes
- ❹ Tipos de datos
- ❺ Nombres de variables (indicados por orden alfabético en cada carpeta)

Significado...

▶	Vista reducida de la carpeta (a la derecha del nombre de la carpeta).
▼	Vista ampliada de la carpeta (a la derecha del nombre de la carpeta).
▼	Indica que puede desplazarse para buscar más variables y/o carpetas (en la esquina inferior izquierda de la pantalla).
✓	Indica que se ha seleccionado con [F4].
🔒	Indica que está bloqueada
📁	Indica que está archivada.

Para desplazarse por la lista:

- Pulse ⏪ o ⏩ (utilice [2nd] ⏪ o [2nd] ⏩ para desplazarse por una página cada vez).
– o bien –
- Escriba una letra. Si hay nombres de variables que comienzan con dicha letra, el cursor se mueve para resaltar el primero de ellos.

Nota: Escriba una letra para desplazarse a los nombres que comienzan por dicha letra.

Tipos de variables indicados en VAR-LINK

Tipo	Descripción
ASM	Programa de lenguaje ensamblador

Tipo	Descripción
DATA	Datos
EXPR	Expresión (valores numéricos incluidos)
FUNC	Función
GDB	Base de datos de gráficos
LIST	Lista
MAT	Matriz
PIC	Imagen gráfica
PRGM	Programa
STR	Cadena
TEXT	Sesión de Text Editor

En la lista anterior no se incluyen diversos tipos de datos utilizados por aplicaciones de software.

Cerrado de la pantalla VAR-LINK

Para cerrar la pantalla **VAR-LINK** y volver a la aplicación actual, utilice **ENTER** o **ESC** de la forma explicada a continuación.

Pulse:	Para:
ENTER	Pegar el nombre resaltado de variable o de carpeta en la posición del cursor en la aplicación en que se encuentre.
ESC	Volver a la aplicación sin pegar el nombre resaltado.

Presentación de información sobre variables en la pantalla Home (Principal)

Desde la pantalla Home (Principal) puede mostrar información sobre variables sin necesidad de abrir la pantalla VAR-LINK.

- Para determinar si en la tabla del sistema existe una variable con un nombre dado, introduzca la función **IsVar()** en la pantalla Home.

IsVar (*nombre_de_variable*)

└─ **IsVar** es una función que debe ir encerrada entre paréntesis.

- Para determinar si una variable está archivada, utilice la función **IsArchiv()**.

IsArchiv (*nombre_de_variable*)

- Para determinar si una variable está bloqueada, utilice la función **IsLocked()**.

IsLocked (*nombre_de_variable*)

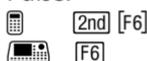
Manejo de variables y carpetas con VAR-LINK

En la pantalla **VAR-LINK** se puede mostrar el contenido de una variable. También se puede seleccionar uno o más de los elementos que aparecen en listas y trabajar con ellos de la manera que se indica en esta sección.

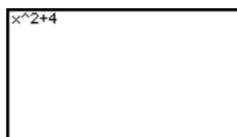
Presentación del contenido de una variable

Puede mostrar todos los tipos de variables excepto **ASM**, **DATA**, **GDB**, y variables creadas por Apps Flash. Por ejemplo, una variable **DATA** deberá abrirse en el **Data/Matrix Editor**.

1. En **VAR-LINK**, mueva el cursor para resaltar la variable.
2. Pulse:



Si resalta una carpeta, la pantalla muestra el número de variables en la misma.



3. Para volver a **VAR-LINK**, pulse cualquier tecla.

Nota: No es posible editar el contenido en esta pantalla.

Selección de elementos de una lista

Para realizar otras operaciones, seleccione una o más variables y/o carpetas.

Para seleccionar:	Realice lo siguiente:
Una sola variable o carpeta	Mueva el cursor para resaltar el elemento, a continuación pulse [F4] .
Un grupo de variables o carpetas	Resalte cada elemento y pulse [F4] . Se presenta una marca ✓ a la izquierda de cada elemento seleccionado. Si selecciona una carpeta, también selecciona todas las variables dentro de la misma. Utilice [F4] para seleccionar o anular la selección de elementos.

Para seleccionar:

Todas las carpetas y variables

**Realice lo siguiente:**

Pulse \leftarrow para ampliar la carpeta, después pulse $\boxed{F5}$ **All** y seleccione **1:Select All**.

Al elegir **3:Select Current** se selecciona el último conjunto de elementos transmitidos a la unidad durante la sesión **VAR-LINK** actual.

Al elegir **4:Expand All** o **5:Collapse All** se expanden o contraen las carpetas o las aplicaciones Flash.

Nota: Pulse \leftarrow o \rightarrow para alternar entre la vista contraída o ampliada de una carpeta cuando está resaltada.

Carpetas y variables

Las carpetas son una manera muy útil de manejar variables organizándolas en grupos relacionados entre sí.

La TI-89 Titanium /Voyage™ 200 calculadora gráfica incorpora una carpeta denominada **MAIN**. A no ser que el usuario cree carpetas y designe una de ellas como carpeta actual, todas las variables se almacenan en la carpeta **MAIN** de forma predeterminada. Sin embargo, una variable del sistema o una variable con un nombre reservado sólo se puede almacenar en la carpeta **MAIN**.

Ejemplos de variables que únicamente se pueden almacenar en MAIN

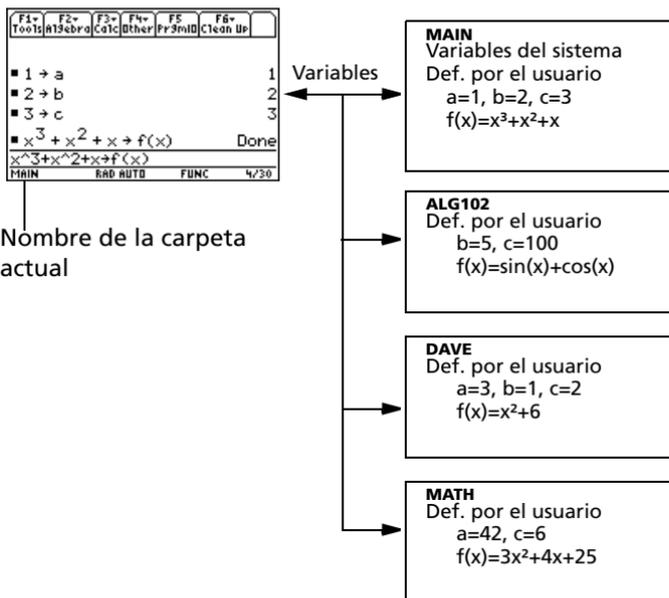
Variables de ventana
(**xmin**, **xmax**, etc.)

Variables de configuración de tablas
(**TblStart**, **ΔTbl**, etc.)

Funciones de Y= Editor
(**y1(x)**, etc.)

Si crea más carpetas, puede guardar en ellas conjuntos independientes de variables (y funciones) definidas por el usuario. Por ejemplo, puede crear carpetas separadas para distintas aplicaciones (Math, Text Editor, etc.) de la TI-89 Titanium / Voyage™ 200. Puede almacenar una variable definida por el usuario en cualquier carpeta.

Las variables definidas por el usuario contenidas en una carpeta son independientes de las de otras carpetas. Por ello, las carpetas pueden guardar conjuntos de variables con los mismos nombres pero con valores diferentes.



No se puede crear una carpeta dentro de otra.

Es posible acceder directamente a las variables del sistema que se encuentren en la carpeta **MAIN**, independientemente de cuál sea la carpeta actual.

Nota: Las variables definidas por el usuario se almacenan en la carpeta en que nos encontremos.

Creación de una carpeta en la pantalla VAR-LINK

1. Pulse **[2nd]** [VAR-LINK].
2. Pulse **[F1]** **Manage** y seleccione **5:Create Folder**.
3. Escriba un nombre de carpeta de hasta ocho caracteres y pulse dos veces **[ENTER]**.



La carpeta nueva que se crea en **VAR-LINK** no se establece automáticamente como la carpeta actual.

Creación de una carpeta en la pantalla Home

Introduzca la orden **NewFold** en la pantalla Home de la calculadora.

NewFold

Nombre de la carpeta que va a crear. Se establece automáticamente como la carpeta actual.

Ajuste de la carpeta actual en la pantalla Home

Introduzca la función **setFold** en la pantalla Home de la calculadora.

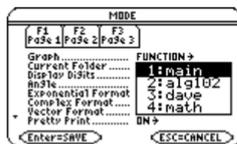
setFold (Nombre_de_carpeta)

setFold es una función, por lo que debe incluir el nombre de la carpeta entre paréntesis.

Al ejecutar **setFold**, se obtiene el nombre de la carpeta establecida previamente como la carpeta actual.

Ajuste de la carpeta actual en el cuadro de diálogo MODE

1. Pulse **MODE**.
2. Resalte el estado de **Current Folder**.
3. Pulse **⏏** para mostrar el menú de las carpetas existentes.



Nota: Para cancelar el menú o salir del cuadro de diálogo sin guardar los cambios, pulse **ESC**.

4. Seleccione una carpeta. Realice lo siguiente:
 - Resalte el nombre de la carpeta y pulse **ENTER**.
 - o bien –
 - Pulse el número o letra correspondiente a la carpeta.
5. Pulse **ENTER** para guardar los cambios y cerrar el cuadro de diálogo.

Nuevo nombre de variables o carpetas

Recuerde que si utiliza **F4** para seleccionar una carpeta, las variables en la misma se seleccionan automáticamente. Utilice **F4** también para anular las variables seleccionadas una a una, si fuese necesario.

1. En **VAR-LINK**, seleccione las variables y/o carpetas.

- Pulse **[F1] Manage** y seleccione **3:Rename**.
- Escriba un nombre y pulse **[ENTER]** dos veces.

Si ha seleccionado varios elementos, se le pedirá que introduzca un nuevo nombre para cada uno.



Uso de variables en carpetas distintas

Es posible acceder a una variable o función definida por el usuario que no esté en la carpeta actual. Especifique la ruta de acceso en vez de sólo el nombre de la variable.

Una ruta de acceso tiene la siguiente forma:

Nombre de carpeta\Nombre_de_variable

– o bien –

Nombre de carpeta\Nombre_de_función

Por ejemplo:

Si la carpeta actual = MAIN	Carpetas y variables															
<table border="1"> <tr><td>1 → a</td><td>1</td></tr> <tr><td>$x^3 + x^2 + x \rightarrow f(x)$</td><td>Done</td></tr> <tr><td>42 → math\ a</td><td>42</td></tr> <tr><td>$3 \cdot x^2 + 4 \cdot x + 25 \rightarrow math \setminus f(x)$</td><td>Done</td></tr> <tr><td colspan="2">$3 * x ^ 2 + 4 * x + 25 \rightarrow math \setminus f(x)$</td></tr> <tr><td>MAIN</td><td>RAD AUTO FUNC 4/30</td></tr> </table>	1 → a	1	$x^3 + x^2 + x \rightarrow f(x)$	Done	42 → math\ a	42	$3 \cdot x^2 + 4 \cdot x + 25 \rightarrow math \setminus f(x)$	Done	$3 * x ^ 2 + 4 * x + 25 \rightarrow math \setminus f(x)$		MAIN	RAD AUTO FUNC 4/30	<table border="1"> <tr><td>MAIN</td></tr> <tr><td>a=1</td></tr> <tr><td>$f(x)=x^2+x^2+x$</td></tr> </table>	MAIN	a=1	$f(x)=x^2+x^2+x$
1 → a	1															
$x^3 + x^2 + x \rightarrow f(x)$	Done															
42 → math\ a	42															
$3 \cdot x^2 + 4 \cdot x + 25 \rightarrow math \setminus f(x)$	Done															
$3 * x ^ 2 + 4 * x + 25 \rightarrow math \setminus f(x)$																
MAIN	RAD AUTO FUNC 4/30															
MAIN																
a=1																
$f(x)=x^2+x^2+x$																
<table border="1"> <tr><td>4 · a</td><td>4</td></tr> <tr><td>4 · math\ a</td><td>168</td></tr> <tr><td>f(5)</td><td>155</td></tr> <tr><td>math\ f(5)</td><td>120</td></tr> <tr><td colspan="2">math\ f(5)</td></tr> <tr><td>MAIN</td><td>RAD AUTO FUNC 4/30</td></tr> </table>	4 · a	4	4 · math\ a	168	f(5)	155	math\ f(5)	120	math\ f(5)		MAIN	RAD AUTO FUNC 4/30	<table border="1"> <tr><td>MATH</td></tr> <tr><td>a=42</td></tr> <tr><td>$f(x)=3x^2+4x+25$</td></tr> </table>	MATH	a=42	$f(x)=3x^2+4x+25$
4 · a	4															
4 · math\ a	168															
f(5)	155															
math\ f(5)	120															
math\ f(5)																
MAIN	RAD AUTO FUNC 4/30															
MATH																
a=42																
$f(x)=3x^2+4x+25$																

Para ver una lista de las carpetas y variables ya existentes, pulse **[2nd] [VAR-LINK]**. En la pantalla **VAR-LINK**, puede resaltar una variable y pulsar **[ENTER]** para pegar el nombre de la misma en la línea de entrada de la aplicación abierta. Si pega un nombre de variable que no está en la carpeta actual, también se pega su ruta de acceso (*Nombre_de_carpeta\Nombre_de_variable*).

Listado de sólo una carpeta y/o tipo de variable o aplicación Flash

Si tiene muchas variables y/o carpetas o aplicaciones Flash, resultará difícil localizar una en particular. Si cambia la visualización de **VAR-LINK**, puede especificar la información que desea ver.

En la pantalla **VAR-LINK**:

1. Pulse **[F2] View**.
2. Resalte el estado que desea cambiar y pulse **[↓]**. Se presenta un menú con las opciones válidas. (Para cancelar un menú, pulse **[ESC]**.)

View — Permite elegir variables, aplicaciones Flash o variables del sistema para su visualización.

Nota: Para mostrar las variables del sistema (variables de ventana, etc.), seleccione **3:System**.

Folder — Como mínimo, siempre muestra **1:All** y **2:main**, apareciendo otras carpetas únicamente si se han creado.

Var Type — Muestra los tipos válidos de variables.

↓ — indica que puede desplazarse para buscar más variables.



3. Seleccione el nuevo estado.
4. Cuando vuelva a la pantalla **VAR-LINK VIEW**, pulse **[ENTER]**.

La pantalla **VAR-LINK** se actualiza para mostrar sólo la carpeta, tipo de variable, que se haya especificado.

Copia o movimiento de variables entre carpetas

Debe tener otra carpeta, por lo menos, además de **MAIN**. No puede utilizar **VAR-LINK** para copiar variables en la misma carpeta.

1. Seleccione las variables en **VAR-LINK**.
2. Pulse **[F1] Manage** y seleccione **2:Copy** o **4:Move**.
3. Seleccione la carpeta de destino.



- Pulse **[ENTER]**. Las variables copiadas o cambiadas de lugar conservan sus nombres originales.

Nota: Para copiar una variable con un nombre distinto en la misma carpeta, utilice **[STO]** (como a1→a2) o la orden **CopyVar** en la pantalla Home.

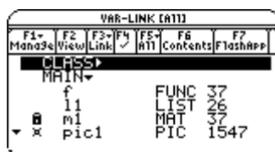
Bloqueo y desbloqueo de variables, carpetas y aplicaciones Flash

Las variables bloqueadas no se pueden borrar, cambiar de nombre ni almacenar, aunque sí se puede copiar, mover o presentar su contenido. Cuando una carpeta está bloqueada, puede manejar las variables en la misma (siempre que no estén bloqueadas), pero no puede borrarla.

- En **VAR-LINK**, seleccione las variables, las carpetas, o la aplicación Flash.
- Pulse **[F1] Manage** y seleccione **6:Lock Variable** o **7:UnLock Variable**.

indica una variable o carpeta bloqueada en RAM.

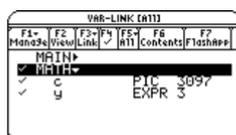
designa una variable archivada, que se bloquea automáticamente.



Borrado de una carpeta en la pantalla VAR-LINK

Cuando se borra una carpeta en la pantalla **VAR-LINK**, también se borran todas las variables de esa carpeta. No es posible borrar la carpeta **MAIN**.

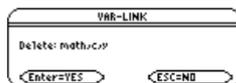
- Pulse **[2nd] [VAR-LINK]**.
- Pulse **[F4]** para seleccionar las carpetas que desea borrar. (Las variables de dichas carpetas se seleccionan automáticamente.)



- Pulse **[F1] 1:Delete** o **[←]**.



- Pulse **[ENTER]** para confirmar el borrado de la carpeta y todas sus variables.



Borrado de una variable o de una carpeta en la pantalla Home

Antes de borrar una carpeta en la pantalla Home de la calculadora, debe indicarle que borre todas las variables almacenadas en la misma.

- Para borrar una variable, introduzca la orden **DelVar** en la pantalla Home de la calculadora.

DelVar *var1* [, *var2*] [, *var3*] ...

- Para borrar todas las variables de un tipo específico, introduzca la orden **DelType** en la pantalla Home de la calculadora.

DelType *tipo_de_variable* donde *tipo_de_variable* corresponde al tipo de la variable en cuestión.

Nota: La orden **DelType** borra todas las variables del tipo especificado de todas las carpetas.

- Si desea borrar una carpeta vacía, introduzca la orden **DelFold** en la pantalla Home de la calculadora.

DelFold *carpeta1* [, *carpeta2*] [, *carpeta3*] ...

Nota: No es posible borrar la carpeta **MAIN**.

Pegado de un nombre de variable en una aplicación

Supongamos que escribe una expresión en la pantalla Home y no recuerda la variable que debe utilizar. Puede presentar la pantalla **VAR-LINK**, seleccionar una variable de la lista y pegar el nombre de dicha variable directamente en la línea de entrada de la pantalla Home.

¿Qué aplicaciones pueden utilizarse?

Puede pegar un nombre de variable en la posición actual del cursor desde las siguientes aplicaciones.

- Home screen, Y= Editor, Table Editor o Data/Matrix Editor: el cursor debe estar en la línea de entrada
- Text Editor, Window Editor, Numeric Solver o Program Editor: el cursor puede estar en cualquier lugar de la pantalla.

También es posible pegar un nombre de variable en la posición actual del cursor en muchas aplicaciones Flash.

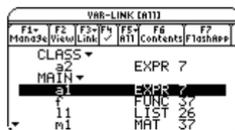
Procedimiento

Comenzando en una de las aplicaciones indicadas arriba:

1. Coloque el cursor en la posición en que quiere insertar el nombre de la variable.

sin(|

2. Pulse [2nd] [VAR-LINK].



3. Resalte la variable correspondiente.

Nota: También puede resaltar y pegar nombres de carpetas.

4. Pulse [ENTER] para pegar el nombre de la variable.

sin(a1|

Nota: Se pega el nombre de la variable, no su contenido (utilice [2nd] [RCL] en vez de [2nd] [VAR-LINK], si desea recuperar el contenido de una variable).

5. Termine de escribir la expresión.

sin(a1)

Si pega un nombre de variable que no está en la carpeta actual, se pega el camino de la variable.

sin(Class\la2

En el caso de que CLASS no sea la carpeta actual, se pega lo que aquí se indica al resaltar la variable a2 en CLASS.

Archivo y extracción de variables

Para archivar o extraer variables del archivo de forma interactiva, utilice la pantalla **VAR-LINK**. También es posible realizar estas operaciones en la pantalla Home (Principal) o desde un programa.

¿Para qué archivar una variable?

El archivo de datos del usuario permite:

- Almacenar datos, programas o cualquier otra variable en un lugar seguro en el que no puedan modificarse ni eliminarse de forma inadvertida.
- Crear más espacio libre de RAM mediante el archivo de variables. Por ejemplo:

- Puede archivar las variables a las que necesite acceder pero no modificar, o variables que no esté utilizando actualmente pero que necesita retener para uso futuro.

Nota: No pueden archivar variables con nombres reservados o variables del sistema.

- Si adquiere otros programas para la TI-89 Titanium / Voyage™ 200 calculadora gráfica, sobre todo si son grandes, puede que necesite crear más espacio libre en la RAM antes de poder instalarlos.

Disponer de más espacio libre en la RAM puede acortar el tiempo de ejecución de determinados tipos de operaciones.

En la pantalla VAR-LINK

Para archivar o extraer del archivo:

1. Pulse **[2nd]** [VAR-LINK] para presentar la pantalla **VAR-LINK**.
2. Seleccione las variables, que pueden encontrarse en diferentes carpetas (puede seleccionar una carpeta completa seleccionando su nombre).

Nota: Para seleccionar una sola variable, resáltela. Para seleccionar dos o más variables, resalte cada variable y pulse **[F4]** ✓.

3. Pulse **[F1]** y seleccione:

8:Archive Variable

– o bien –

9:Unarchive Variable



Si selecciona **8:Archive Variable**, las variables se desplazan al archivo de datos del usuario.

⌘ = variables archivadas



Puede acceder a una variable guardada con el mismo procedimiento que a cualquier variable bloqueada. En cualquier caso, una variable guardada sigue encontrándose en su carpeta original; simplemente se almacena en el archivo de datos del usuario en vez de en la RAM.

Nota: Las variables archivadas se bloquean de forma automática. Puede acceder a ellas, pero no modificarlas ni eliminarlas.

En la pantalla Home o desde un programa

Utilice las órdenes **Archive** y **Unarchiv**.

Archive *variable1, variable2, ...*

Unarchiv *variable1, variable2, ...*

Mensaje Garbage Collection

Si el archivo de datos del usuario se utiliza abusivamente, puede que se presente el mensaje Garbage Collection, lo que se produce al intentar archivar una variable cuando el archivo no tiene suficiente memoria libre. No obstante, la TI-89 Titanium / Voyage™ 200 intentará distribuir las variables archivadas para obtener más espacio.

Respuesta al mensaje de Garbage Collection

Cuando vea el mensaje de la derecha:

- Para continuar archivando, pulse **ENTER**.
– o bien –
- Para cancelar la operación, pulse **ESC**.



Nota: Si las pilas están bajas de energía sustitúyalas antes de comenzar el proceso para evitar que se pierdan los datos archivados en la memoria.

Después de redistribuir la memoria, según el espacio libre adicional que se libere, puede que la variable se archive o no. En caso negativo, extraiga algunas variables del archivo e inténtelo de nuevo.

Razones para no ejecutar la redistribución de memoria automáticamente sin el mensaje correspondiente

El mensaje:

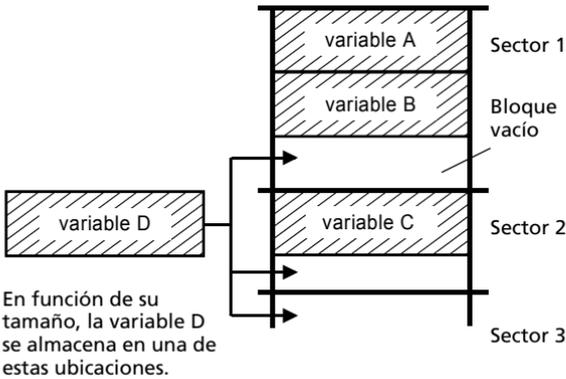
- Permite conocer las causas por las que una operación de archivo lleva más tiempo del normal. También advierte que la operación de archivado puede fallar si no hay suficiente memoria.
- Puede avisarle cuando un programa caiga en un bucle que, de forma repetida, llene el archivo de datos del usuario. Cancele la operación y estudie la causa.

Necesidad de redistribuir la memoria

El archivo de datos del usuario está dividido en sectores. Cuando se inicia una operación de archivo, las variables se almacenan de forma consecutiva en el sector 1. Así se continua hasta el final del sector. Si no hay suficiente espacio disponible en el sector, la siguiente variable se almacena en el inicio del siguiente sector. Por lo general, esto ocasiona que quede un bloque vacío al final del sector anterior.

Cada variable que se archiva se almacena en el primer bloque vacío de tamaño suficiente como para darle cabida.

Nota: Una variable archivada se almacena en un bloque continuo dentro de un sector; no puede cruzar el límite de un sector.

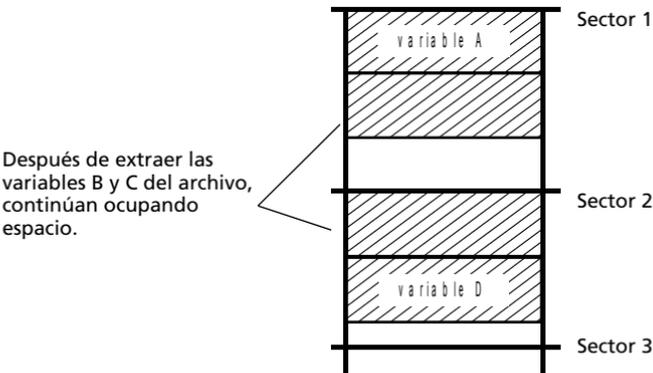


Este proceso continúa hasta el final del último sector. En función del tamaño de cada variable, los bloques vacíos pueden contener una cantidad significativa de espacio.

Nota: La redistribución de memoria se produce cuando el tamaño de la variable que se está archivando es mayor que cualquier bloque vacío.

Influencia de la extracción de una variable del archivo en el proceso

Cuando se extrae una variable del archivo, se copia en la RAM pero no se elimina realmente de la memoria del archivo de datos del usuario.

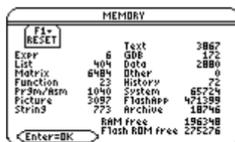


Las variables extraídas del archivo están "marcadas para eliminar", lo que significa que se eliminarán durante la siguiente redistribución de memoria.

Si la pantalla MEMORY indica que hay suficiente espacio libre

Aun en el caso de que la pantalla **MEMORY** indique suficiente espacio libre para archivar una variable, puede que siga recibiendo un mensaje de redistribución de memoria.

Esta pantalla de memoria de la TI-89 Titanium muestra el espacio libre que habrá disponible después de eliminar todas las variables “marcadas para eliminar”.



MEMORY			
Fi	RESET		
Exp	6	Text	3867
List	404	ODE	122
Matrix	6484	Data	2880
Function	23	Other	0
Program	1040	System	72
Picture	3087	FlashApp	6524
String	772	Archive	471589
		RAM free	186348
		Flash ROM free	275276

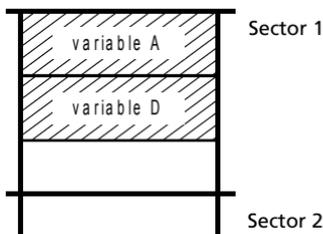
Cuando se extrae una variable del Flash ROM, la cantidad indicada por Archive free aumenta inmediatamente, pero el espacio no está realmente disponible hasta después de realizarse la siguiente redistribución de memoria.

La Voyage™ 200 tiene 2,7 MB de memoria Flash ROM disponible para el usuario. Los 2,7 MB completos pueden usarse para Apps Flash, pero de ellos sólo alrededor de 1 MB pueden emplearse para archivar datos del usuario.

Proceso de redistribución de memoria

El proceso de redistribución de memoria:

- Elimina las variables extraídas del archivo de datos del usuario.
- Distribuye las restantes variables en bloques consecutivos.



Error de memoria al acceder a una variable del archivo

Una variable del archivo se considera de forma similar a una variable bloqueada. Es posible acceder a la variable, pero no puede modificarse ni eliminarse. En algunos casos, no obstante, puede que obtenga un mensaje de error de memoria cuando intente acceder a una variable del archivo.

Causas del error de memoria

El mensaje **Memory Error** se presenta cuando no hay suficiente RAM libre para acceder a la variable del archivo. Es posible que esto le haga preguntarse, "¿si la variable se encuentra en el archivo de datos del usuario, ¿qué tiene que ver la cantidad de RAM disponible?" La respuesta es que las siguientes operaciones pueden llevarse a cabo sólo si una variable se encuentra en la RAM.

- Apertura de una variable de texto en Text Editor.
- Apertura de una variable de datos, lista o matriz en Data/Matrix Editor.
- Apertura de un programa o función en Program Editor.
- Ejecución de un programa haciendo referencia a una función.

Nota: Como se explica a continuación, una copia temporal permite abrir o ejecutar una variable archivada. No obstante, no es posible guardar ningún cambio efectuado en la misma.

Así que no se tienen que extraer variables del archivo de forma innecesaria: la TI-89 Titanium / Voyage™ 200 calculadora gráfica lleva a cabo una copia "entre bastidores". Por ejemplo, si se ejecuta un programa que se encuentra en el archivo de datos del usuario, la TI-89 Titanium / Voyage™ 200:

1. Copia el programa en la RAM.
2. Ejecuta el programa.
3. Elimina la copia de la RAM cuando el programa finaliza.

El mensaje de error se presenta si en la RAM no hay suficiente espacio libre para la copia temporal.

Nota: Excepto para programas y funciones, hacer referencia a una variable archivada no ocasiona su copia. Si la variable ab está archivada, no se copia si se ejecuta **6*ab**.

Corrección del error

Para liberar suficiente espacio en la RAM para acceder a la variable:

1. Utilice la pantalla **VAR-LINK** (2nd [VAR-LINK]) para calcular el tamaño de la variable archivada a la que desea acceder.
2. Utilice la pantalla **MEMORY** (2nd [MEM]) para verificar el tamaño de RAM libre.
3. Libere la cantidad de memoria necesaria mediante:
 - La eliminación de variables innecesarias de la RAM.

- El archivo de variables o programas de gran tamaño (mediante su traslado desde la RAM al archivo de datos del usuario).

Nota: Por lo general, el tamaño de RAM libre debe superar al de la variable archivada.

Apéndice A: Funciones e instrucciones

Listado catégorico de operaciones.....	178
Listado alfabético de operaciones	182

En esta sección se describe la sintaxis y la acción de todas las funciones e instrucciones de la TI-89 Titanium / Voyage™ 200 incluidas en el sistema operativo (SO). Las instrucciones específicas de las aplicaciones de software para dispositivos de mano (Apps) se encuentran en los módulos correspondientes a dichas Apps.

Nombre de la función o instrucción.

Tecla o menú para introducir el nombre. También puede escribirlo.

Ejemplo

Circle CATALOG

Circle *x, y, r[, modoDraw]*

Dibuja una circunferencia con el centro en las coordenadas de ventana (*x, y*) y con un radio *r*.

x, y, y r deben ser valores reales.

Si *modoDraw* = 1, dibuja la circunferencia (predeterminado).

Si *modoDraw* = 0, desactiva la circunferencia.

Si *modoDraw* = -1, invierte los pixels de la circunferencia.

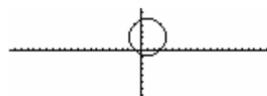
Nota: Al volver a representar gráficamente, se borran todos los elementos dibujados.

Los argumentos se muestran en *cursiva*. Los argumentos entre corchetes [] son opcionales. No escriba los corchetes.

La línea de sintaxis muestra el orden y el tipo de argumentos que se deben introducir. Debe separar los argumentos con una coma (,).

En una ventana de visualización ZoomSqr:

ZoomSqr:Circle 1,2,3 [ENTER]



Explicación de la función o instrucción.

Listado categórico de operaciones

Esta sección proporciona una lista de las funciones e instrucciones de la TI-89 Titanium / Voyage™ 200 en grupos funcionales, junto con las páginas en las que aparecen descritas en este anexo.

Álgebra

 ("with")	209	cFactor()	187	comDenom()	190
cSolve()	196	cZeros()	200	expand()	209
factor()	209	getDenom()	209	getNum()	209
nSolve()	209	propFrac()	209	randPoly()	209
solve()	209	tCollect()	209	tExpand()	209
zeros()	209				

Cálculo

f() (integrar)	209	Π() (prod.)	209	Σ() (suma)	209
arcLen()	184	avgRC()	185	d()	202
deSolve()	205	fMax()	209	fMin()	209
ImpDif()	209	limit()	209	nDeriv()	209
nInt()	209	' (primo)	209	seq()	209
taylor()	209				

Gráficos

AndPic	183	BldData	186	Circle	188
ClrDraw	188	ClrGraph	189	CyclePic	200
DrawFunc	209	DrawInv	209	DrawParm	209
DrawPol	209	DrawSlp	209	DrwCtour	209
FnOff	209	FnOn	209	Graph	209
Line	209	LineHorz	209	LineTan	209
LineVert	209	NewPic	209	PtChg	209
PtOff	209	PtOn	209	ptTest()	209
PtText	209	PxlChg	209	PxlCrcI	209
PxlHorz	209	PxlLine	209	PxlOff	209
PxlOn	209	pxlTest()	209	PxlText	209
PxlVert	209	RclGDB	209	RclPic	209
RplcPic	209	Shade	209	StoGDB	209
StoPic	209	Style	209	Trace	209
XorPic	209	ZoomBox	209	ZoomData	209
ZoomDec	209	ZoomFit	209	ZoomIn	209
ZoomInt	209	ZoomOut	209	ZoomPrev	209
ZoomRcl	209	ZoomSqr	209	ZoomStd	209
ZoomSto	209	ZoomTrig	209		

Listas

+ (suma)	209	- (resta)	209	* (multiplic.)	209
/ (división)	209	- (negativo)	209	^ (potencia)	209
augment()	185	crossP()	194	cumSum()	198
dim()	207	dotP()	209	explist()	209
left()	209	listmat()	209	Δlist()	209
matlist()	209	max()	209	mid()	209
min()	209	newList()	209	polyEval()	209
product()	209	right()	209	rotate()	209
shift()	209	SortA	209	SortD	209
sum()	209				

Matemáticas

+ (suma)	209	- (resta)	209	* (multiplic.)	209
/ (división)	209	- (negativo)	209	% (porcent.)	209
! (factorial)	209	√() (raíz cuad.)	209	^ (potencia)	209
° (gradian)	209	° (grados)	209	∠ (ángulo)	209
° , ' , "	209	_ (subrayado)	209	▶ (conversión)	209
10^()	209	0b , 0h	209	▶Bin	185
▶Cylind	200	▶DD	203	▶Dec	203
▶DMS	208	▶Grad	182	▶Hex	209
▶In	209	▶logbase	209	▶Polar	209
▶Rad	209	▶Rect	209	▶Sphere	209
abs()	182	and	182	angle()	183
approx()	184	ceilng()	186	conj()	191
cos()	191	cos⁻¹()	192	cosh()	193
cosh⁻¹()	193	cot()	193	cot⁻¹()	194
coth()	194	coth⁻¹()	194	csc()	195
csc⁻¹()	195	csch()	195	csch⁻¹()	195
E	209	e[^]	209	exact()	209
floor()	209	fPart()	209	gcd()	209
imag()	209	impDif()	209	int()	209
intDiv()	209	iPart()	209	isPrime()	209
lcm()	209	ln()	209	log()	209
max()	209	min()	209	mod()	209
nCr()	209	nPr()	209	▶Rx()	209
▶Ry()	209	r (radianes)	209	▶Rθ()	209
▶Pr()	209	real()	209	remain()	209
root()	209	rotate()	209	round()	209
sec()	209	sec⁻¹()	209	sech()	209
sech⁻¹()	209	shift()	209	sign()	209
sin()	209	sin⁻¹()	209	sinh()	209
sinh⁻¹()	209	tan()	209	tan⁻¹()	209
tanh()	209	tanh⁻¹()	209	tmpCnv()	209
ΔtmpCnv()	209	x⁻¹	209		

Matrices

+ (suma)	209	- (resta)	209	* (multiplic.)	209
/ (división)	209	- (negativo)	209	+ (pto., suma)	209
- (pto., resta)	209	.* (pto., mult.)	209	./ (pto., div.)	209
.^ (pto., pot.)	209	^ (potencia)	209	augment()	185
colDim()	190	colNorm()	190	crossP()	194
cumSum()	198	data▶mat	202	det()	206
diag()	206	dim()	207	dotP()	209
eigVc()	209	eigVl()	209	Fill	209
identity()	209	list▶mat()	209	LU	209
mat▶data	209	mat▶list()	209	max()	209
mean()	209	median()	209	min()	209
mRow()	209	mRowAdd()	209	newMat()	209
norm()	209	product()	209	QR	209
randMat()	209	ref()	209	rowAdd()	209
rowDim()	209	rowNorm()	209	rowSwap()	209
rref()	209	simult()	209	stdDev()	209
stdDevPop()	209	subMat()	209	sum()	209
T (trasp.)	209	unitV()	209	variance()	209
x⁻¹	209				

Programación

=	209	≠	209	<	209
≤	209	>	209	≥	209
# (dir. indirec.)	209	> (almac.)	209	Ⓞ (coment.)	209
and	182	ans()	184	Archive	184
checkTmr()	188	ClockOff	188	ClockOn	188
ClrErr	189	ClrGraph	189	ClrHome	189
ClrIO	189	ClrTable	190	CopyVar	191
CustmOff	199	CustmOn	199	Custom	199
Cycle	199	dayOfWk()	202	Define	203
DelFold	204	DelType	204	DelVar	204
Dialog	207	Disp	207	DispG	208
DispHome	208	DispTbl	208	DropDown	209
Else	209	Elseif	209	EndCustm	209
EndDlog	209	EndFor	209	EndFunc	209
EndIf	209	EndLoop	209	EndPrgm	209
EndTBar	209	EndTry	209	EndWhile	209
entry()	209	Exec	209	Exit	209
For	209	format()	209	Func	209
Get	209	GetCalc	209	getConfig()	209
getDate()	209	getDtfmt()	209	getDtStr()	209
getFold()	209	getKey()	209	getMode()	209
getTime()	209	getTmfmt()	209	getTmStr()	209
getTmZn()	209	getType()	209	getUnits()	209
Goto	209	If	209	Input	209
InputStr	209	isArchiv()	209	IsClkOn()	209
isLocked ()	209	isVar()	209	Item	209
Lbl	209	left()	209	Local	209
Lock	209	Loop	209	MoveVar	209
NewFold	209	NewProb	209	not	209
or	209	Output	209	part()	209
PassErr	209	Pause	209	PopUp	209
Prgm	209	Prompt	209	Rename	209
Request	209	Return	209	right()	209
Send	209	SendCalc	209	SendChat	209
setDate()	209	setDtfmt()	209	setFold()	209
setGraph()	209	setMode()	209	setTable()	209
setTime()	209	setTmfmt()	209	setTmZn()	209
setUnits()	209	starttmr()	209	Stop	209
Style	209	switch()	209	Table	209
Text	209	Then	209	timeCnv()	209
Title	209	ToolBar	209	Try	209
Unarchiv	209	Unlock	209	when()	209
While	209	xor	209		

Estadística	! (factorial)	209	BldData	186	CubicReg	198	
	cumSum()	198	ExpReg	209	LinReg	209	
	LnReg	209	Logistic	209	mean()	209	
	median()	209	MedMed	209	nCr()	209	
	NewData	209	NewPlot	209	nPr()	209	
	OneVar	209	PlotsOff	209	PlotsOn	209	
	PowerReg	209	QuadReg	209	QuartReg	209	
	rand()	209	randNorm()	209	RandSeed	209	
	ShowStat	209	SinReg	209	SortA	209	
	SortD	209	stdDev()	209	TwoVar	209	
	variance()	209					
	Cadenas	& (anex.)	209	# (dir. indirec.)	209	char()	187
		dim()	207	expr()	209	format()	209
inString()		209	left()	209	mid()	209	
ord()		209	right()	209	rotate()	209	
shift()		209	string()	209			

Listado alfabético de operaciones

Las operaciones con nombres sin letras (como +, ! y >) aparecen al final de este anexo, a partir de la página 209. A menos que se indique lo contrario, todos los ejemplos de esta sección se realizaron en el modo de inicio predeterminado, presuponiendo que ninguna de las variables estaba definida. Además, debido a limitaciones del formato, los resultados aproximados se han truncado a tres espacios decimales (3.14159265359 aparece como 3.141...).

abs() Menú MATH/Number

abs(*expresión1*) ⇒ *expresión*

abs(*lista1*) ⇒ *lista*

abs(*matriz1*) ⇒ *matriz*

Devuelve el valor absoluto del argumento.

Si el argumento es un número complejo, halla el módulo del número.

Nota: Trata todas las variables no definidas como variables reales.

abs($\{\pi/2, \pi/3\}$) **[ENTER]**

$\left\{\frac{\pi}{2}, \frac{\pi}{3}\right\}$

abs($2-3i$) **[ENTER]**

$\sqrt{13}$

abs(*z*) **[ENTER]**

$|z|$

abs($x+y$) **[ENTER]**

$\sqrt{x^2+y^2}$

and Menús MATH/Test y MATH/Base

expresión booleana1 **and** *expresión2* ⇒ *expresión booleana*

lista booleana1 **and** *lista2* ⇒ *lista booleana*

matriz booleana1 **and** *matriz2* ⇒ *matriz booleana*

Devuelve true o false, o la entrada original simplificada.

$x \geq 3$ **and** $x \geq 4$ **[ENTER]**

$x \geq 4$

$\{x \geq 3, x \leq 0\}$ **and** $\{x \geq 4, x \leq -2\}$ **[ENTER]**

$\{x \geq 4, x \leq -2\}$

entero1 **and** *entero2* ⇒ *entero*

Compara dos números enteros bit a bit mediante una operación **and**. Internamente, ambos enteros se convierten en números binarios de 32 bits con su correspondiente signo. Cuando se comparan los bits correspondientes, el resultado es 1 si ambos bits son 1; en caso contrario, el resultado es 0. El valor devuelto representa los resultados de bits y se presenta de acuerdo con el estado del modo Base.

Los enteros pueden introducirse en cualquier base. Para una entrada binaria o hexadecimal, se debe utilizar el prefijo 0b ó 0h, respectivamente. Sin un prefijo, los enteros se tratan como decimales (base 10).

Si se introduce un entero decimal demasiado grande para una forma binaria de 32 bits con signo, se utiliza una operación de módulos simétricos para llevar el valor al rango apropiado.

En el modo de base Hex:

0h7AC36 **and** 0h3D5F **[ENTER]**

0h2C16

Importante: Cero, no la letra O.

En el modo de base Bin:

0b100101 **and** 0b100 **[ENTER]**

0b100

En el modo de base Dec:

37 **and** 0b100 **[ENTER]**

4

Nota: Las entradas binarias pueden tener hasta 32 dígitos (sin contar el prefijo 0b); las hexadecimales, un máximo de 8 dígitos.

AndPic CATALOG

AndPic *picVar*, *fila*, *columna*

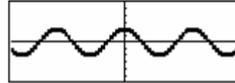
Muestra la pantalla Graph y procesa con "AND" lógico la imagen almacenada en *picVar* y la pantalla gráfica actual en las coordenadas del pixel (*fila*, *columna*).

picVar debe ser un tipo de imagen.

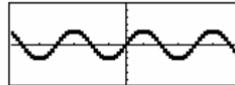
Las coordenadas por omisión son (0,0), que es la esquina superior izquierda de la pantalla.

En el modo de gráficas de función e Y= Editor:

$y1(x) = \cos(x)$ 
 **2nd**[F6] Style = 3:Square
 **F6** Style = 3:Square
F2 Zoom = 7:ZoomTrig
F1 = 2:Save Copy As...
 Type = Picture. Variable = PIC1

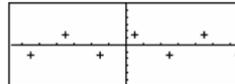


$y2(x) = \sin(x)$
 **2nd**[F6] Style = 3:Square
 **F6** Style = 3:Square
 $y1$ = no checkmark (F4 to deselect)
F2 Zoom = 7:ZoomTrig



 **HOME**
 **CALC HOME**
 AndPic PIC1 **ENTER**

Done



angle() Menú MATH/Complex

angle(*expresión1*) \Rightarrow *expresión*

Devuelve el ángulo de la *expresión1*, interpretando la *expresión1* como un número complejo.

Nota: Trata todas las variables no definidas como variables reales.

En el modo Angle, en grados:

$\text{angle}(0+2i)$ **ENTER** 90

En el modo Angle, en grados centesimales:

$\text{angle}(0+3i)$ **ENTER** 100

En el modo Angle, en radianes:

$\text{angle}(1+i)$ **ENTER** $\frac{\pi}{4}$

$\text{angle}(z)$ **ENTER**

$\text{angle}(x+iy)$ **ENTER**

$$\begin{aligned} \blacksquare \text{angle}(z) &= \frac{-\pi \cdot (\text{sign}(z) - 1)}{2} \\ \blacksquare \text{angle}(x + i \cdot y) &= \frac{\pi \cdot \text{sign}(y)}{2} - \tan^{-1}\left(\frac{x}{y}\right) \end{aligned}$$

angle(*lista1*) \Rightarrow *lista*

angle(*matriz1*) \Rightarrow *matriz*

Devuelve una lista o matriz de los ángulos de los elementos en la *lista1* o *matriz1*, interpretando cada elemento como un número complejo que represente las coordenadas rectangulares bidimensionales de un punto.

En el modo Angle, en radianes:

$\text{angle}(\{1+2i, 3+0i, 0-4i\})$ **ENTER**

$$\blacksquare \text{angle}(\{1+2i, 3+0i, 0-4i\}) \begin{Bmatrix} \frac{\pi}{2} - \tan^{-1}(1/2) & 0 & -\frac{\pi}{2} \end{Bmatrix}$$

ans() Tecla 2^{nd} [ANS]

ans() \Rightarrow *valor*
ans(entero) \Rightarrow *valor*

Devuelve una respuesta anterior del área de historia de la pantalla Home.

El *entero*, si se incluye, especifica la respuesta anterior a la que esté llamando. El rango válido del *entero* está comprendido entre 1 y 99, y no puede ser una expresión. El valor por omisión es 1, la respuesta más reciente.

Para utilizar **ans()** y generar la sucesión Fibonacci en la pantalla Home, pulse:

```
1 [ENTER] 1
1 [ENTER] 1
2nd [ANS] + 2nd [ANS]  $\leftarrow$  2 [ENTER] 2
[ENTER] 3
[ENTER] 5
```

approx() Menú MATH/Algebra

approx(*expresión*) \Rightarrow *valor*

Devuelve el valor de la *expresión* como número decimal cuando sea posible, sin tomar en cuenta el modo Exact/Approx actual.

Equivale a introducir la *expresión* y pulsar \leftarrow [ENTER] en la pantalla Home.

approx(π) [ENTER] 3.141...

approx(*lista*1) \Rightarrow *lista*

approx(*matriz*1) \Rightarrow *matriz*

Devuelve una lista o matriz en la que cada elemento se ha convertido a sus valores decimales.

```
approx({sin( $\pi$ ),cos( $\pi$ )}) [ENTER] {0. -1.}
approx([ $\sqrt{2}$ ), $\sqrt{3}$ ]) [ENTER] [1.414... 1.732...]
```

Archive CATALOG

Archive *var*1 [, *var*2] [, *var*3] ...

Desplaza las variables especificadas desde la RAM hasta la memoria de archivos de datos del usuario.

Puede acceder a una variable archivada de la misma forma que a una variable de la RAM. No obstante, no es posible borrar, renombrar o almacenar una variable archivada debido a que se bloquea de forma automática.

Para desarchivar variables, utilice **Unarchiv**.

```
10 $\rightarrow$ arctest [ENTER] 10
Archive arctest [ENTER] Done
5*arctest [ENTER] 50
15 $\rightarrow$ arctest [ENTER]
```



```
[ESC]
Unarchiv arctest [ENTER] Done
15 $\rightarrow$ arctest [ENTER] 15
```

arLen() Menú MATH/Calculus

arLen(*expresión*1, *var*, *inicio*, *fin*) \Rightarrow *expresión*

Devuelve la longitud de arco de la *expresión*1 entre *inicio* y *fin* con respecto a la variable *var*.

Con independencia del modo de representación gráfica, la longitud de arco se calcula como una integral, presuponiendo que se ha definido una función.

arLen(cos(x).x.0. π) [ENTER] 3.820...

arLen(f(x).x.a.b) [ENTER]

$$\int_a^b \sqrt{\left(\frac{d}{dx}(f(x))\right)^2 + 1} dx$$

arLen(*lista*1, *var*, *inicio*, *fin*) \Rightarrow *lista*

Devuelve una lista de las longitudes de arco de cada elemento de la *lista*1 entre *inicio* y *fin* respecto a *var*.

```
arLen({sin(x),cos(x)}.x.0. $\pi$ )
(3.820... 3.820...)
```

augment() Menú MATH/Matrix

augment(lista1, lista2) ⇒ lista

Devuelve una nueva lista compuesta por la *lista2* anexada al final de la *lista1*.

augment({1, -3.2}, {5,4}) **ENTER**
{1 -3 2 5 4}

augment(matrix1, matrix2) ⇒ matrix

augment(matrix1; matrix2) ⇒ matrix

Devuelve una nueva matriz anexando *matrix2* a *matrix1*. Cuando se utiliza el carácter ",", las matrices deben el mismo número de filas, y *matrix2* se anexa a *matrix1* como nuevas columnas. Cuando se utiliza el carácter ";", las matrices deben tener el mismo número de columnas, y *matrix2* se anexa a *matrix1* como nuevas filas. No modifica *matrix1* ni *matrix2*.

[1.2;3.4]►M1 **ENTER** $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$
[5:6]►M2 **ENTER** $\begin{bmatrix} 5 \\ 6 \end{bmatrix}$
augment(M1,M2) **ENTER** $\begin{bmatrix} 1 & 2 & 5 \\ 3 & 4 & 6 \end{bmatrix}$
[5,6]►M2 **ENTER** $\begin{bmatrix} 5 & 6 \end{bmatrix}$
augment(M1;M2) **ENTER** $\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$

avgRC() CATALOG

avgRC(expresión1, var[, h]) ⇒ expresión

Devuelve el cociente de diferencia incremental (índice de cambio promedio).

La *expresión1* puede ser el nombre de una función definida por el usuario (consulte **Func**).

h es el valor del incremento. Si se omite *h*, el valor por omisión es 0.001.

Tenga en cuenta que la función similar **nDeriv()** utiliza el cociente de diferencia central.

avgRC(f(x),x,h) **ENTER** $\frac{f(x+h) - f(x)}{h}$
avgRC(sin(x),x,h)|x=2 **ENTER** $\frac{\sin(h+2) - \sin(2)}{h}$
avgRC(x^2-x+2,x) **ENTER** 2 • (x - .4995)
avgRC(x^2-x+2,x,.1) **ENTER** 2 • (x - .45)
avgRC(x^2-x+2,x,3) **ENTER** 2 • (x+1)

Bin Menú MATH/Base

entero1►Bin ⇒ entero

Convierte el *entero1* en un número binario. Los números binarios o hexadecimales siempre tienen un prefijo 0b ó 0h, respectivamente.

┌ Cero, no la letra O, seguido de b ó h.

0b Número binario

0h Número hexadecimal

└ Los números binarios pueden tener hasta 32 dígitos; los hexadecimales, un máximo de

Sin un prefijo, el *entero1* es tratado como decimal (base 10). El resultado aparece en forma binaria, independientemente del estado del modo Base.

Si se introduce un entero decimal demasiado grande para una forma binaria de 32 bits con su correspondiente signo, se emplea una operación de módulos simétricos para llevar el valor al rango apropiado.

256►Bin **ENTER** 0b100000000
0h1F►Bin **ENTER** 0b11111

BldData CATALOG

BldData [dataVar]

Crea la variable de datos *dataVar* basándose en la información utilizada para representar la gráfica actual. **BldData** es válida en todos los modos de representación gráfica.

Si se omite *dataVar*, los datos se almacenan en la variable *sysData* del sistema.

Nota: Cuando se inicie por primera vez el Data/Matrix Editor después de utilizar **BldData**, *dataVar* o *sysData* (según el argumento utilizado con **BldData**) se establece como la variable de datos actual.

Los valores de incremento empleados para cualquier variable independiente (*x* en el ejemplo de la derecha) se calculan de acuerdo con los valores de las variables de la ventana.

Para más información sobre los incrementos utilizados para obtener una gráfica, consulte el capítulo de este manual, en el que se describe dicho modo de representación gráfica.

El modo 3D tiene dos variables independientes. En los datos de ejemplo de la derecha, observe que *x* permanece constante a medida que *y* se incrementa en su rango de valores.

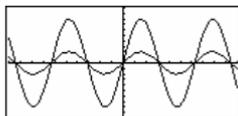
A continuación, *x* se incrementa a su siguiente valor *e* y se incrementa de nuevo en su rango. Este modelo continúa hasta que *x* se ha incrementado en su rango.

En el modo de representación de funciones y en el modo Angle en radianes:

$8 \cdot \sin(x) \rightarrow y_1(x)$ [ENTER] Done

$2 \cdot \sin(x) \rightarrow y_2(x)$ [ENTER] Done

ZoomStd [ENTER]



[HOME]

[CALC HOME]

BldData [ENTER] Done

APPS] 6 [ENTER]

DATA	x	y1	y2
	c1	c2	c3
1	-10.	4.3522	1.0888
2	-9.8323	3.168	1.792
3	-9.6641	1.8945	4.7363
4	-9.496	5.6769	1.1412

Nota: Los siguientes datos de ejemplo pertenecen a una gráfica 3D.

DATA	x	y	z1
	c1	c2	c3
1	-10.	-10.	0.
2	-10.	-8.5715	5.8309
3	-10.	-7.1438	9.9706
4	-10.	-5.7149	9.8677

ceiling() Menú MATH/Number

ceiling(*expresión*) \Rightarrow entero

Devuelve el entero más próximo que sea \geq que el argumento.

El argumento puede ser un número real o complejo.

Nota: Consulte además **floor**().

ceiling(*lista*) \Rightarrow lista

ceiling(*matriz*) \Rightarrow matriz

Devuelve una lista o matriz con el entero superior más próximo a cada elemento.

ceiling(0.456) [ENTER]

1.

ceiling({-3.1,1.2,5}) [ENTER]

{-3. 1 3.}

ceiling([0. -3.2; 1.3,4]) [ENTER]

$\begin{bmatrix} 0 & -3. \\ 2. & 4 \end{bmatrix}$

cFactor() Menú MATH/Algebra/Complex

cFactor(*expresión1*, *var1*) ⇒ *expresión*

cFactor(*lista1*, *var1*) ⇒ *lista*

cFactor(*matriz1*, *var1*) ⇒ *matriz*

cFactor(*expresión1*) devuelve la *expresión1*, factorizada respecto a todas las variables, sobre un común denominador.

La *expresión1* se descompone todo lo posible en factores racionales lineales, aunque con ello aparezcan otros números no reales. Esta alternativa es útil si se desea factorizar respecto a más de una variable.

cFactor($a^3 \cdot x^2 + a \cdot x^2 + a^3 + a$) **ENTER**

cFactor($x^2 + 4/9$) **ENTER**

cFactor($x^2 + 3$) **ENTER**

$x^2 + 3$

cFactor($x^2 + a$) **ENTER**

$x^2 + a$

cFactor(*expresión1*, *var*) devuelve la *expresión1* factorizada respecto a la variable *var*.

La *expresión1* se descompone todo lo posible en factores que sean lineales en *var*, con constantes no reales, aunque esto introduzca constantes irracionales o subexpresiones que son irracionales en otras variables.

Los factores y sus términos se clasifican utilizando *var* como la variable principal. Las potencias similares en *var* se agrupan en cada factor. Incluya *var* si necesita una factorización sólo respecto a esta variable, y si puede aceptar expresiones irracionales en otras para incrementar la factorización respecto a *var*. Puede haber factorización respecto a otras variables.

En el estado AUTO del modo Exact/Approx, al incluir *var* también se permiten aproximaciones de coma flotante cuando los coeficientes irracionales no se pueden expresar de manera concisa y explícita con las funciones incorporadas. Incluso cuando hay una sola variable, al incluir *var* puede calcularse una factorización más completa.

Nota: Consulte además **factor()**.

cFactor($a^3 \cdot x^2 + a \cdot x^2 + a^3 + a, x$) **ENTER**

$a \cdot (a^2 + 1) \cdot (x + i) \cdot (x + i)$

cFactor($x^2 + 3, x$) **ENTER**

$(x + \sqrt{3} \cdot i) \cdot (x + \sqrt{3} \cdot i)$

cFactor($x^2 + a, x$) **ENTER**

$(x + \sqrt{a} \cdot i) \cdot (x + \sqrt{a} \cdot i)$

cFactor($x^5 + 4x^4 + 5x^3 - 6x - 3$) **ENTER**

$x^5 + 4 \cdot x^4 + 5 \cdot x^3 - 6 \cdot x - 3$

cFactor(**ans**(1), *x*) **ENTER**

$(x - .965) \cdot (x + .612) \cdot (x + 2.13) \cdot$

$(x + 1.11 - 1.07 \cdot i) \cdot$

$(x + 1.11 + 1.07 \cdot i)$

char() Menú MATH/String

char(*entero*) ⇒ *carácter*

Devuelve una cadena de caracteres que contiene el *carácter* correspondiente al *entero* en el conjunto de caracteres de la TI-89 Titanium / Voyage™ 200. Consulte el anexo B para una lista completa de los códigos de caracteres.

El rango válido para *entero* es 0–255.

char(38) **ENTER**

"&"

char(65) **ENTER**

"A"

checkTmr() CATALOG

checkTmr(*tiempoinic*) ⇒ *entero*

Devuelve un entero que representa el número de segundos transcurridos desde que se inició un temporizador. *tiempoinic* es un entero devuelto por la función **startTmr()**.

También se puede utilizar una matriz de enteros de *tiempoinic*. Los enteros de *tiempoinic* deben estar entre 0 y la hora actual del reloj.

Es posible ejecutar varios temporizadores a la vez.

Nota: Consulte también **startTmr()** y **timeCnv()**.

startTmr() [ENTER] 148083315

checkTmr(148083315) 34

startTmr()>Tempor1

⋮

startTmr()>Tempor2

⋮

checkTmr(Tempor1)>ValorTempor1

⋮

checkTmr(Tempor2)>ValorTempor2

Circle CATALOG

Circle *x*, *y*, *r* [, *modoDraw*]

Dibuja una circunferencia con su centro en las coordenadas (*x*, *y*) y con un radio *r*.

x, *y*, y *r* deben ser valores reales.

Si *modoDraw*= 1, dibuja la circunferencia (por omisión).

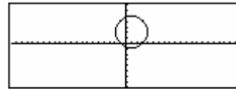
Si *modoDraw*= 0, desactiva la circunferencia.

Si *modoDraw*= -1, invierte los pixels de la circunferencia.

Nota: Al repetir la representación gráfica (Regraph), se borran todos los elementos dibujados. Consulte además **PxlCrcl**.

En una ventana de visualización ZoomSqr:

ZoomSqr:Circle 1.2.3 [ENTER]



ClockOff CATALOG

ClockOff

Desactiva el reloj.

ClockOn CATALOG

ClockOn

Activa el reloj.

ClrDraw CATALOG

ClrDraw

Vacía la pantalla Graph y reinicia la función Smart Graph, para que se dibuje otra vez la gráfica al mostrar la pantalla Graph.

Mientras visualiza la pantalla Graph, puede borrar todos los elementos dibujados (como rectas y puntos) si pulsa:

 [2nd] [F6]

 [F6]

y selecciona 1:ClrDraw.

ClrErr CATALOG

ClrErr

Anula un estado de error. Ajusta *errornum* en cero y borra las variables internas de error de contexto.

En el programa, la cláusula **Else** de **Try...EndTry** debe utilizar **ClrErr** o **PassErr**. Si se va a procesar o ignorar el error, utilice **ClrErr**. Si no sabe cómo tratar el error, envíelo al siguiente gestor de errores con **PassErr**. Si no hay más gestores pendientes **Try...EndTry**, se muestra el recuadro de diálogo de errores de la forma usual.

Nota: Consulte además **PassErr** y **Try**.

Listado del programa:

```
:clearerr()
:Prgm
:PlotsOff:FnOff:ZoomStd
:For i,0,238
: $\Delta x * i + x_{min} \rightarrow x_{cord}$ 
: Try
: PtOn xcord,ln(xcord)
: Else
: If errornum=800 or
                                errornum=260 Then
:   ClrErr  $\bullet$  clear the error
: Else
:   PassErr  $\bullet$  pass on any other error
: EndIf
: EndTry
: EndFor
: EndPrgm
```

ClrGraph CATALOG

ClrGraph

Borra las gráficas de funciones o expresiones dibujadas mediante la orden **Graph** o creadas con la orden **Table** (consulte **Graph** o **Table**).

Las funciones $Y=$ seleccionadas previamente se representarán la próxima vez que se pasa a la pantalla gráfica.

ClrHome CATALOG

ClrHome

Borra todos los elementos, tanto de entrada (**entry ()**) como de respuesta (**ans ()**), almacenados en el área de historia de la pantalla Home. No vacía la línea de entrada actual.

Mientras visualiza la pantalla Home, puede vaciar el área de historia si pulsa **[F1]** y selecciona 8:Clear Home.

En funciones tales como **solve()** que devuelven constantes o enteros (**@1**, **@2**, etc.) arbitrarios, **ClrHome** reinicia el sufijo a 1.

ClrIO CATALOG

ClrIO

Vacía la pantalla Program I/O.

ClrTable CATALOG

ClrTable

Borra todos los valores de una tabla. Sólo puede aplicarse en el estado ASK del recuadro de diálogo Table Setup.

Mientras visualiza la pantalla Table en el modo Ask, puede borrar los valores si pulsa $\boxed{F1}$ y selecciona 8: Clear Table.

colDim() Menú MATH/Matrix/Dimensions

colDim(matriz) \Rightarrow expresión

colDim([0.1.2:3.4.5]) $\boxed{\text{ENTER}}$

3

Devuelve el número de columnas que contiene una matriz.

Nota: Consulte además **rowDim()**.

colNorm() Menú MATH/Matrix/Norms

colNorm(matriz) \Rightarrow expresión

[1. -2.3:4.5. -6] \rightarrow mat $\boxed{\text{ENTER}}$

Devuelve el máximo de las sumas de los valores absolutos de los elementos de las columnas de matriz.

colNorm(mat) $\boxed{\text{ENTER}}$

$$\begin{bmatrix} 1 & -2 & 3 \\ 4 & 5 & -6 \end{bmatrix}$$

9

Nota: No se admiten los elementos de matriz no definidos. Consulte además **rowNorm()**.

comDenom() Menú MATH/Algebra

comDenom(expresión1, var) \Rightarrow expresión

comDenom(lista1, var) \Rightarrow lista

comDenom(matriz1, var) \Rightarrow matriz

comDenom(expresión1) devuelve la fracción reducida de un numerador y un denominador totalmente desarrollados.

comDenom(expresión1, var) devuelve la fracción reducida de un numerador y denominador desarrollados respecto a *var*. Los términos y sus factores se clasifican utilizando *var* como la variable principal. Se agrupan las potencias similares de *var*. Puede haber una factorización incidental de los coeficientes agrupados. En comparación con la omisión de *var*, esto ahorra tiempo, memoria y espacio en la pantalla, haciendo que la expresión sea más comprensible. También hace que las operaciones posteriores con el resultado sean más rápidas y no agoten toda la memoria.

Si no se utiliza *var* en la expresión1, **comDenom**(expresión1, var) devuelve una fracción reducida con un numerador no desarrollado y un denominador no desarrollado. Este resultado parcialmente factorizado ahorra incluso más tiempo, memoria y espacio en la pantalla. Dicho resultado hace que las operaciones con el mismo sean más rápidas y no agoten toda la memoria.

comDenom((y^2+y)/(x+1)^2+y^2+y) $\boxed{\text{ENTER}}$

$$\frac{x^2 \cdot y^2 + x^2 \cdot y + 2 \cdot x \cdot y^2 + 2}{x^2 + 2 \cdot x + 1}$$

comDenom((y^2+y)/(x+1)^2+y^2+y.x) $\boxed{\text{ENTER}}$

$$\frac{x^2 \cdot y \cdot (y+1) + 2 \cdot x \cdot y \cdot (y+1)}{x^2 + 2 \cdot x + 1}$$

comDenom((y^2+y)/(x+1)^2+y^2+y.y) $\boxed{\text{ENTER}}$

$$\frac{y^2 \cdot (x^2 + 2 \cdot x + 2) + y \cdot (x^2 + 2 \cdot x + 1)}{x^2 + 2 \cdot x + 1}$$

comDenom(exprn.abc) \rightarrow
comden(exprn) $\boxed{\text{ENTER}}$

Done

comden((y^2+y)/(x+1)^2+y^2+y) $\boxed{\text{ENTER}}$

$$\frac{y^2 \cdot (x^2 + 2 \cdot x + 2) \cdot y \cdot (y+1)}{(x+1)^2}$$

Incluso cuando no hay un denominador, la función **comden** es una forma rápida de obtener una factorización parcial si **factor()** es demasiado lenta o consume toda la memoria.

comden(1234x²*(y³-y)+2468x*(y²-1)) **ENTER**
 $1234 \cdot x \cdot (x \cdot y + 2) \cdot (y^2 - 1)$

Sugerencia: Introduzca una definición de esta función **comden()** y pruébela como alternativa de **comDenom()** y **factor()**.

conj() Menú MATH/Complex

conj(*expresión*) ⇒ *expresión*

conj(*lista*) ⇒ *lista*

conj(*matriz*) ⇒ *matriz*

Devuelve el número complejo conjugado del argumento.

Nota: Todas las variables no definidas se tratan como variables reales.

conj(1+2*i*) **ENTER** 1 - 2 • *i*

conj([2, 1-3*i*; -*i*, -7]) **ENTER**

$$\begin{bmatrix} 2 & 1+3 \cdot i \\ i & -7 \end{bmatrix}$$

conj(*z*) *z*

conj(*x*+*iy*) *x* + - *i* • *y*

CopyVar CATALOG

CopyVar *var1*, *var2*

Copia el contenido de la variable *var1* en *var2*. Si *var2* no existe, **CopyVar** la crea.

Nota: **CopyVar** es parecida a la instrucción de almacenamiento (⇒) cuando se copia una expresión, lista, matriz o cadena de caracteres, excepto que no se realiza ninguna simplificación al utilizarla. Debe utilizar **CopyVar** con una variable no algebraica como Pic o las variables GDB.

x+*y* ⇒ *a* **ENTER** *x* + *y*

10 ⇒ *x* **ENTER** 10

CopyVar *a*, *b* **ENTER** Done

a ⇒ *c* **ENTER** *y* + 10

DelVar *x* **ENTER** Done

b **ENTER** *x* + *y*

c **ENTER** *y* + 10

cos() Tecla [2nd] [COS] Tecla [COS]

cos(*expresión*) ⇒ *expresión*

cos(*lista*) ⇒ *lista*

cos(*expresión*) devuelve el coseno del argumento.

cos(*lista*) devuelve la lista de los cosenos de todos los elementos de la *lista*.

Nota: El argumento se interpreta como un ángulo expresado en grados, grados centesimales o radianes, según el valor de modo de ángulo actual. Puede utilizar °, ° o ° para anular temporalmente el modo de ángulo.

En el modo Angle, en grados:

cos((π/4)°) **ENTER** $\frac{\sqrt{2}}{2}$

cos(45) **ENTER** $\frac{\sqrt{2}}{2}$

cos({0.60,90}) **ENTER** {1 1/2 0}

En el modo Angle, en grados centesimales:

cos({0.50,100}) **ENTER** {1 $\frac{\sqrt{2}}{2}$ 0}

En el modo Angle, en radianes:

cos(π/4) **ENTER** $\frac{\sqrt{2}}{2}$

cos(45°) **ENTER** $\frac{\sqrt{2}}{2}$

cos(*Matriz cuadrada*) \Rightarrow *Matriz cuadrada*

Devuelve el coseno de *Matriz cuadrada*1. Esto *no* es lo mismo que calcular el coseno de cada elemento.

Cuando una función escalar $f(A)$ opera sobre *Matriz cuadrada*1 (A), el resultado se obtiene mediante el algoritmo:

1. Calcula los valores propios (λ_i) y vectores propios (V_i) de A .

*Matriz cuadrada*1 debe ser diagonalizable. Además, no puede tener variables simbólicas a las que no se haya asignado un valor.

2. Construye las matrices:

$$B = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & \lambda_n \end{bmatrix} \text{ y } X = [V_1, V_2, \dots, V_n]$$

3. A continuación, $A = X B X^{-1}$ y $f(A) = X f(B) X^{-1}$. Por ejemplo, $\cos(A) = X \cos(B) X^{-1}$ donde:

$$\cos(B) = \begin{bmatrix} \cos(\lambda_1) & 0 & \dots & 0 \\ 0 & \cos(\lambda_2) & \dots & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & \cos(\lambda_n) \end{bmatrix}$$

Todos los cálculos se realizan mediante aritmética de coma flotante.

En el modo Angle, en radianes:

$\cos([1.5, 3; 4, 2, 1; 6, -2, 1])$ **[ENTER]**

$$\begin{bmatrix} .212\dots & .205\dots & .121\dots \\ .160\dots & .259\dots & .037\dots \\ .248\dots & -.090\dots & .218\dots \end{bmatrix}$$

cos⁻¹()

Tecla **[COS⁻¹]**

Tecla **[2nd] [COS⁻¹]**

cos⁻¹(expresión) \Rightarrow *expresión*

cos⁻¹(lista) \Rightarrow *lista*

cos⁻¹(expresión) devuelve el ángulo cuyo coseno es *expresión*1.

cos⁻¹(lista) devuelve la lista de los ángulos de cosenos de los elementos de la *lista*1.

Nota: El argumento se interpreta como un ángulo expresado en grados, grados centesimales o radianes, según el valor de modo de ángulo actual. Puede utilizar $\frac{\circ}{\circ}$, $\frac{\circ}{\circ}$ o $\frac{\circ}{\circ}$ para anular temporalmente el modo de ángulo.

En el modo Angle, en grados:

$\cos^{-1}(1)$ **[ENTER]**

0

En el modo Angle, en grados centesimales:

$\cos^{-1}(0)$ **[ENTER]**

100

En el modo Angle, en radianes:

$\cos^{-1}(\{0, .2, .5\})$ **[ENTER]**

$$\left\{ \frac{\pi}{2}, 1.369\dots, 1.047\dots \right\}$$

cos⁻¹(Matriz cuadrada) \Rightarrow *Matriz cuadrada*

Devuelve el arccoseno de la matriz de *Matriz cuadrada*1. Esto *no* es lo mismo que calcular el arccoseno de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.

*Matriz cuadrada*1 debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

En el modo Angle en radianes, y en el modo de formato rectangular complejo:

$\cos^{-1}([1.5, 3; 4, 2, 1; 6, -2, 1])$ **[ENTER]**

$$\begin{bmatrix} 1.734\dots+.064\dots \cdot i & -1.490\dots+.2105\dots \cdot i & \dots \\ -.725\dots+1.515\dots \cdot i & .623\dots+.778\dots \cdot i & \dots \\ -2.083\dots+.2632\dots \cdot i & 1.790\dots-1.271\dots \cdot i & \dots \end{bmatrix}$$

cosh() Menú MATH/Hyperbolic**cosh**(*expresión*) ⇒ *expresión*cosh(1.2) **ENTER** 1.810...**cosh**(*lista*) ⇒ *lista*cosh({0,1,2}) **ENTER** {1 1.810...}**cosh** (*expresión*) devuelve el coseno hiperbólico del argumento.**cosh** (*lista*) devuelve una lista de los cosenos hiperbólicos de los elementos de la *lista*.**cosh**(*Matriz cuadrada*) ⇒ *Matriz cuadrada*

En el modo Angle, en radianes:

Devuelve el coseno hiperbólico de la *Matriz cuadrada*. Esto *no* es lo mismo que calcular el coseno hiperbólico de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.cosh([1.5,3:4,2,1:6, -2,1]) **ENTER***Matriz cuadrada* debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

421.255	253.909	216.905
327.635	255.301	202.958
226.297	216.623	167.628

cosh⁻¹() Menú MATH/Hyperbolic**cosh⁻¹**(*expresión*) ⇒ *expresión*cosh⁻¹(1) **ENTER** 0**cosh⁻¹**(*lista*) ⇒ *lista*cosh⁻¹({1,2,1,3}) **ENTER**
{0 1.372... cosh⁻¹(3)}**cosh⁻¹** (*expresión*) devuelve el coseno hiperbólico inverso del argumento.**cosh⁻¹** (*lista*) devuelve una lista con los cosenos hiperbólicos inversos de cada elemento de la *lista*.**cosh⁻¹**(*Matriz cuadrada*) ⇒ *Matriz cuadrada*

En el modo Angle, en radianes, y en el modo de formato rectangular complejo:

Devuelve el coseno hiperbólico inverso de la *Matriz cuadrada*. Esto *no* es lo mismo que calcular el coseno hiperbólico inverso de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.cosh⁻¹([1.5,3:4,2,1:6, -2,1]) **ENTER***Matriz cuadrada* debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

2.525...+1.734...·i	-.009...-1.490...·i	...
.486...-.725...·i	1.662...+623...·i	...
-.322...-2.083...·i	1.267...+1.790...·i	...

cot() Menú MATH/Trig (MATEMÁTICAS/Trigonometría)**cot**(*expresión*) ⇒ *expresión*

En el modo Angle, en grados:

cot(*lista*) ⇒ *lista*cot(45) **ENTER** 1Devuelve la cotangente de *expresión* o una lista de las cotangentes de todos los elementos de *lista*.

En el modo Angle, en grados centesimales:

Nota: El resultado devuelve un ángulo expresado en grados, grados centesimales o radianes, según el valor de modo de ángulo actual.cot(50) **ENTER** 1

En el modo Angle, en radianes:

cot({1,2,1,3}) **ENTER**

$$\frac{1}{\tan(1)} - .584... \frac{1}{\tan(3)}$$

cot⁻¹() Menú MATH/Trig (MATEMÁTICAS/Trigonometría)**cot⁻¹(expresión1)** ⇒ expresión**cot⁻¹(lista1)** ⇒ lista

Devuelve el ángulo cuya cotangente es *expresión1* o una lista de las cotangentes inversas de todos los elementos de *lista1*.

Nota: El resultado devuelve un ángulo expresado en grados, grados centesimales o radianes, según el valor de modo de ángulo actual.

En el modo Angle, en grados:

cot⁻¹(1) [ENTER] 45

En el modo Angle, en grados centesimales:

cot⁻¹(1) [ENTER] 50

En el modo Angle, en radianes:

cot⁻¹(1) [ENTER] $\frac{\pi}{4}$ **coth()** Menú MATH/Hyperbolic (MATEMÁTICAS/Hiperbólicas)**coth(expresión1)** ⇒ expresión**coth(lista1)** ⇒ lista

Devuelve la cotangente hiperbólica de *expresión1* o una lista de las cotangentes hiperbólicas de todos los elementos de *lista1*.

coth(1.2) [ENTER] 1.199...

coth({1,3.2}) [ENTER]

 $\frac{1}{\tanh(1)}$ 1.003...**coth⁻¹()** Menú MATH/Hyperbolic (MATEMÁTICAS/Hiperbólicas)**coth⁻¹(expresión1)** ⇒ expresión**coth⁻¹(lista1)** ⇒ lista

Devuelve la cotangente hiperbólica inversa de *expresión1* o una lista de las cotangentes hiperbólicas inversas de todos los elementos de *lista1*.

coth⁻¹(3.5) [ENTER] .293...coth⁻¹({-2,2,1.6}) [ENTER] $\frac{-\ln(3)}{2}$.518... $\frac{\ln(7/5)}{2}$ **crossP()** Menú MATH/Matrix/Vector ops**crossP(lista1, lista2)** ⇒ lista

Devuelve la lista formada por el producto vectorial de la *lista1* y la *lista2*.

La *lista1* y la *lista2* deben tener la misma dimensión, que debe ser 2 o 3.

crossP({a1,b1},{a2,b2}) [ENTER]
 {0 0 a1•b2-a2•b1}
 crossP({0,1,2,2,-5},{1,-.5,0})
 [ENTER]
 {-2.5 -5. -2.25}

crossP(vector1, vector2) ⇒ vector

Devuelve un vector fila o columna (dependiendo de los argumentos) que es el producto vectorial de *vector1* y *vector2*.

Tanto el *vector1* como el *vector2* deben ser ambos vectores fila o columna. Ambos vectores deben tener la misma dimensión, que debe ser 2 o 3.

crossP([1,2,3],[4,5,6]) [ENTER]
 [-3 6 -3]
 crossP([1,2],[3,4]) [ENTER]
 [0 0 -2]

csc() Menú MATH/Trig (MATEMÁTICAS/Trigonometría) $\text{csc}(\text{expresión1}) \Rightarrow \text{expresión}$ $\text{csc}(\text{lista1}) \Rightarrow \text{lista}$

Devuelve la cosecante de *expresión1* o una lista de las cosecantes de todos los elementos de *lista1*.

En el modo Angle, en grados:

 $\text{csc}(45) \text{ [ENTER]}$ $\sqrt{2}$

En el modo Angle, en grados centesimales:

 $\text{csc}(50) \text{ [ENTER]}$ $\sqrt{2}$

En el modo Angle, en radianes:

 $\text{csc}(\{1, \pi/2, \pi/3\}) \text{ [ENTER]}$
 $\frac{1}{\sin(1)} \quad 1 \quad \frac{2\sqrt{3}}{3}$ **csc⁻¹()** Menú MATH/Trig (MATEMÁTICAS/Trigonometría) $\text{csc}^{-1}(\text{expresión1}) \Rightarrow \text{expresión}$ $\text{csc}^{-1}(\text{lista1}) \Rightarrow \text{lista}$

Devuelve el ángulo cuya cosecante es *expresión1* o una lista de las cosecantes inversas de todos los elementos de *lista1*.

Nota: El resultado devuelve en ángulo expresado en grados, grados centesimales o radianes, según el valor de modo de ángulo actual.

En el modo Angle, en grados:

 $\text{csc}^{-1}(1) \text{ [ENTER]}$ 90

En el modo Angle, en grados centesimales:

 $\text{csc}^{-1}(1) \text{ [ENTER]}$ 100

En el modo Angle, en radianes:

 $\text{csc}^{-1}(\{1, 4, 6\}) \text{ [ENTER]}$
 $\frac{\pi}{2} \sin^{-1}(1/4) \sin^{-1}(1/6)$ **csch()** Menú MATH/Hyperbolic (MATEMÁTICAS/Hiperbólicas) $\text{csch}(\text{expresión1}) \Rightarrow \text{expresión}$ $\text{csch}(\text{lista1}) \Rightarrow \text{lista}$

Devuelve la cosecante hiperbólica de *expresión1* o una lista de las cosecantes hiperbólicas de todos los elementos de *lista1*.

 $\text{csch}(3) \text{ [ENTER]}$ $\frac{1}{\sinh(3)}$ $\text{csch}(\{1, 2, 1, 4\}) \text{ [ENTER]}$
 $\frac{1}{\sinh(1)} \quad .248\dots \quad \frac{1}{\sinh(4)}$ **csch⁻¹()** Menú MATH/Hyperbolic (MATEMÁTICAS/Hiperbólicas) $\text{csch}^{-1}(\text{expresión1}) \Rightarrow \text{expresión}$ $\text{csch}^{-1}(\text{lista1}) \Rightarrow \text{lista}$

Devuelve la cosecante hiperbólica inversa de *expresión1* o una lista de las cosecantes hiperbólicas inversas de todos los elementos de *lista1*.

 $\text{csch}^{-1}(1) \text{ [ENTER]}$ $\sinh^{-1}(1)$ $\text{csch}^{-1}(\{1, 2, 1, 3\}) \text{ [ENTER]}$
 $\sinh^{-1}(1) \quad .459\dots \quad \sinh^{-1}(1/3)$

cSolve() Menú MATH/Algebra/Complex

cSolve(ecuación, var) \Rightarrow expresión booleana

Devuelve posibles soluciones complejas para *var* de una ecuación. El objetivo es obtener todas las posibles soluciones, tanto reales como no reales. Aunque la *ecuación* sea real, **cSolve()** permite obtener resultados no reales.

Aunque la TI-89 Titanium / Voyage™ 200 procesa todas las variables no definidas como si fueran reales, **cSolve()** puede resolver ecuaciones polinómicas con soluciones complejas.

cSolve() establece temporalmente el dominio complejo al hallar la solución, incluso si el dominio actual es real. En el dominio complejo, las potencias fraccionarias con denominadores impares utilizan la solución principal en vez de la real. En consecuencia, las soluciones con **solve()** de ecuaciones con estas potencias fraccionarias no son, necesariamente, un subconjunto de las soluciones con **cSolve()**.

cSolve() comienza con operaciones simbólicas exactas. Excepto en el modo EXACT, **cSolve()** también utiliza, si es necesario, la factorización iterativa aproximada de polinomios complejos.

Nota: Consulte además **cZeros()**, **solve()** y **zeros()**.

Nota: Si *ecuación* no es un polinomio con funciones tales como **abs()**, **angle()**, **conj()**, **real()** o **imag()**, al final de *var* debe colocarse un guión de subrayado $_$ (\square \square [-], \square \square [2nd] [-]). Por omisión, una variable se trata como un valor real. Si se utiliza *var_*, la variable se trata como compleja.

También debe emplearse *var_* para cualquier otra variable de *ecuación* que pueda tener valores no reales. De no hacerlo, pueden obtenerse resultados imprevistos.

cSolve(ecuación1 and ecuación2 [and ...],
{var0Aproximación1, var0Aproximación2 [, ...]})
 \Rightarrow expresión booleana

Devuelve posibles soluciones complejas de un sistema de ecuaciones, donde cada *var0Aproximación* especifica una variable que se desea resolver.

De forma opcional, puede especificarse una aproximación inicial para una variable. Cada *var0Aproximación* debe tener la forma:

variable
- 0 -
variable = número real o no real

Por ejemplo, *x* es válido, lo mismo que *x=3+i*.

Si todas las ecuaciones son polinómicas y NO se desea especificar ninguna aproximación inicial, **cSolve()** utiliza el método de eliminación léxica de Gröbner/Buchberger para intentar determinar **todas** las soluciones complejas.

cSolve($x^3 = -1, x$) [ENTER]
solve($x^3 = -1, x$) [ENTER]

\square **cSolve**($x^3 = -1, x$)
 \square $1/2 + \frac{\sqrt{3}}{2} \cdot i$ or $x = 1/2 - \frac{\sqrt{3}}{2} \cdot i$
 \square **solve**($x^3 = -1, x$) $x = -1$

cSolve($x^{1/3} = -1, x$) [ENTER] false
solve($x^{1/3} = -1, x$) [ENTER] $x = -1$

Modo Display Digits en **Fix 2:**

exact(**cSolve**($x^5 + 4x^4 + 5x^3 - 6x - 3 = 0, x$)) [ENTER]

cSolve(**ans**(1), *x*) [ENTER]

\square **exact**(**cSolve**($x^5 + 4 \cdot x^4 + 5 \cdot x^3 - 6 \cdot x - 3 = 0, x$))
 \square $x \cdot (x^4 + 4 \cdot x^3 + 5 \cdot x^2 - 6) = 3$
 \square **cSolve**($x \cdot (x^4 + 4 \cdot x^3 + 5 \cdot x^2 - 6) = 3, x$)
 \square $x = -1.1138 + 1.07314 \cdot i$ or

z se trata como real:

cSolve(**conj**(*z*)=1+i, *z*) [ENTER] $z = 1 + i$

z_ se trata como compleja:

cSolve(**conj**(*z_*)=1+i, *z_*) [ENTER] $z_ = 1 - i$

Las soluciones complejas pueden incluir tanto soluciones reales como no reales, como en el ejemplo de la derecha.

`cSolve(u_*v_-u_=v_and
v_^2=-u_.{u_.v_})` **[ENTER]**

$$u_ = 1/2 + \frac{\sqrt{3}}{2} \cdot i \text{ and } v_ = 1/2 - \frac{\sqrt{3}}{2} \cdot i$$

$$\text{or } u_ = 1/2 - \frac{\sqrt{3}}{2} \cdot i \text{ and } v_ = 1/2 + \frac{\sqrt{3}}{2} \cdot i$$

$$\text{or } u_ = 0 \text{ and } v_ = 0$$

Los sistemas de ecuaciones *polinómicas* pueden tener variables extra que no tengan valores, pero representen valores numéricos dados que puedan sustituirse más adelante.

`cSolve(u_*v_-u_=c_*v_and
v_^2=-u_.{u_.v_})` **[ENTER]**

$$u_ = \frac{-(\sqrt{1-4 \cdot c} + 1)}{4} \text{ and } v_ = \frac{\sqrt{1-4 \cdot c} + 1}{2}$$

or

$$u_ = \frac{-(\sqrt{1-4 \cdot c} - 1)}{4} \text{ and } v_ = \frac{-(\sqrt{1-4 \cdot c} - 1)}{2}$$

$$\text{or } u_ = 0 \text{ and } v_ = 0$$

También es posible incluir variables solución que no aparecen en las ecuaciones. Estas soluciones muestran cómo las familias de soluciones pueden contener constantes arbitrarias de la forma @k, donde k es un parámetro entero comprendido entre 1 y 255. El parámetro se pone en 1 al utilizarse **CirHome** o **[F1] 8:Clear Home**.

`cSolve(u_*v_-u_=v_and
v_^2=-u_.{u_.v_.w_})` **[ENTER]**

$$u_ = 1/2 + \frac{\sqrt{3}}{2} \cdot i \text{ and } v_ = 1/2 - \frac{\sqrt{3}}{2} \cdot i$$

$$\text{and } w_ = @1$$

or

$$u_ = 1/2 - \frac{\sqrt{3}}{2} \cdot i \text{ and } v_ = 1/2 + \frac{\sqrt{3}}{2} \cdot i$$

$$\text{and } w_ = @1$$

$$\text{or } u_ = 0 \text{ and } v_ = 0 \text{ and } w_ = @1$$

Para sistemas de polinomios, el tiempo de cálculo y el consumo de la memoria dependen en gran medida del orden en que se listen las variables solución. Si la opción inicial consume la memoria o su paciencia, intente reordenar las variables en las ecuaciones y en el listado *varOAproximación*.

Si no se incluye ninguna aproximación y ninguna ecuación es polinómica en cualquier variable pero todas las ecuaciones son lineales en todas las variables solución, **cSolve()** utiliza la eliminación gaussiana para intentar determinar todas las soluciones.

`cSolve(u_+v_=e^(w_) and u_-v_=i.
{u_.v_})` **[ENTER]**

$$u_ = \frac{e^w}{2} + 1/2 \cdot i \text{ and } v_ = \frac{e^w - i}{2}$$

Si un sistema no es polinómico en todas sus variables ni lineal en sus variables solución, **cSolve()** determina a lo sumo una solución mediante un método iterativo aproximado. Para ello, el número de variables solución debe ser idéntico al número de ecuaciones, y todas las demás variables de las ecuaciones deben simplificarse a números.

`cSolve(e^(z_)=w_and w_=z_^2, {w_.z_})` **[ENTER]**

$$w_ = .494... \text{ and } z_ = -.703...$$

A menudo es necesaria una aproximación no real para determinar una solución no real. Por convergencia, una aproximación puede que tenga que ser bastante cercana a una solución.

`cSolve(e^(z_)=w_and w_=z_^2,
{w_.z_=1+i})` **[ENTER]**

$$w_ = .149... + 4.891... \cdot i \text{ and } z_ = 1.588... + 1.540... \cdot i$$

CubicReg Menú MATH/Statistics/Regressions

CubicReg *lista1*, *lista2*, [*lista3*], [*lista4*, *lista5*]

Calcula la regresión polinómica de tercer grado y actualiza todas las variables estadísticas.

Todas las listas deben tener el mismo tamaño, excepto la *lista5*.

La *lista1* representa *x*lista.

La *lista2* representa *y*lista.

La *lista3* representa la frecuencia.

La *lista4* representa los códigos de categoría.

La *lista5* representa la lista de categorías.

Nota: Desde la *lista1* hasta la *lista4* deben ser nombres de variable o c1–c99 (las columnas de la última variable de datos mostrada en Data/Matrix Editor). La *lista5* no tiene que ser un nombre de variable y no puede ser c1–c99.

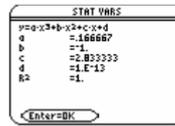
En el modo de gráficas de función:

{0.1.2.3} → L1 [ENTER] {0 1 2 3}

{0.2.3.4} → L2 [ENTER] {0 2 3 4}

CubicReg L1,L2 [ENTER] Done

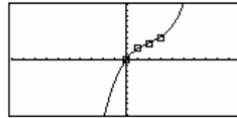
ShowStat [ENTER]



[ENTER] regeq(x) → y1(x) [ENTER] Done

NewPlot 1,1,L1,L2 [ENTER] Done

[GRAPH]



cumSum() Menú MATH/List

cumSum(*lista1*) ⇒ *lista*

cumSum({1.2.3.4}) [ENTER] {1 3 6 10}

Devuelve una *lista* de las sumas acumuladas de los elementos en la *lista1*, empezando por el elemento 1.

cumSum(*matriz1*) ⇒ *matriz*

Devuelve una *matriz* de las sumas acumuladas de los elementos en *matriz1*. Cada elemento es la suma acumulada de la columna, desde arriba hacia abajo.

[1.2.3.4:5.6] → m1 [ENTER]

cumSum(m1) [ENTER]

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \\ 1 & 2 \\ 4 & 6 \\ 9 & 12 \end{bmatrix}$$

CustmOff CATALOG

CustmOff

Suprime una barra de herramientas personalizada.

CustmOn y **CustmOff** permiten a un programa controlar una barra de herramientas personalizada. De forma manual, se puede pulsar **[2nd] [CUSTOM]** para activar y desactivar una barra de herramientas personalizada. Además, una barra de herramientas personalizada se suprime de forma automática al cambiar de aplicación.

Vea el ejemplo de listado del programa **Custom**.

CustmOn CATALOG

CustmOn

Activa una barra de herramientas personalizada que ya se haya configurado en un bloque **Custom...EndCustm**.

CustmOn y **CustmOff** activan un programa para controlar una barra de herramientas personalizada. De forma manual, se puede pulsar **[2nd] [CUSTOM]** para activar y desactivar una barra de herramientas personalizada.

Vea el ejemplo de listado del programa **Custom**.

Custom Tecla **[2nd] [CUSTOM]**

Custom

bloque

EndCustm

Configura una barra de herramientas que se activa al pulsar **[2nd] [CUSTOM]**. Es muy similar a la instrucción **ToolBar**, excepto que los enunciados Title e Item no pueden tener etiquetas.

bloque puede ser un único enunciado o una serie de enunciados separados con el carácter “.”.

Nota: **[2nd] [CUSTOM]** actúa como un conmutador. La primera vez llama al menú y la segunda vez lo cierra. El menú también se cierra cuando se cambia de aplicación.

Listado del programa:

```
:Test()
:Prgm
:Custom
:Title           "Lists"
:Item            "List1"
:Item            "Scores"
:Item            "L3"
:Item            "Fractions"
:Item            "f(x)"
:Item            "h(x)"
:Title           "Graph"
:EndCustm
:EndPrgm
```

Cycle CATALOG

Cycle

Transfiere el control del programa justo a la siguiente iteración del bucle actual (**For**, **While** o **Loop**).

Cycle únicamente está permitida en las tres estructuras de bucle (**For**, **While** o **Loop**).

Listado del programa:

```
:● Sum the integers from 1 to
                                100 skipping 50.
:0→temp
:For i,1,100,1
:If i=50
:Cycle
:temp+i→temp
:EndFor
:Disp temp
```

Contenido de temp después de la ejecución:50

CyclePic CATALOG

CyclePic Cadena de nombre *pic*, *n*, [*espera*], [*ciclos*], [*dirección*]

Muestra todas las variables PIC especificadas y en el intervalo especificado. El usuario tiene un control opcional del tiempo entre cada imagen, el número de veces que pasa por las imágenes y la dirección en que se mueve, circularmente o avanzando y retrocediendo.

El valor de *dirección* es 1 para moverse circularmente y -1 para avanzar y retroceder. Por omisión = 1.

1. Guarde tres imágenes con el nombre **1**
2. **Introduzca:** CyclePic "pic".3..5.4..-1
3. Las tres imágenes (3) se presentan :

►Cylind Menú MATH/Matrix/Vector ops

vector►Cylind

[2.2.3] ►Cylind **[ENTER]**

Muestra un vector-fila o columna con forma cilíndrica [*r*, $\angle\theta$, *z*].

$$[2 \cdot \sqrt{2} \angle \frac{\pi}{4} \quad 3]$$

El *vector* debe tener exactamente tres elementos. Puede ser una fila o una columna.

cZeros() Menú MATH/Algebra/Complex

cZeros(*expresión*, *var*) \Rightarrow *lista*

Modo Display Digits en **Fix 3**:

Devuelve la lista de posibles valores, tanto reales como no reales, de *var* que hacen *expresión*=0. **cZeros()** lo hace operando **►list(cSolve**(*expresión*=0, *var*), *var*). De lo contrario, **cZeros()** es similar a **zeros()**.

$$\begin{aligned} & \text{cZeros}(x^5+4x^4+5x^3-6x-3,x) \text{ [ENTER]} \\ & \{-2.125 \quad -.612 \quad .965 \\ & \quad -1.114 - 1.073 \cdot i \\ & \quad -1.114 + 1.073 \cdot i\} \end{aligned}$$

Nota: Consulte además **cSolve()**, **solve()** y **zeros()**.

z se considera como real:

Nota: Si *expresión* no es un polinomio con funciones tales como **abs()**, **angle()**, **conj()**, **real()** o **imag()**, debe colocarse un guión de subrayado $_$ ([] , [] , [] , []) al final de *var*. Por omisión, una variable se considera como un valor real. Si se utiliza *var_*, la variable se considera como compleja.

$$\text{cZeros}(\text{conj}(z) - 1 - i, z) \text{ [ENTER]} \quad \{1+i\}$$

z_ se considera como compleja:

También debe utilizarse *var_* para todas las demás variables en *expresión* que puedan tener valores no reales. De no hacerse, es posible obtener resultados imprevistos.

$$\text{cZeros}(\text{conj}(z_) - 1 - i, z_) \text{ [ENTER]} \quad \{1-i\}$$

cZeros({*expresión*1, *expresión*2 [, ...]}, {*var*0Aproximación1, *var*0Aproximación2 [, ...]}) \Rightarrow *matriz*

Devuelve las posibles posiciones donde las expresiones son cero simultáneamente. Cada *var*0Aproximación especifica una incógnita cuyo valor se desea hallar.

De forma opcional, puede especificarse una aproximación inicial para una variable. Cada *var*0Aproximación debe tener la forma:

variable

— 0 —

variable = número real o no real

Por ejemplo, x es válido, lo mismo que $x=3+i$.

Si todas las expresiones son polinómicas y NO específica ninguna aproximación inicial, **cZeros()** utiliza el método de eliminación léxica de Gröbner/Buchberger para intentar determinar **todas** las raíces complejas.

Las raíces complejas pueden incluir tanto raíces reales como no reales, como en el ejemplo de la derecha.

Cada fila de la matriz resultante representa una raíz alternativa, con los componentes ordenados de forma similar al listado de *varOAproximación*. Para extraer una fila, debe indexarse la matriz por [fila].

Un sistema polinomial puede tener variables extra que no tengan valores, pero representan valores numéricos dados que puedan sustituirse más adelante.

También es posible incluir incógnitas que no aparezcan en las expresiones. Estas raíces muestran cómo las familias de raíces pueden contener constantes arbitrarias de la forma @ k , donde k es un sufijo entero comprendido entre 1 y 255. Este parámetro toma el valor 1 al utilizar **ClrHome** o [F1] 8:Clear Home.

Para sistemas de polinomios, el tiempo de cálculo y el consumo de la memoria dependen en gran medida del orden en que se listen las incógnitas. Si la opción inicial consume la memoria o su paciencia, intente reordenar las variables en las expresiones y en la lista de *varOAproximación*.

Si no se incluye ninguna aproximación y si todas las expresiones son no polinómicas en cualquier variable pero todas las expresiones son lineales en todas las incógnitas, **cZeros()** utiliza la eliminación gaussiana para intentar determinar todas las raíces.

Si un sistema no es polinómico en todas sus variables ni lineal en sus incógnitas, **cZeros()** determina a lo sumo una raíz mediante un método iterativo aproximado. Para ello, el número de incógnitas debe ser igual al número de expresiones, y todas las demás variables en las expresiones deben simplificarse a números.

A menudo es necesaria una aproximación no real para determinar una raíz no real. Por convergencia, una aproximación puede que tenga que ser bastante cercana a una raíz.

Nota: Los siguientes ejemplos utilizan un guión de subrayado $_$ [] [-], [2nd] [-]) para que las variables sean consideradas como complejas.

`cZeros({u_*v_-u_-v_-v_-^2+u_},
{u_-v_-})` [ENTER]

$$\begin{bmatrix} 1/2 - \frac{\sqrt{3}}{2} \cdot i & 1/2 + \frac{\sqrt{3}}{2} \cdot i \\ 1/2 + \frac{\sqrt{3}}{2} \cdot i & 1/2 - \frac{\sqrt{3}}{2} \cdot i \\ 0 & 0 \end{bmatrix}$$

Extraer fila 2:

`ans(1)[2]` [ENTER]

$$\left[1/2 + \cdot i \quad 1/2 - \frac{\sqrt{3}}{2} \cdot i \right]$$

`cZeros({u_*v_-u_- (c_*v_-).v_-^2+u_},
{u_-v_-})` [ENTER]

$$\begin{bmatrix} \frac{-\sqrt{1-4 \cdot c} + 1}{4} & \frac{\sqrt{1-4 \cdot c} + 1}{2} \\ \frac{-\sqrt{1-4 \cdot c} - 1}{4} & \frac{-\sqrt{1-4 \cdot c} - 1}{2} \\ 0 & 0 \end{bmatrix}$$

`cZeros({u_*v_-u_-v_-v_-^2+u_},
{u_-v_-w_-})` [ENTER]

$$\begin{bmatrix} 1/2 - \frac{\sqrt{3}}{2} \cdot i & 1/2 + \frac{\sqrt{3}}{2} \cdot i & @1 \\ 1/2 + \frac{\sqrt{3}}{2} \cdot i & 1/2 - \frac{\sqrt{3}}{2} \cdot i & @1 \\ 0 & 0 & @1 \end{bmatrix}$$

`cZeros({u+v_-e^(w_-).u_-v_-j},
{u_-v_-})` [ENTER]

$$\left[\frac{e^{w_-}}{2} + 1/2 \cdot i \quad \frac{e^{w_-} - j}{2} \right]$$

`cZeros({e^(z_-)-w_-w_-z_-^2},
{w_-z_-})` [ENTER]

$$[.494... \quad -.703...]$$

`cZeros({e^(z_-)-w_-w_-z_-^2},
{w_-z_-=1+j})` [ENTER]

$$[.149...+4.89... \cdot i \quad 1.588...+1.540... \cdot j]$$

d() Tecla $\boxed{2\text{nd}} \boxed{[d]}$ o menú MATH/Calculus $d(\text{expresión1}, \text{var} [, \text{orden}]) \Rightarrow \text{expresión}$ $d(\text{lista1}, \text{var} [, \text{orden}]) \Rightarrow \text{lista}$ $d(\text{matriz1}, \text{var} [, \text{orden}]) \Rightarrow \text{matriz}$

Devuelve la primera derivada de la *expresión1* respecto a *var*. La *expresión1* puede ser una lista o matriz.

El *orden*, si se incluye, debe ser un entero. Si el orden es menor que cero, el resultado será una primitiva.

d() no sigue el mecanismo normal de simplificar por completo sus argumentos y aplicar la función definida a dichos argumentos. Por el contrario, **d()** sigue los pasos indicados a continuación:

1. Simplifica el segundo argumento siempre que no produzca un resultado que no sea una variable.
2. Simplifica el primer argumento siempre que no llame a ningún valor almacenado de la variable determinada en el paso 1.
3. Calcula la derivada simbólica del resultado del paso 2 respecto a la variable del paso 1.
4. Si la variable del paso 1 tiene un valor almacenado o un valor especificado con un operador (!) "with", sustituye a dicho valor en el resultado del paso 3.

 $d(3x^3 - x + 7, x) \boxed{\text{ENTER}}$ $9x^2 - 1$ $d(3x^3 - x + 7, x, 2) \boxed{\text{ENTER}}$ $18 \cdot x$ $d(f(x) \cdot g(x), x) \boxed{\text{ENTER}}$

$$\frac{d}{dx}(f(x)) \cdot g(x) + \frac{d}{dx}(g(x)) \cdot f(x)$$

 $d(\sin(f(x)), x) \boxed{\text{ENTER}}$

$$\cos(f(x)) \cdot \frac{d}{dx}(f(x))$$

 $d(x^3, x) | x=5 \boxed{\text{ENTER}}$ 75 $d(x^2 + y^3, y) \boxed{\text{ENTER}}$ $6 \cdot y^2 \cdot x$ $d(x^2, x, -1) \boxed{\text{ENTER}}$ $\frac{x^3}{3}$ $d\{x^2, x^3, x^4\}, x) \boxed{\text{ENTER}}$ $\{2 \cdot x \quad 3 \cdot x^2 \quad 4 \cdot x^3\}$ **data▶mat** CATALOG/MATH/List menu**data▶mat** *datos, mat*, *fila1* [, *col1*] [, *fila2*] [, *col2*]**data▶mat** *d1, m1, l1, . . . l1* $\boxed{\text{ENTER}}$

Convierte los datos en una matriz.

Done

Cada argumento *[, fila1]*, *[, col1]*, *[, fila2]*, *[, col2]* se puede omitir de forma individual. Si se omite *fila1*, el valor predeterminado es 1. Si se omite *col1*, el valor predeterminado es 1. Si el argumento omitido es *fila2*, el valor predeterminado es "fila máxima"; si el argumento omitido es *col2*, el valor predeterminado es "columna máxima".

La estructura DATA permite el uso de celdas vacías. Las filas no tienen que ser de igual tamaño. Cuando se guardan los datos en una matriz, las celdas vacías se propagan con el nombre "sin definir".

dayOfWk() CATALOG**dayOfWk**(*año, mes, día*) \Rightarrow entero**dayOfWk**(1948, 9, 6)

2

Devuelve un entero del 1 al 7 que representa el correspondiente día de la semana. Use **dayOfWk()** para averiguar en qué día de la semana cae una fecha determinada.

Nota: El resultado puede no ser exacto para años anteriores a 1583 (calendario gregoriano).

Introduzca el año como un número entero de cuatro cifras. El mes y el día pueden ser enteros de una o dos cifras.

Valores enteros:

1 = Domingo

2 = Lunes

3 = Martes

4 = Miércoles

5 = Jueves

6 = Viernes

7 = Sábado

►DD Menú MATH/Angle

número ►DD ⇒ valor
lista1 ►DD ⇒ lista
matriz1 ►DD ⇒ matriz

Devuelve el equivalente decimal del argumento expresado en grados. El argumento es un número, lista o matriz que se interpreta en función del valor de Modo especificado, es decir, en grados centesimales, radianes o grados.

En el modo Angle, en grados:
1.5° ►DD [ENTER] 1.5°
45° 22' 14.3" ►DD [ENTER] 45.370...°
{45° 22' 14.3" ,60° 0' 0"} ►DD [ENTER] {45.370... 60}°
En el modo Angle, en grados centesimales:
1►DD [ENTER] (9/10)°
En el modo Angle, en radianes:
1.5 ►DD [ENTER] 85.9°

►Dec Menú MATH/Base

entero1 ►Dec ⇒ entero

Convierte el *entero1* en un número decimal (base 10). Una entrada binaria o hexadecimal debe tener siempre el prefijo 0b ó 0h, respectivamente.

└─ Cero, no la letra O, seguido por b o h.

0b Número binario

0h Número hexadecimal

└─ Los números binarios pueden tener hasta 32 dígitos; los hexadecimales, un máximo de

Sin prefijo, el *entero1* se considera como decimal. El resultado se muestra en decimal, independientemente del estado del modo Base.

0b10011 ►Dec [ENTER] 19
0h1F ►Dec [ENTER] 31

Define CATALOG

Define Nombre de función (Nombre de arg1, Nombre de arg2, ...) = expresión

Crea *Nombre de función* como una función definida por el usuario. Puede utilizar *Nombre de función()* igual que las funciones implementadas. La función calcula la *expresión* utilizando los argumentos dados y devuelve el resultado.

Nombre de función no puede ser el nombre de una variable del sistema o de una función implementada.

Los nombres de argumentos son posiciones, por lo que no debe utilizar estos mismos nombres al calcular la función.

Nota: Esta forma de **Define** equivale a ejecutar la expresión: *expresión*► *Nombre de función* (*Nombre de arg1*, *Nombre de arg2*).

Esta orden también sirve para definir variables simples, por ejemplo, Define a=3.

Define g(x,y)=2x-3y [ENTER] Done
g(1,2) [ENTER] -4
1►a:2►b:g(a,b) [ENTER] -4
Define h(x)=when(x<2,2x-3, -2x+3) [ENTER] Done
h(-3) [ENTER] -9
h(4) [ENTER] -5
Define eigenv1(a)=
cZeros(det(identity(dim(a)
[1])-x*a).x) [ENTER] Done
eigenv1([-1.2;4.3]) [ENTER]
$$\left\{ \frac{2 \cdot \sqrt{3} - 1}{11} \quad \frac{-(2 \cdot \sqrt{3} + 1)}{11} \right\}$$

Define *Nombre de función*(*Nombre de arg1*, *Nombre de arg2*, ...) = **Func** *bloque*
EndFunc

Es idéntica a la forma anterior de **Define**, excepto que aquí la función definida por el usuario *Nombre de función*(**)** puede ejecutar un bloque de varios enunciados.

El *bloque* puede ser un único enunciado o una serie de varios enunciados separados con el carácter ";". El *bloque* también puede incluir expresiones e instrucciones (tal como **If**, **Then**, **Else** y **For**). Así, permite que la función *Nombre de función*(**)** utilice la instrucción **Return** para devolver un resultado determinado.

Nota: Es más fácil crear y editar esta forma de Func en Program Editor que en la línea de entrada.

```
Define g(x,y)=Func:If x>y Then
:Return x:Else:Return y:EndIf
:EndFunc [ENTER]
g(3,-7) [ENTER]
```

Done
3

Define *Nombre de programa*(*Nombre de arg1*, *Nombre de arg2*, ...) = **Prgm** *bloque*
EndPrgm

Crea *Nombre de programa* como un programa o subprograma, aunque no puede devolver un resultado con **Return**. Puede ejecutar un bloque de varios enunciados.

El *bloque* puede ser un único enunciado o una serie de varios enunciados separados con el carácter ";". El *bloque* también puede incluir expresiones e instrucciones (como **If**, **Then**, **Else** y **For**) sin limitaciones.

Nota: Es más fácil crear y editar un bloque de un programa en Program Editor que en la línea de entrada.

```
Define listinnt()=prgm:Local
n,i,str1,num:InputStr "Enter name of
list",str1:Input "No. of
elements":n:For i,1,n,1:Input
"element "&string(i).num:
num->#str1[i]:EndFor:EndPrgm [ENTER]
listinnt() [ENTER] Enter name of list
```

Done

DelFold CATALOG

DelFold *Nombre de carpeta1*[, *Nombre de carpeta2* [, *Nombre de carpeta3*] ...

Borra las carpetas definidas por el usuario con los nombres *Nombre de carpeta1*, *Nombre de carpeta2*, etc. Se muestra un mensaje de error si las carpetas contienen variables.

Nota: No se puede borrar la carpeta main.

```
NewFold games [ENTER]
(crea la carpeta games)
```

Done

```
DelFold games [ENTER]
(borra la carpeta games)
```

Done

DelType CATALOG

DelType *tipo_de_variable*

Borra todas las variables no bloqueadas del tipo especificado en *tipo_de_variable*.

Nota: Los valores posibles para *tipo_de_variable* son:

ASM, DATA, EXPR, FUNC, GDB, LIST, MAT, PIC, PRGM, STR, TEXT, AppVar_nombre_tipo, All.

```
Deltype "LIST" [ENTER]
```

Done

DelVar CATALOG

DelVar *var1*[, *var2*] [, *var3*] ...

Borra de la memoria las variables especificadas.

```
2->a [ENTER]
```

2

```
(a+2)^2 [ENTER]
```

16

```
DelVar a [ENTER]
```

Done

```
(a+2)^2 [ENTER]
```

(a + 2)²

deSolve() Menú MATH/Calculus

deSolve(Edo de primer o segundo orden, Var independiente, Var dependiente) ⇒ solución general

Devuelve una ecuación que, explícita o implícitamente, especifica una solución general de la ecuación diferencial ordinaria de primer o segundo orden (EDO). En la EDO:

- Utilice un símbolo de prima ('), pulse $\boxed{2\text{nd}} \boxed{[']}$ para indicar la primera derivada de la variable dependiente con respecto a la variable independiente.
- Utilice dos símbolos de prima para indicar la correspondiente segunda derivada.

El símbolo ' se utiliza para derivadas sólo dentro de **deSolve()**. En otros casos, utilice $d()$.

La solución general de una ecuación de primer orden contiene una constante arbitraria de la forma @k, donde k es un entero comprendido entre 1 y 255. Dicho entero toma el valor 1 cuando se utiliza **ClrHome** o $\boxed{\text{F1}} \boxed{8}$: Clear Home. La solución de una ecuación de segundo orden contiene dos constantes semejantes.

Aplique **solve()** a una solución implícita si desea intentar convertirla en una o más soluciones explícitas equivalentes.

Al comparar los resultados con soluciones de libros de texto o manuales, tenga en cuenta que los diferentes métodos introducen constantes arbitrarias en distintos momentos momentos del cálculo, lo que puede dar lugar a diferentes soluciones generales.

Nota: Para escribir el símbolo "prima" ('), pulse $\boxed{2\text{nd}} \boxed{[']}$.

```
deSolve(y''+2y'+y=x^2,x,y)  $\boxed{\text{ENTER}}$ 
y=(@1*x+@2)*e^-x+x^2-4*x+6
right(ans(1))>temp  $\boxed{\text{ENTER}}$ 
(@1*x+@2)*e^-x+x^2-4*x+6
d(temp,x,2)+2*d(temp,x)+temp-x^2
 $\boxed{\text{ENTER}}$  0
delVar temp  $\boxed{\text{ENTER}}$  Done
```

```
deSolve(y'=(cos(y))^2*x,x,y)  $\boxed{\text{ENTER}}$ 
tan(y)= $\frac{x^2}{2}$ +@3
solve(ans(1),y)  $\boxed{\text{ENTER}}$ 
y=tan( $\left(\frac{x^2+2*\text{@3}}{2}\right)$ )+@n1*\pi
```

Nota: Para escribir un símbolo @, pulse:

```
 $\boxed{\text{2nd}} \boxed{[*]} \boxed{\text{STO}}$ 
 $\boxed{\text{2nd}} \boxed{\text{R}}$ 
ans(1)|@3=c-1 and @n1=0  $\boxed{\text{ENTER}}$ 
y=tan( $\left(\frac{x^2+2*(c-1)}{2}\right)$ )
```

deSolve(1Edo de primer orden and Condición inicial, Var independiente, Var dependiente) ⇒ solución particular

Devuelve una solución particular que satisface Edo de primer orden y Condición inicial. Por lo general, esto es más sencillo que determinar una solución general, sustituir valores iniciales, dar una solución para la constante arbitraria y, a continuación, sustituir este valor en la solución general.

Condición inicial es una ecuación de la forma:

Var dependiente (Valor independiente inicial) = Valor dependiente inicial

Valor independiente inicial y Valor dependiente inicial pueden ser variables tales como x0 y y0 que no tengan valores almacenados. La diferenciación implícita puede ayudar a verificar las soluciones implícitas.

```
sin(y)=(y*e^x+cos(y))y'>ode  $\boxed{\text{ENTER}}$ 
sin(y)=(e^x*y+cos(y))*y'
deSolve(ode and y(0)=0,x,y)>soln
 $\boxed{\text{ENTER}}$ 
 $\frac{-(2*\sin(y)+y^2)}{2} = -(e-1)*e^x*\sin(y)$ 
soln|x=0 and y=0  $\boxed{\text{ENTER}}$  true
d(right(eq)-left(eq),x)/
(d(left(eq)-right(eq),y))
>impdif(eq,x,y)  $\boxed{\text{ENTER}}$  Done
ode|y'=impdif(soln,x,y)  $\boxed{\text{ENTER}}$  true
delVar ode.soln  $\boxed{\text{ENTER}}$  Done
```

deSolve(*Edo de segundo orden and Condición inicial1 and Condición inicial2, Var independiente, Var dependiente*) ⇒ *solución particular*

Devuelve una solución particular que satisface *Edo de segundo orden* y tiene el valor concreto de la variable dependiente y su primera derivada en un punto.

Para *Condición inicial1*, utilice la forma:

Var dependiente (Valor independiente inicial) = Valor dependiente inicial

Para *Condición inicial2*, utilice la forma:

Var dependiente¹ (Valor independiente inicial) = Valor inicial primera derivada

deSolve(y''=y^{-1/2} and y(0)=0 and y'(0)=0,t.y) **[ENTER]**

$$\frac{2 \cdot y^{3/4}}{3} = t$$

solve(ans(1),y) **[ENTER]**

$$y = \frac{2^{2/3} \cdot (3 \cdot t)^{4/3}}{4} \text{ and } t \geq 0$$

deSolve(*Edo de segundo orden and límiteCondición1 and límiteCondición2, Var independiente, Var dependiente*) ⇒ *solución particular*

Devuelve una solución particular que satisface *Edo de segundo orden* y tiene valores concretos en dos puntos diferentes.

deSolve(w'' - 2w'/x + (9+2/x^2)w = x * e^x(x) and w(π/6)=0 and w(π/3)=0,x,w) **[ENTER]**

$$w = \frac{e^x \cdot x \cdot \cos(3 \cdot x)}{10}$$

$$- \frac{e^x \cdot x \cdot \sin(3 \cdot x)}{10} + \frac{x \cdot e^x}{10}$$

det() Menú MATH/Matrix

det(*Matriz cuadrada, tol*) ⇒ *expresión*

Devuelve el determinante de *Matriz cuadrada*.

De forma opcional, cualquier elemento de matriz se trata como cero si su valor absoluto es menor que *tol*. Esta tolerancia se utiliza sólo si la matriz tiene entradas de coma flotante y no contiene ninguna variable simbólica sin valor asignado. De no ser así, *tol* se ignora.

- Si se utiliza **[\square]** **[ENTER]** o se establece el modo en Exact/Approx=APPROXIMATE, los cálculos se realizan mediante aritmética de coma flotante.
- Si *tol* se omite o no se utiliza, la tolerancia por omisión se calcula como:

$$5 \cdot 10^{-14} \cdot \max(\mathbf{dim}(\mathbf{Matriz\ cuadrada}), \mathbf{rowNorm}(\mathbf{Matriz\ cuadrada}))$$

det([a,b;c,d]) **[ENTER]** a · d - b · c

det([1,2;3,4]) **[ENTER]** -2

det(identity(3) - x * [1, -2, 3; -2, 4, 1; -6, -2, 7]) **[ENTER]**
 $-(98 \cdot x^3 - 55 \cdot x^2 + 12 \cdot x - 1)$

[1;20,1;0,1]·mat1 $\begin{bmatrix} 1. \cdot 20 & 1 \\ 0 & 1 \end{bmatrix}$

det(mat1) **[ENTER]** 0

det(mat1..1) **[ENTER]** 1. · 20

diag() Menú MATH/Matrix

diag(*lista*) ⇒ *matriz*

diag(*Matriz de fila*) ⇒ *matriz*

diag(*Matriz de columna*) ⇒ *matriz*

Devuelve una matriz con los valores de la lista de argumentos situados en la diagonal principal.

diag({2,4,6}) **[ENTER]** $\begin{bmatrix} 2 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 6 \end{bmatrix}$

diag(*Matriz cuadrada*) ⇒ *Matriz de fila*

Devuelve una matriz-fila que contiene los elementos de la diagonal principal de *Matriz cuadrada*.

Matriz cuadrada debe ser cuadrada.

[4,6,8;1,2,3;5,7,9] **[ENTER]** $\begin{bmatrix} 4 & 6 & 8 \\ 1 & 2 & 3 \\ 5 & 7 & 9 \end{bmatrix}$

diag(ans(1)) **[ENTER]** [4 2 9]

Dialog CATALOG

Dialog

bloque

EndDialog

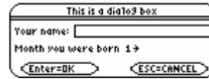
Genera un recuadro de diálogo cuando se ejecuta el programa.

El *bloque* puede ser un único enunciado o una serie de varios enunciados separados por el carácter ":". Las opciones válidas de *bloque* en el elemento del menú [F3] I/O, 1:Dialog de Program Editor, son 1:Text, 2:Request, 4:DropDown y 7:Title.

Las variables en un recuadro de diálogo pueden tener valores que se mostrarán como los valores por omisión (o iniciales). Si se pulsa [ENTER], las variables se actualizan en el recuadro de diálogo y la variable ok se ajusta en 1. Si se pulsa [ESC], las variables no se actualizan, y la variable del sistema ok se establece en cero.

Listado del programa:

```
:Dlogtest()
:Prgm
:Dialog
:Title           "This is a dialog box"
:Request         "Your name".Str1
:DropDown       "Month you were born".
                seq(string(i),i,1,12).Var1
:EndDialog
:EndPrgm
```



dim() Menú MATH/Matrix/Dimensions

dim(lista) ⇒ *entero*

Devuelve la dimensión de la *lista*.

dim({0,1,2}) [ENTER]

3

dim(matriz) ⇒ *lista*

Devuelve las dimensiones de *matriz* como una lista de dos elementos (filas, columnas).

dim([1, -1,2; -2,3,5]) [ENTER]

{2 3}

dim(cadena) ⇒ *entero*

Devuelve el número de caracteres contenidos en la cadena de caracteres *cadena*.

dim("Hello") [ENTER]

5

dim("Hello"&" there") [ENTER]

11

dir. indirect. Consulte #(), página 209.

Disp CATALOG

Disp [*exprOCadena1*] [, *exprOCadena2*] ...

Muestra el contenido actual de la pantalla Program I/O. Si se especifica una o más *exprOCadena*, muestra cada expresión o cadena de caracteres en una línea distinta de la pantalla Program I/O.

Una expresión puede incluir operaciones de conversión tales como ►DD y ►Rect. También puede utilizarse el operador ► para realizar conversiones de unidades y bases de numeración.

Si Pretty Print = ON, las expresiones se muestran en "pretty print".

En la pantalla Program I/O, se puede pulsar [F5] para mostrar la pantalla Home; también un programa puede utilizar **DispHome**.

Disp "Hello" [ENTER] Hello

Disp cos(2.3) [ENTER] -.666...

{1,2,3,4}►L1 [ENTER]

Disp L1 [ENTER] {1 2 3 4}

Disp 180_min►_hr [ENTER] 3.◦_hr

Nota: Para escribir un guión de subrayado (_), pulse:

◻ [_]

◻ [2nd] [_]

Para escribir ►, pulse [2nd] [►].

DispG CATALOG

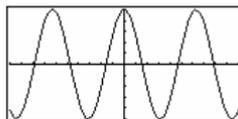
DispG

Muestra el contenido actual de la pantalla Graph.

En el modo de gráficas de función:

Parte de un programa:

```
:  
:5*cos(x)→y1(x)  
:-10→xmin  
:10→xmax  
:-5→ymin  
:5→ymax  
:DispG  
:
```



DispHome CATALOG

DispHome

Muestra el contenido actual de la pantalla Home.

Parte de un programa:

```
:  
:Disp "The result is: ".xx  
:Pause "Press Enter to quit"  
:DispHome  
:EndPrgm
```

DispTbl CATALOG

DispTbl

Presenta el contenido actual de la pantalla Table.

Nota: La tecla del cursor está activada para que pueda desplazarse. Pulse **ESC** o **ENTER** para reanudar la ejecución de un programa, en caso necesario.

5*cos(x)→y1(x) **ENTER**

DispTbl **ENTER**

F1	F2	F3	F4	F5	F6
Tool	Set	Stat	Draw	Func	Prog
x	y1				
-2.	-2.081				
-1.	2.7015				
0.	5.				
1.	2.7015				
2.	-2.081				

x=-2.
MIN RAD AUTO FUNC

DMS Menú MATH/Angle

expresión **►DMS**

lista **►DMS**

matriz **►DMS**

Interpreta el argumento como un ángulo y presenta el número equivalente de DMS (GGGGG°MMSS.s"). Consulte °, ' , " para más información sobre el formato DMS (grados, minutos, segundos).

Nota: **►DMS** convierte de radianes a grados cuando se utiliza en el modo de radianes. Si la entrada está seguida del símbolo de grados (°), no se produce la conversión. Sólo se puede emplear **►DMS** al final de la línea de entrada.

En el modo Angle, en grados:

45.371 **►DMS** **ENTER** 45° 22' 15.6"

{45.371,60} **►DMS** **ENTER** {45° 22' 15.6" 60° }

dotP() Menú MATH/Matrix/Vector ops

dotP(*lista1*, *lista2*) ⇒ *expresión*

Devuelve el producto "escalar" de dos listas.

`dotP({a,b,c},{d,e,f})` **ENTER**
 $a \cdot d + b \cdot e + c \cdot f$

`dotP({1,2},{5,6})` **ENTER** 17

dotP(*vector1*, *vector2*) ⇒ *expresión*

Devuelve el producto "escalar" de dos vectores.

Ambos deben ser vectores fila o columna, respectivamente.

`dotP([a,b,c],[d,e,f])` **ENTER**
 $a \cdot d + b \cdot e + c \cdot f$

`dotP([1,2,3],[4,5,6])` **ENTER** 32

DrawFunc CATALOG

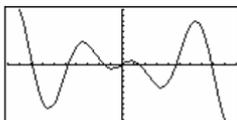
DrawFunc *expresión*

Realiza la gráfica de *expresión*, considerándola como una función, con *x* como variable independiente.

Nota: Al volver a dibujar una gráfica, se borran todos los elementos complementarios dibujados.

En el modo de gráficas de función y con la ventana **ZoomStd**:

`DrawFunc 1.25x*cos(x)` **ENTER**



DrawInv CATALOG

DrawInv *expresión*

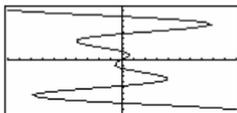
Dibuja la inversa de la *expresión* y representa los valores de *x* en el eje *y*, y los valores de *y* en el eje *x*.

x es la variable independiente.

Nota: Al volver a dibujar una gráfica, se borran todos los elementos dibujados.

En el modo de gráficas de función y con la ventana **ZoomStd**:

`DrawInv 1.25x*cos(x)` **ENTER**



DrawParm CATALOG

DrawParm *expresión1*, *expresión2*
[, *tmin*] [, *tmax*] [, *tpaso*]

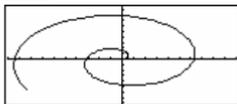
Dibuja la gráfica en paramétricas de la *expresión1* y la *expresión2*, con *t* como variable independiente.

Los valores por omisión de *tmin*, *tmax* y *tpaso* son los actuales de las variables de ventana *tmin*, *tmax* y *tstep*. Especificar valores no altera los estados de la ventana. Si el modo de representación gráfica actual no es en paramétricas, se requieren los tres argumentos indicados arriba.

Nota: Al volver a dibujar una gráfica, se borran todos los elementos complementarios dibujados.

En el modo de gráficas de función y con la ventana **ZoomStd**:

`DrawParm t*cos(t).t*sin(t).0.10..1`
ENTER



DrawPol CATALOG

DrawPol *expresión*, θ_{min} [, θ_{max}] [, θ_{paso}]

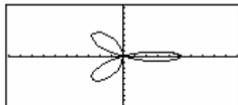
Dibuja la gráfica en polares de *expresión*, con θ como la variable independiente.

Los valores por omisión de θ_{min} , θ_{max} y θ_{paso} son los actuales de las variables de ventana θ_{min} , θ_{max} y θ_{step} . Especificar valores no altera los estados de la ventana. Si el modo de representación gráfica actual no es en polares, se requieren estos argumentos.

Nota: Al volver a dibujar una gráfica, se borran todos los elementos complementarios dibujados.

En el modo de gráficas de función y en una ventana ZoomStd:

DrawPol 5*cos(3* θ),0.3.5..1 **ENTER**



DrawSlp CATALOG

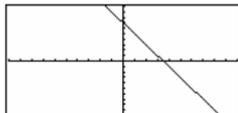
DrawSlp $x1$, $y1$, *pendiente*

Dibuja la recta de ecuación $y - y1 = \text{pendiente} \cdot (x - x1)$.

Nota: Al volver a dibujar una gráfica, se borran todos los elementos complementarios dibujados.

En el modo de gráficas de función y con la ventana ZoomStd:

DrawSlp 2.3, -2 **ENTER**



DropDown CATALOG

DropDown *títuloCadena*, {*elemento1Cadena*, *elemento2Cadena*, ...}, *Nombre de var*

Muestra un menú que se abre con el nombre *títuloCadena* y que contiene los elementos

1: *elemento1Cadena*, **2:** *elemento2Cadena*, etc.

DropDown debe estar dentro de un bloque **Dialog...EndDialog**.

Si *Nombre de var* existe y tiene un valor dentro del rango de elementos, se muestra el elemento referido como la selección por omisión. De lo contrario, el primer elemento del menú es la selección por omisión.

Cuando selecciona un elemento de un menú, el número correspondiente del elemento se almacena en la variable *Nombre de var* (si fuera necesario, **DropDown** también crea *Nombre de var*).

Consulte el ejemplo de listado del programa **Dialog**.

DrwCtour CATALOG

DrwCtour *expresión*

DrwCtour *lista*

Dibuja los contornos de la gráfica 3D actual en los valores z especificados por *expresión* o *lista*. El modo de gráficas 3D debe haberse establecido previamente. **DrwCtour** ajusta de forma automática el estilo del formato de la representación a CONTOUR LEVELS.

Por omisión, la representación contiene automáticamente el número de contornos equiespaciados especificados por la variable de ventana ncontour. **DrwCtour** dibuja contornos además de los valores por omisión.

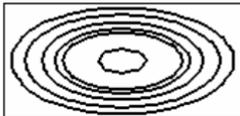
Para desactivar los contornos por omisión, ajuste ncontour a cero, mediante la pantalla Window o almacene 0 en la variable de sistema ncontour.

En el modo de gráficas 3D:

$(1/5)x^2+(1/5)y^2-10 \cdot z1(x,y)$ [ENTER]

-10 • xmin:10 • xmax [ENTER] Done
 -10 • ymin:10 • ymax [ENTER] 10
 -10 • zmin:10 • zmax [ENTER] 10
 0 • ncontour [ENTER] 0

DrwCtour {-9,-4.5,-3.0,4.5,9} [ENTER]



- Utilice el cursor para cambiar el ángulo de visualización. Pulse 0 (cero) para volver a la visualización original.

Para cambiar entre distintos estilos de formato gráfico, pulse:



- Pulse X, Y o Z para tener una vista descendente del eje correspondiente.

E	Tecla [EE]	Tecla [2nd][EE]	
	<i>mantisa</i> \times <i>exponente</i>		
	Introduce un número en notación científica. El número se interpreta como <i>mantisa</i> \times 10 <i>exponente</i> .		
	Sugerencia: Si quiere introducir una potencia de 10 sin obtener un resultado en valores decimales, utilice 10 ^{entero} .		
		2.3E 4 [ENTER]	23000.
		2.3E 9+4.1E 15 [ENTER]	4.1E 15
		3* 10^4 [ENTER]	30000

e^()	Tecla [e^x]	Tecla [2nd][e^x]	
	$e^{(expresión)}$ \Rightarrow <i>expresión</i>		
	Devuelve <i>e</i> elevado a la potencia dada por <i>expresión</i> .		
	Nota: En la TI-89 Titanium, pulsar [e^x] para presentar e^x (es distinto que pulsar [alpha][E]). En la Voyage 200, pulsar [2nd][e^x] para mostrar e^x es distinto que acceder al carácter e desde el teclado QWERTY.		
	Aunque puede introducir un número complejo en forma polar $r e^{j\theta}$, utilice este formato sólo para modo de Angle en radianes; en cualquier otro modo de ángulo, grados o grados centesimales, genera un error de dominio, Domain error.		
		$e^{(1)}$ [ENTER]	<i>e</i>
		$e^{(1.)}$ [ENTER]	2.718...
		$e^{(3)^2}$ [ENTER]	e^9

e^(lista)	\Rightarrow <i>lista</i>	
	Devuelve <i>e</i> elevado a la potencia de cada elemento de la <i>lista</i> .	
		$e^{(\{1.1..0.5\})}$ [ENTER] $\{e \ 2.718... \ 1 \ 1.648...\}$

e^{\wedge} (Matriz cuadrada) \Rightarrow Matriz cuadrada

Devuelve la matriz exponencial de *Matriz cuadrada1*. Esto *no* es lo mismo que calcular *e* elevado a cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.

Matriz cuadrada1 debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

e^{\wedge} ([1.5,3;4.2,1;6,-2,1]) **ENTER**

782.209	559.617	456.509
680.546	488.795	396.521
524.929	371.222	307.879

eigVc() Menú MATH/Matrix

eigVc(Matriz cuadrada) \Rightarrow matriz

Devuelve una matriz que contiene los vectores propios para una *Matriz cuadrada* real o compleja, donde cada columna en el resultado corresponde a un valor propio. Tenga en cuenta que un vector propio no es único; puede venir afectado por cualquier factor constante. Los vectores propios están normalizados, lo que significa que si $V = [x_1, x_2, \dots, x_n]$, entonces:

$$\sqrt{x_1^2 + x_2^2 + \dots + x_n^2} = 1$$

A *Matriz cuadrada* se le aplican transformaciones similares hasta que las normas de las filas y columnas se aproximan al mismo valor todo lo posible. A continuación, *Matriz cuadrada* se reduce a la forma Hessenberg superior y los vectores propios se obtienen desde esta última matriz.

En el modo de formato complejo rectangular:

[-1.2,5;3,-6.9;2,-5.7] \rightarrow m1 **ENTER**

-1	2	5
3	-6	9
2	-5	7

eigVc(m1) **ENTER**

-.800...	.767...	.767...
.484...	.573...+ .052... <i>i</i>	.573...- .052... <i>i</i>
.352...	.262...+ .096... <i>i</i>	.262...- .096... <i>i</i>

eigVl() Menú MATH/Matrix

eigVl(Matriz cuadrada) \Rightarrow lista

Devuelve una lista de los valores propios de una *Matriz cuadrada* real o compleja.

A *Matriz cuadrada* se le aplican transformaciones similares hasta que las normas de las filas y columnas se aproximan al mismo valor todo lo posible. A continuación, *Matriz cuadrada* se reduce a la forma Hessenberg superior y los vectores propios se obtienen desde esta última matriz.

En el modo de formato complejo rectangular:

[-1.2,5;3,-6.9;2,-5.7] \rightarrow m1 **ENTER**

-1	2	5
3	-6	9
2	-5	7

eigVl(m1) **ENTER**

{- 4.409... 2.204...+ .763...*i* /2.204...- .763...*i*}

Else Consulte **If**, página 209.

Elseif **CATALOG** Consulte además **If**, página 209.

```
If expresión booleana1 Then
  bloque1
Elseif expresión booleana2 Then
  bloque2
:
Elseif expresión booleanaN Then
  bloqueN
EndIf
:
```

Elseif puede utilizarse como una instrucción de programa para provocar una bifurcación.

```
Parte de un programa:
:
:If choice=1 Then
: Goto option1
: Elseif choice=2 Then
: Goto option2
: Elseif choice=3 Then
: Goto option3
: Elseif choice=4 Then
: Disp "Exiting Program"
Return
:EndIf
:
```

EndCustm Consulte **Custom**, página 199.

EndDlog Consulte **Dialog**, página 207.

EndFor Consulte **For**, página 209.

EndFunc Consulte **Func**, página 209.

EndIf Consulte **If**, página 209.

EndLoop Consulte **Loop**, página 209.

EndPrgm Consulte **Prgm**, página 209.

EndTBar Consulte **ToolBar**, página 209.

EndTry Consulte **Try**, página 209.

EndWhile Consulte **While**, página 209.

entry() CATALOG

entry() ⇒ *expresión*

entry(entero) ⇒ *expresión*

Trae a la línea de entrada una expresión previamente introducida y que se halle en el área de historia de la pantalla Home.

El *entero*, si se incluye, especifica la expresión concreta del área de historia. El valor por omisión es 1, la entrada más reciente. El rango válido está comprendido entre 1 y 99, y no puede ser una expresión.

Nota: Si la última entrada sigue resaltada en la pantalla Home, pulsar **ENTER** será lo mismo que ejecutar **entry(1)**.

En la pantalla Home:

1+1/x **ENTER** $\frac{1}{x} + 1$

1+1/entry(1) **ENTER** $2 - \frac{1}{x+1}$

ENTER $\frac{1}{2 \cdot (2 \cdot x + 1)} + 3/2$

ENTER $5/3 - \frac{1}{3 \cdot (3 \cdot x + 2)}$

entry(4) **ENTER** $\frac{1}{x} + 1$

exact() Menú MATH/Number

exact(expresión1 [, tol]) ⇒ *expresión*

exact(lista1 [, tol]) ⇒ *lista*

exact(matriz1 [, tol]) ⇒ *matriz*

Utiliza la aritmética del modo Exact independientemente del estado del modo Exact/Approx para devolver, en los casos en que sea posible, el argumento en forma racional.

tol especifica la tolerancia de la conversión, y su valor por omisión es 0 (cero).

exact(.25) **ENTER** $1/4$

exact(.333333) **ENTER** $\frac{333333}{1000000}$

exact(.333333 .001) $1/3$

exact(3.5x+y) **ENTER** $\frac{7 \cdot x}{2} + y$

exact({.2 .33.4.125}) **ENTER** $\{1/5 \frac{33}{100} 33/8\}$

Exec CATALOG

Exec *cadena* [, *expresión1*] [, *expresión2*] ...

Ejecuta una *cadena* consistente en una serie de códigos op de Motorola 68000. Estos códigos actúan de forma similar a un lenguaje ensamblador. En caso necesario, las *expresiones* opcionales permiten pasar uno o más argumentos al programa.

Para más información, consulte el sitio web de TI: education.ti.com

Advertencia: **Exec** proporciona acceso a todas las funciones del microprocesador. Tenga presente que puede cometer fácilmente un error que bloquee la calculadora y le haga perder datos. Conviene realizar una copia de seguridad del contenido de la calculadora antes de utilizar la orden **Exec**.

Exit CATALOG

Exit

Provoca la salida de un bloque **For**, **While** o **Loop**.

Exit únicamente está permitida en las tres estructuras de bucle (**For**, **While** o **Loop**).

Listado del programa:

```
:0→temp
:For i,1,100,1
: temp+i→temp
: If temp>20
: Exit
:EndFor
:Disp temp
```

Contenido de **temp** después de la ejecución:
21

explist() CATALOG

explist(*expresión*, *var*) ⇒ *lista*

Devuelve una lista con todas las soluciones de una ecuación. Esto ofrece una manera sencilla de extraer algunas soluciones incorporadas a los resultados de las funciones **solve()**, **cSolve()**, **fMin()** y **fMax()**.

Nota: **explist()** no es obligatoria con las funciones **zeros** y **cZeros()**, ya que éstas devuelven directamente una lista de soluciones.

```
solve(x^2-x-2=0,x) [ENTER] x=2 or x=-1
explist(solve(x^2-x-2=0,x),x)
[ENTER] { -1 2 }
```

expand() Menú MATH/Algebra

expand(*expresión1* [, *var*]) ⇒ *expresión*

expand(*lista1* [, *var*]) ⇒ *lista*

expand(*matriz1* [, *var*]) ⇒ *matriz*

expand(*expresión1*) devuelve la *expresión1* desarrollada respecto a todas sus variables. El desarrollo es polinómico en el caso de polinomios y de un desarrollo parcial fraccionario para expresiones racionales.

El objetivo de **expand()** es transformar la *expresión1* en una suma y/o diferencia de términos sencillos. Por el contrario, el objetivo de **factor()** es transformar la *expresión1* en un producto y/o cociente de factores simples.

```
expand((x+y+1)^2) [ENTER]
x^2+2*x*y+2*x+y^2+2*y+1
expand((x^2-x+y^2-y)/(x^2*y^2-x^2
*y-x*y^2+x*y)) [ENTER]
```

$$\text{expand}\left(\frac{x^2-x+y^2-y}{x^2 \cdot y^2 - x^2 \cdot y - x \cdot y^2 + x \cdot y}\right)$$

expand(*expresión1*, *var*) devuelve la *expresión* desarrollada respecto a *var*. Se agrupan potencias similares de *var*. Los términos y sus factores se clasifican utilizando *var* como la variable principal. Puede haber una factorización o desarrollo incidental de los coeficientes agrupados. Comparado con la omisión de *var*, esto suele ahorrar tiempo, memoria y espacio en la pantalla, además de hacer más comprensible la expresión.

Incluso cuando sólo hay una variable, si utiliza *var* puede hacer que la factorización del denominador en el desarrollo parcial fraccionario sea más completa.

Sugerencia: En expresiones racionales, **propFrac()** es una alternativa más rápida aunque menos completa que **expand()**.

Nota: Consulte además **comDenom()** para desarrollar un numerador sobre un denominador también desarrollado.

expand(*expresión1*, [*var*]) también desarrolla logaritmos y potencias fraccionarias sin tomar en cuenta *var*. Para un mejor desarrollo de los logaritmos y potencias fraccionarias, puede ser necesario restringir algunos valores para hacer que no sean negativos.

expand(*expresión1*, [*var*]) también distribuye valores absolutos, **sign()** y exponentes, sin tomar en cuenta *var*.

Nota: Consulte además **tExpand()** para ver la suma trigonométrica de ángulos y el desarrollo de varios ángulos a la vez.

$$\text{expand}((x+y+1)^2, y) \quad \text{ENTER} \quad y^2 + 2 \cdot y \cdot (x+1) + (x+1)^2$$

$$\text{expand}((x+y+1)^2, x) \quad \text{ENTER} \quad x^2 + 2 \cdot x \cdot (y+1) + (y+1)^2$$

$$\text{expand}((x^2 - x + y^2 - y) / (x^2 \cdot y^2 - x^2 \cdot y - x \cdot y^2 + x \cdot y), y) \quad \text{ENTER}$$

$$\text{expand}\left(\frac{x^2 - x + y^2 - y}{x^2 \cdot y^2 - x^2 \cdot y - x \cdot y^2 + x \cdot y}, y\right)$$

$$\text{expand}(\text{ans}(1), x) \quad \text{ENTER}$$

$$\text{expand}\left(\frac{1}{y-1} - \frac{1}{y} + \frac{1}{x \cdot (x-1)}\right)$$

$$\text{expand}((x^3 + x^2 - 2) / (x^2 - 2)) \quad \text{ENTER}$$

$$\frac{2 \cdot x}{x^2 - 2} + x + 1$$

$$\text{expand}(\text{ans}(1), x) \quad \text{ENTER}$$

$$\frac{1}{x - \sqrt{2}} + \frac{1}{x + \sqrt{2}} + x + 1$$

$$\ln(2x \cdot y) + \sqrt{(2x \cdot y)} \quad \text{ENTER}$$

$$\ln(2 \cdot x \cdot y) + \sqrt{(2 \cdot x \cdot y)}$$

$$\text{expand}(\text{ans}(1)) \quad \text{ENTER}$$

$$\ln(x \cdot y) + \sqrt{2} \cdot \sqrt{(x \cdot y)} + \ln(2)$$

$$\text{expand}(\text{ans}(1)) | y >= 0 \quad \text{ENTER}$$

$$\ln(x) + \sqrt{2} \cdot \sqrt{x} \cdot \sqrt{y} + \ln(y) + \ln(2)$$

$$\text{sign}(x \cdot y) + \text{abs}(x \cdot y) + e^{(2x+y)} \quad \text{ENTER}$$

$$e^{2 \cdot x+y} + \text{sign}(x \cdot y) + |x \cdot y|$$

$$\text{expand}(\text{ans}(1)) \quad \text{ENTER}$$

$$(\text{sign}(x) \cdot \text{sign}(y) + |x| \cdot |y| + (e^2)^2 \cdot e^y)$$

expr() Menú MATH/String

expr(*cadena*) \Rightarrow *expresión*

Devuelve la cadena de caracteres contenida en *cadena* como una expresión y la ejecuta inmediatamente.

$$\text{expr}("1+2+x^2+x") \quad \text{ENTER} \quad x^2 + x + 3$$

$$\text{expr}("\text{expand}((1+x)^2)") \quad \text{ENTER}$$

$$x^2 + 2 \cdot x + 1$$

$$\text{"Define cube}(x)=x^3" \Rightarrow \text{funcstr} \quad \text{ENTER}$$

$$\text{"Define cube}(x)=x^3"$$

$$\text{expr}(\text{funcstr}) \quad \text{ENTER}$$

$$\text{Done}$$

$$\text{cube}(2) \quad \text{ENTER}$$

$$8$$

ExpReg Menú MATH/Statistics/Regressions

ExpReg *lista1, lista2* [, *[lista3]* [, *lista4, lista5*]

Calcula la regresión exponencial y actualiza todas las variables estadísticas del sistema.

Todas las listas deben tener el mismo tamaño, excepto la *lista5*.

La *lista1* representa *x*lista.

La *lista2* representa *y*lista.

La *lista3* representa la frecuencia.

La *lista4* representa códigos de categoría.

La *lista5* representa la lista de categorías.

Nota: La *lista1* hasta la *lista4* deben ser un nombre de variable o c1–c99 (columnas en la última variable de datos mostrada en Data/Matrix Editor). La *lista5* no tiene que ser un nombre de variable y no puede ser c1–c99.

En el modo de gráficas de función:

{1,2,3,4,5,6,7,8} → L1 **[ENTER]** {1 2 ...}

{1,2,2,2,3,4,5,7} → L2 **[ENTER]** {1 2 ...}

ExpReg L1,L2 **[ENTER]** Done

ShowStat **[ENTER]**

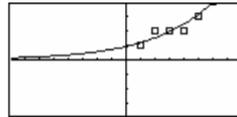


[ENTER]

Regeq(x)→y1(x) **[ENTER]** Done

NewPlot 1,1,L1,L2 **[ENTER]** Done

♦ **[GRAPH]**



factor() Menú MATH/Algebra

factor(*expresión* *f*, *var*) ⇒ *expresión*

factor(*lista* *f*, *var*) ⇒ *lista*

factor(*matriz* *f*, *var*) ⇒ *matriz*

factor(*expresión* *f*) devuelve la *expresión1* factorizada respecto a todas sus variables, sobre un denominador común.

La *expresión1* se descompone todo lo posible en factores racionales lineales sin introducir nuevas subexpresiones no reales. Esta alternativa es apropiada si desea factorizar respecto a más de una variable.

factor(*expresión1*, *var*) devuelve *expresión1* factorizada respecto a la variable *var*.

La *expresión1* se descompone todo lo posible en factores reales que son lineales en *var*, aunque esto introduzca constantes irracionales o subexpresiones que son irracionales en otras variables.

Los factores y sus términos se clasifican con *var* como la variable principal. Las potencias similares de *var* se agrupan en cada factor. Incluya *var* si necesita la factorización sólo respecto a dicha variable, y puede aceptar expresiones irracionales en cualquier otra variable con el fin de incrementar la factorización respecto a *var*. Puede haber una factorización incidental respecto a otras variables.

factor($a^3 * x^2 - a * x^2 - a^3 + a$) **[ENTER]**

factor($x^2 + 1$) **[ENTER]** $x^2 + 1$

factor($x^2 - 4$) **[ENTER]** $(x - 2) \cdot (x + 2)$

factor($x^2 - 3$) **[ENTER]** $x^2 - 3$

factor($x^2 - a$) **[ENTER]** $x^2 - a$

factor($a^3 * x^2 - a * x^2 - a^3 + a, x$) **[ENTER]**

$a \cdot (a^2 - 1) \cdot (x - 1) \cdot (x + 1)$

factor($x^2 - 3, x$) **[ENTER]** $(x + \sqrt{3}) \cdot (x - \sqrt{3})$

factor($x^2 - a, x$) **[ENTER]** $(x + \sqrt{a}) \cdot (x - \sqrt{a})$

En el estado AUTO del modo Exact/Approx, si incluye *var*, permite aproximaciones con coeficientes de coma flotante en los casos en que los coeficientes irracionales no se pueden expresar de forma explícita y concisa respecto a las funciones incorporadas. Incluso cuando hay una sola variable, al incluir *var* puede obtenerse una factorización más completa.

Nota: Consulte además **comDenom()** para ver una manera rápida de obtener una factorización parcial cuando **factor()** no es suficientemente rápida o utiliza toda la memoria.

Nota: Consulte además **cFactor()** para descomponer en coeficientes complejos con el fin de obtener factores lineales.

```
factor(x^5+4x^4+5x^3-6x-3) [ENTER]
x^5 + 4 • x^4 + 5 • x^3 - 6 • x - 3
factor(ans(1),x) [ENTER]
(x - .964...) • (x + .611...) •
(x + 2.125...) • (x^2 + 2.227... +
x + 2.392...)
```

factor(Número racional) devuelve la factorización a números primos del número racional. Para números compuestos, el tiempo de cálculo crece exponencialmente de acuerdo al número de dígitos del segundo factor mayor. Por ejemplo, la factorización de un entero de 30 dígitos puede llevar más de un día, y la factorización de un número de 100 dígitos, más de un siglo.

```
factor(152417172689) [ENTER]
123457 • 1234577
isPrime(152417172689) [ENTER] false
```

Nota: Para detener (interrumpir) un cálculo, pulse **ON**.

Si sólo desea determinar si un número es primo, utilice **isPrime()**. Es mucho más rápido, en particular si *Número racional* no es primo y si el segundo factor mayor tiene más de cinco dígitos.

Fill Menú MATH/Matrix

Fill *expresión, Varmatriz* ⇒ *matriz*

Sustituye cada elemento de la variable *Varmatriz* por la *expresión*.

Varmatriz debe ser una variable ya existente.

```
[1.2;3.4]►amatrix [ENTER]
[ 1 2
 3 4]
Fill 1.01.amatrix [ENTER]
Done
amatrix [ENTER]
[ 1.01 1.01
 1.01 1.01]
```

Fill *expresión, Varlista* ⇒ *lista*

Sustituye cada elemento de la variable *Varlista* por la *expresión*.

Varlista debe existir previamente.

```
{1.2,3.4,5}►alist [ENTER]
{ 1 2 3 4 5}
Fill 1.01.alist [ENTER]
Done
alist [ENTER]
{ 1.01 1.01 1.01 1.01 1.01}
```

floor() Menú MATH/Number

floor(*expresión*) ⇒ *entero*

Devuelve el mayor número entero que es ≤ que el argumento. Esta función es idéntica a **int()**.

El argumento puede ser un número real o complejo.

```
floor(-2.14) [ENTER] -3.
```

floor(*lista*) ⇒ *lista*

floor(*matriz*) ⇒ *matriz*

Devuelve una lista o matriz con los números enteros inmediatamente inferiores a cada elemento.

Nota: Consulte además **ceiling()** e **int()**.

```
floor({3/2,0,-5.3}) [ENTER]
{ 1 0 -6.}
floor([1.2,3.4;2.5,4.8]) [ENTER]
[ 1. 3.
 2. 4.]
```

fMax() Menú MATH/Calculus

fMax(*expresión, var*) ⇒ *expresión booleana*

Devuelve una expresión booleana que determina los posibles valores de *var* que maximizan la *expresión* o hallan la menor de sus cotas superiores.

Utilice el operador "|" para restringir el intervalo de soluciones y/o especificar el signo de otras variables no definidas.

En el estado APPROX del modo Exact/Approx, **fMax()** obtiene iterativamente un máximo aproximado local. Esto suele ser lo más rápido, sobre todo si se utiliza el operador "|" para limitar la búsqueda en un intervalo relativamente pequeño que contenga un solo máximo local.

Nota: Consulte además **fMin()** y **max()**.

fMax(1 - (x - a)^2 - (x - b)^2, x) [ENTER]

$$x = \frac{a+b}{2}$$

fMax(.5x^3 - x - 2, x) [ENTER]

$$x = \infty$$

fMax(.5x^3 - x - 2, x) | x ≤ 1 [ENTER]

$$x = -.816...$$

fMax(a * x^2, x) [ENTER]

$$x = \infty \text{ or } x = -\infty \text{ or } x = 0 \text{ or } a = 0$$

fMax(a * x^2, x) | a < 0 [ENTER]

$$x = 0$$

fMin() Menú MATH/Calculus

fMin(*expresión, var*) ⇒ *expresión booleana*

Devuelve una expresión booleana que especifica posibles valores de *var* que minimizan la *expresión* o localizan la mayor de sus cotas inferiores.

Utilice el operador "|" para restringir el intervalo de soluciones y/o especificar el signo de otras variables no definidas.

En el estado APPROX del modo Exact/Approx, **fMin()** busca iterativamente un mínimo aproximado local. Esto suele ser lo más rápido, especialmente si utiliza el operador "|" para restringir la búsqueda en un intervalo relativamente pequeño que contiene un sólo mínimo local.

Nota: Consulte además **fMax()** y **min()**.

fMin(1 - (x - a)^2 - (x - b)^2, x)

[ENTER]

$$x = \infty \text{ or } x = -\infty$$

fMin(.5x^3 - x - 2, x) | x ≥ 1 [ENTER]

$$x = 1$$

fMin(a * x^2, x) [ENTER]

$$x = \infty \text{ or } x = -\infty \text{ or } x = 0 \text{ or } a = 0$$

fMin(a * x^2, x) | a > 0 and x > 1 [ENTER]

$$x = 1.$$

fMin(a * x^2, x) | a > 0 [ENTER]

$$x = 0$$

FnOff CATALOG

FnOff

Anula la selección de todas las funciones Y= en el modo de representación gráfica actual.

En las pantallas divididas y en el modo Two-Graph, **FnOff** sólo puede aplicarse a la gráfica activa.

FnOff [1], [2] ... [99]

Anula la selección de todas las funciones Y= en el modo de representación gráfica actual.

En el modo de gráficas de función:

FnOff 1,3 [ENTER] anula la selección de y1(x) e y3(x).

En el modo de gráficas en paramétricas:

FnOff 1,3 [ENTER] anula la selección de xt1(t), yt1(t) xt3(t) e yt3(t).

FnOn CATALOG

FnOn

Selecciona todas las funciones Y= que están definidas en modo de representación gráfica actual.

En las pantallas divididas y el modo Two-Graph, **FnOn** sólo se aplica a la gráfica activa.

FnOn [1] [, 2] ... [,99]

Selecciona las funciones Y= especificadas en el modo de representación gráfica actual.

Nota: En el modo 3D, sólo puede seleccionarse una función a la vez. FnOn 2 selecciona z2(x,y) y anula cualquier función seleccionada previamente. En los demás modos de representación gráfica, las funciones seleccionadas previamente no se ven afectadas.

For CATALOG

For *var*, *inferior*, *superior* [, *paso*]
bloque

EndFor

Ejecuta iterativamente los enunciados de *bloque* para cada valor de *var*, de *inferior* a *superior*, con los incrementos de *paso*.

var no puede ser una variable del sistema.

paso puede ser positivo o negativo. El valor por omisión es 1.

bloque puede ser un enunciado único o una serie de varios enunciados separados por el carácter “;”.

Parte de un programa:

```
:  
:  
:0→tempsum : 1→step  
:For i.1.100,step  
: tempsum+i→tempsum  
:EndFor  
:Disp tempsum  
:  
:
```

Contenido de tempsum después de la ejecución: 5050

Contenido de tempsum cuando step se cambia a 2: 2500

format() Menú MATH/String

format(*expresión*, *formatoCadena*) ⇒ *cadena*

Devuelve la *expresión* como una cadena de caracteres de acuerdo con el formato que se indique.

La *expresión* debe simplificarse en un número. El *formatoCadena* es una cadena que debe estar de la siguiente forma: "F[*n*]", "S[*n*]", "E[*n*]", "G[*n*][*c*]", en la que [] indica las partes opcionales.

F[*n*]: Formato fijo. *n* es el número de dígitos que se muestran después del punto decimal.

S[*n*]: Formato científico. *n* es el número de dígitos que se muestran después del punto decimal.

E[*n*]: Formato técnico. *n* es el número de dígitos mostrados después del primer dígito significativo. El exponente se ajusta en un múltiplo de tres, y el punto decimal se mueve a la derecha ninguno, uno o dos dígitos.

G[*n*][*c*]: Igual al formato fijo, aunque separa los dígitos a la izquierda de la base en grupos de tres. *c* especifica el carácter separador del grupo, y es una coma por omisión. Si *c* es un punto, la base se muestra como una coma.

[*Rc*]: Cualquiera de los especificadores anteriores puede tener el sufijo del indicador de base *Rc*, donde *c* es un único carácter que especifica lo que se sustituye en el punto de base.

format(1.234567, "f3") **ENTER** "1.235"
format(1.234567, "s2") **ENTER** "1.23E 0"
format(1.234567, "e3") **ENTER** "1.235E 0"
format(1.234567, "g3") **ENTER** "1.235"
format(1234.567, "g3") **ENTER** "1.234.567"
format(1.234567, "g3,r:") **ENTER** "1:235"

fPart() Menú MATH/Number

fPart(*expresión*) ⇒ *expresión*

fPart(*lista*) ⇒ *lista*

fPart(*matriz*) ⇒ *matriz*

Devuelve la parte decimal del argumento.

En el caso de una lista o matriz, devuelve las partes decimales de los elementos.

El argumento puede ser un número real o complejo.

fPart(-1.234) **ENTER** -.234
fPart({1, -2.3, 7.003}) **ENTER** {0 -.3 .003}

Func CATALOG

Func

bloque

EndFunc

Necesario como primer enunciado para una función definida por varios enunciados.

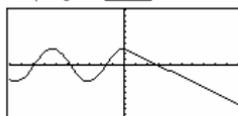
El *bloque* puede ser un único enunciado o una serie de varios enunciados separados por el carácter ":".

Nota: **when()** también puede utilizarse para definir y representar las gráficas de funciones definidas por intervalos.

Define una función por intervalos en el modo de gráficas de función:

Define g(x)=Func:If x<0 Then
:Return 3*cos(x):Else:Return
3-x:EndIf:EndFunc **ENTER** Done

Graph g(x) **ENTER**



gcd() Menú MATH/Number

gcd(*número1*, *número2*) ⇒ *expresión* gcd(18,33) **ENTER** 3

Devuelve el máximo común divisor de dos argumentos. El valor **gcd** de dos fracciones es el valor **gcd** de sus numeradores dividido entre el **lcm** de sus denominadores.

En el modo automático o aproximado, la **gcd** de números fraccionarios de coma flotante es 1.0.

gcd(*lista1*, *lista2*) ⇒ *lista* gcd({12,14,16},{9,7,5}) **ENTER** {3 7 1}

Devuelve el máximo común divisor de los elementos correspondientes de la *lista1* y la *lista2*.

gcd(*matriz1*, *matriz2*) ⇒ *matriz* gcd([2,4;6,8],[4,8;12,16]) **ENTER** $\begin{bmatrix} 2 & 4 \\ 6 & 8 \end{bmatrix}$

Devuelve el máximo común divisor de los elementos correspondientes de la *matriz1* y la *matriz2*.

Get CATALOG

Get *var* Parte de un programa:
:
:Send {3.1,-1.0}
:For i,1,99
: Get data[i]
: PtOn i,data[i]
:EndFor
:
:

Recupera un valor CBL™ (Calculator-Based Laboratory™) o CBR™ (Calculator-Based Ranger™) del puerto de conexión y lo almacena en la variable *var*.

GetCalc CATALOG

GetCalc *var* Parte de un programa:
:
:Disp "Press Enter when ready"
:Pause
:GetCalc L1
:Disp "List L1 received"
:
:

Recupera un valor del puerto de conexión y lo almacena en la variable *var*. Se utiliza para la conexión de una unidad con otra.

Nota: Para obtener una variable desde otra unidad mediante el puerto de conexión, utilice **2nd** [VAR-LINK] en la otra unidad, con el fin de seleccionar y enviar la variable, o ejecute **SendCalc** en la misma.

 **GetCalc** *var*,*port*

Recupera un valor del puerto de enlace y lo almacena en la variable *var* de la TI-89 Titanium receptora.

Si no se ha especificado un puerto, o si el valor especificado es *port* = 0, la TI-89 Titanium espera recibir datos procedentes de cualquier puerto.

Si *port* = 1, la TI-89 Titanium espera recibir los datos del puerto USB.

Si *port* = 2, la TI-89 Titanium espera recibir los datos del puerto E/S.

getConfig() CATALOG

getConfig() ⇒ *Lista pares*

Devuelve una lista de atributos de la calculadora. El nombre del atributo se enumera primero, seguido por su valor.



```
getConfig() [ENTER]
{"Product Name" "Advanced
  Mathematics Software"
"Version" "2.00. 09/25/1999"
"Product ID" "03-1-4-68"
"ID #" "01012 34567 ABCD"
"Cert. Rev. #" 0
"Screen Width" 160
"Screen Height" 100
"Window Width" 160
"Window Height" 67
"RAM Size" 262132
"Free RAM" 197178
"Archive Size" 655360
"Free Archive" 655340}
```



```
getConfig() [ENTER]
{"Product Name" "Advanced
  Mathematics Software"
"Version" "2.00. 09/25/1999"
"Product ID" "01-1-4-80"
"ID #" "01012 34567 ABCD"
"Cert. Rev. #" 0
"Screen Width" 240
"Screen Height" 120
"Window Width" 240
"Window Height" 91
"RAM Size" 262144
"Free RAM" 192988
"Archive Size" 720896
"Free Archive" 720874}
```

Nota: Su pantalla puede presentar valores diferentes a los aquí mostrados. El atributo Cert. Rev. # aparece sólo si se ha adquirido e instalado software adicional en la calculadora.

getDate() CATALOG

getDate() ⇒ *lista*

Devuelve una lista con la fecha correspondiente al valor actual del reloj. La lista tiene el formato {*año, mes, día*}.

```
getDate() [ENTER] {2002 2 22}
```

getDenom() Menú MATH/Algebra/Extract

getDenom(*expresión1*) ⇒ *expresión*

Transforma la *expresión1* en otra equivalente que tiene como denominador el más sencillo posible, y después devuelve este denominador.

```
getDenom((x+2)/(y-3)) [ENTER] y - 3
getDenom(2/7) [ENTER] 7
getDenom(1/x+(y^2+y)/y^2) [ENTER] x * y
```

getDtFmt() CATALOG

getDtFmt() ⇒ *entero*

Devuelve un entero que representa el formato de fecha que hay definido en ese momento en el dispositivo.

Valores enteros:

- 1 = MM/DD/AA
- 2 = DD/MM/AA
- 3 = MM.DD.AA
- 4 = DD.MM.AA
- 5 = AA.MM.DD
- 6 = MM-DD-AA
- 7 = DD-MM-AA
- 8 = AA-MM-DD

getDtStr() CATALOG

getDtStr(*entero*) ⇒ *cadena*

Devuelve una cadena con la fecha actual en el formato de fecha actual. Por ejemplo, la cadena devuelta *28/09/02* representa el día 28 de septiembre de 2002 (cuando el formato de fecha está definido en DD/MM/AA).

Si introduce el entero opcional que corresponde a un formato de fecha, la cadena devuelve la fecha actual en el formato especificado.

Valores enteros opcionales:

- 1 = MM/DD/AA
- 2 = DD/MM/AA
- 3 = MM.DD.AA
- 4 = DD.MM.AA
- 5 = AA.MM.DD
- 6 = MM-DD-AA
- 7 = DD-MM-AA
- 8 = AA-MM-DD

getFold() CATALOG

getFold() ⇒ *nombreCadena*

Devuelve el nombre de la carpeta actual como una cadena.

getFold() **[ENTER]**

"main"

getFold()⇒oldfoldr **[ENTER]**

"main"

oldfoldr **[ENTER]**

"main"

getKey() CATALOG

getKey() ⇒ *entero*

Devuelve el código de la tecla que ha pulsado. Devuelve 0 si no ha pulsado ninguna tecla.

Las teclas con prefijo (mayús **[↑]**, segunda función **[2nd]**, opción **[*]**, alfabética **[alpha]** y arrastre **[@]**) no se reconocen por separado, aunque modifican los códigos de las teclas posteriores a ellas. Por ejemplo: **[*][X] ≠ [X] ≠ [2nd][X]**.

Para ver una lista de los códigos de teclas, consulte el anexo B.

Listado del programa:

```
:Disp
:Loop
: getKey()→key
: while key=0
:   getKey()→key
: EndWhile
: Disp key
: If key = ord("a")
:   Stop
:EndLoop
```

getMode() CATALOG

getMode(modoNombreCadena) ⇒ *cadena*

getMode("ALL") ⇒ *ListaCadenaPares*

Si el argumento es un nombre de modo concreto, devuelve una cadena con el estado actual de dicho modo.

Si el argumento es "ALL", devuelve una lista de los pares de cadenas que contienen los estados de todos los modos. Si quiere restablecer los estados de los modos más adelante, deberá almacenar el resultado **getMode("ALL")** en una variable y, después, utilizar **setMode** para restablecer los modos.

Para ver una lista de los nombres de modos y sus posibles estados, consulte **setMode**.

Nota: Para definir o obtener información sobre el modo Unit System, utilice **setUnits()** o **getUnits()** en vez de **setMode()** o **getMode()**.

getMode("angle") [ENTER]	"RADIAN"
getMode("graph") [ENTER]	"FUNCTION"
getMode("all") [ENTER]	{ "Graph" "FUNCTION" "Display Digits" "FLOAT 6" "Angle" "RADIAN" "Exponential Format" "NORMAL" "Complex Format" "REAL" "Vector Format" "RECTANGULAR" "Pretty Print" "ON" "Split Screen" "FULL" "Split 1 App" "Home" "Split 2 App" "Graph" "Number of Graphs" "1" "Graph 2" "FUNCTION" "Split Screen Ratio" "1.1" "Exact/Approx" "AUTO" "Base" "DEC" }

Nota: Su pantalla puede presentar modos diferentes a los aquí mostrados.

getNum() Menú MATH/Algebra/Extract

getNum(*expresión1*) ⇒ *expresión*

Transforma la *expresión1* en otra equivalente que tiene como denominador el más sencillo posible, y devuelve su numerador.

getNum((x+2)/(y-3)) [ENTER]	x + 2
getNum(2/7) [ENTER]	2
getNum(1/(x+1/y)) [ENTER]	x + y

getTime() CATALOG

getTime() ⇒ *lista*

Devuelve una lista con la hora correspondiente al valor actual del reloj. La lista tiene el formato {*hora,minuto,segundo*}. La hora se devuelve con formato de 24 horas.

getTmFmt() CATALOG

getTmFmt() ⇒ *entero*

Devuelve un entero que representa el formato de hora del reloj que hay definido en ese momento en el dispositivo.

Valores enteros:
12 = reloj de 12 horas
24 = reloj de 24 horas

getTmStr() CATALOG

getTmStr(*entero*) ⇒ *cadena*

Devuelve una cadena con la hora actual del reloj en el formato de hora actual.

Si introduce el entero opcional que corresponde a un formato de hora del reloj, la cadena devuelve la hora actual en el formato especificado.

Valores enteros opcionales:
12 = reloj de 12 horas
24 = reloj de 24 horas

getTmZn() CATALOG

getTmZn() ⇒ *entero*

Devuelve un entero que representa la zona horaria que hay definida en ese momento en el dispositivo.

El entero devuelto representa el número de minutos de diferencia entre esa zona horaria y la hora media de Greenwich (GMT), establecida en Greenwich, Inglaterra. Por ejemplo, si la zona horaria se diferencia dos horas de la GMT, el dispositivo devuelve 120 (minutos).

Los enteros correspondientes a las zonas situadas al oeste de la GMT son negativos.

Los enteros correspondientes a las zonas situadas al este de la GMT son positivos.

Si la hora media de Greenwich es 14:07:07, son las:

8:07:07 a.m. en Denver, Colorado (hora diurna de las Montañas Rocosas)
(-360 minutos respecto a GMT)

16:07:07 p.m. en Bruselas, Bélgica (hora estándar de Europa central)
(+120 minutos respecto a GMT)

getType() CATALOG

getType(*var*) ⇒ *cadena*

Devuelve una cadena que indica el tipo de datos que hay en la variable *var*.

Si no se ha definido *var*, devuelve la cadena "NONE".

```
{1.2.3} → temp [ENTER] {1 2 3}
getType(temp) [ENTER] "LIST"
2+3 → temp [ENTER] 2 + 3/
getType(temp) [ENTER] "EXPR"
DelVar temp [ENTER] Done
getType(temp) [ENTER] "NONE"
```

Tipo de datos	Contenido de la variable
"ASM"	Programa de lenguaje ensamblador
"DATA"	Tipo de datos
"EXPR"	Expresión (incluye expresiones complejas/arbitrarias/no definidas, ∞, -∞, TRUE, FALSE, pi, e)
"FUNC"	Función
"GDB"	Base de datos de gráficos
"LIST"	Lista
"MAT"	Matriz
"NONE"	La variable no existe
"NUM"	Número real
"OTHER"	Datos diversos para uso futuro por parte de las aplicaciones de software
"PIC"	Imagen gráfica
"PRGM"	Programa
"STR"	Cadena
"TEXT"	Texto
"VAR"	Nombre de otra variable

getUnits() CATALOG

getUnits() ⇒ *lista*

Devuelve una lista de cadenas que contiene las unidades por omisión actuales de todas las categorías excepto constantes, temperatura, cantidad de sustancia, intensidad luminosa y aceleración. *lista* tiene la forma:

```
{ "sistema" "cat1" "unidad1" "cat2" "unidad2" ... }
```

La primera cadena da el sistema (SI, ENG/US o CUSTOM). Los pares de cadenas subsiguientes dan una categoría (como Longitud) y su unidad por omisión (como *_m* para metros).

Para establecer las unidades por omisión, utilice **setUnits()**.

getUnits() **[ENTER]**

```
{ "SI" "Area" "NONE"  
  "Capacitance" "_F"  
  "Charge" "_cou1"  
  ... }
```

Nota: Su pantalla puede presentar unidades por omisión diferentes a las aquí mostradas.

Goto CATALOG

Goto *Nombre de etiqueta*

Transfiere el control de un programa a la etiqueta *Nombre de etiqueta*.

Nombre de etiqueta debe estar definido en el mismo programa utilizando la instrucción **Lbl**.

Parte de un programa:

```
:  
:  
:0→temp  
:1→i  
:Lbl TOP  
: temp+i→temp  
: If i<10 Then  
: i+1→i  
: Goto TOP  
: EndIf  
:Disp temp  
:  
:
```

►Grad Menú CATALOG/MATH/Angle

► Grad *expresión*

Convierte una expresión a la medida de ángulo grados centesimales.

En el modo Angle, en grados:

```
1.5 ►Grad [ENTER]  
1.66667°
```

En el modo Angle, en radianes:

```
1.5 ►Grad [ENTER]  
95.493°
```

Graph CATALOG

Graph *expresión1* [, *expresión2*] [, *var1*] [, *var2*]

La función Smart Graph dibuja las gráficas de las expresiones o funciones utilizando el modo de representación gráfica actual.

A las expresiones introducidas con las órdenes **Graph** o **Table** se les asigna números de función cada vez mayores comenzando desde 1. Puede modificarlos o borrarlos uno por uno con las funciones de edición disponibles cuando se presenta la tabla pulsando **F4** Header. Se ignoran las funciones Y= actualmente seleccionadas.

Si omite un argumento opcional de *var*, **Graph** utiliza la variable independiente del modo de representación gráfica actual.

Nota: No todos los argumentos opcionales son válidos en todos los modos, debido a que nunca pueden utilizarse los cuatro argumentos a la vez.

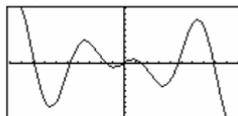
Algunas variaciones válidas de esta instrucción son:

Gráficas de funciones	Graph <i>expr</i> , <i>x</i>
Gráficas en paramétricas	Graph <i>xExpr</i> , <i>yExpr</i> , <i>t</i>
Gráficas en polares	Graph <i>expr</i> , θ
Gráficas de sucesiones	No admitidas.
Gráficas en 3D	Graph <i>expr</i> , <i>x</i> , <i>y</i>
Gráficas de ecuaciones diferenciales	No admitidas.

Nota: Utilice **ClrGraph** para borrar estas funciones o vaya a Y= Editor para activar nuevamente las funciones Y= del sistema.

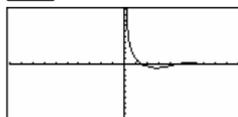
En el modo de gráficas de función y con la ventana ZoomStd:

Graph 1.25a*cos(a).a **ENTER**



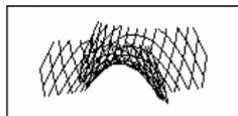
En el modo de gráficas en paramétricas y la ventana ZoomStd:

Graph time.2cos(time)/time.time **ENTER**



En el modo de representación gráfica en 3D:

Graph (v^2 - w^2)/4, v, w **ENTER**



Hex Menú MATH/Base

entero1 ►Hex ⇒ *entero*

256 ►Hex **ENTER**

0h100

Convierte el *entero1* en un número hexadecimal. Los números binarios o hexadecimales siempre tienen el prefijo 0b o 0h, respectivamente.

0b111100001111 ►Hex **ENTER**

0hF0F

┌ Cero, no la letra O, seguido por b o h.

0b *Número binario*

0h *Número hexadecimal*

┌ Los números binarios pueden tener hasta 32 dígitos; los hexadecimales, un máximo de

Sin un prefijo, el *entero1* se considera decimal (base 10). El resultado se muestra como hexadecimal, independientemente del estado del modo Base.

Si se introduce un entero decimal demasiado grande para una forma binaria de 32 bits con su correspondiente signo, se utiliza una operación de módulos simétricos para llevar el valor al rango apropiado.

identity() Menú MATH/Matrix

identity(*expresión*) ⇒ *matriz*

Devuelve la matriz de identidad de dimensión *expresión*.

expresión debe dar como resultado un entero positivo.

identity(4) **[ENTER]**

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

If CATALOG

If *enunciado de expresión booleana*

If *expresión booleana* **Then** *bloque*
Endif

Si *expresión booleana* es verdadera, ejecuta el *enunciado* único o el bloque de enunciados *bloque* antes de continuar con la ejecución.

Si *expresión booleana* es falsa, continúa la ejecución sin ejecutar el *enunciado* o el bloque de enunciados.

bloque puede ser un único enunciado o una sucesión de varios enunciados separados por el carácter ":".

Parte de un programa:

```
:  
:  
:If x<0  
:Disp "x is negative"  
:  
:-o-  
:  
:  
:If x<0 Then  
: Disp "x is negative"  
: abs(x)→x  
:EndIf  
:  
:
```

If *expresión booleana* **Then** *bloque1*

Else

bloque2

Endif

Si *expresión booleana* es verdadera, ejecuta el *bloque1* y se salta el *bloque2*.

Si la *expresión booleana* es falsa, pasa por alto el *bloque1* y ejecuta el *bloque2*.

bloque1 y *bloque2* pueden tener un solo enunciado.

Parte de un programa:

```
:  
:  
:If x<0 Then  
: Disp "x is negative"  
: Else  
: Disp "x is positive or zero"  
:EndIf  
:  
:
```

If *expresión booleana1* **Then** *bloque1*

Elseif *expresión booleana2* **Then** *bloque2*

⋮

Elseif *expresión booleanaN* **Then** *bloqueN*

Endif

Permite la ramificación de un programa. Si la *expresión booleana1* es verdadera, ejecuta el *bloque1*. Si la *expresión booleana1* es falsa, calcula la *expresión booleana2*, etc.

Parte de un programa:

```
:  
:  
:If choice=1 Then  
: Goto option1  
: Elseif choice=2 Then  
: Goto option2  
: Elseif choice=3 Then  
: Goto option3  
: Elseif choice=4 Then  
: Disp "Exiting Program"  
: Return  
:EndIf  
:  
:
```

imag() Menú MATH/Complex

imag(*expresión*) ⇒ *expresión*

imag(*expresión*) devuelve la parte imaginaria del argumento.

Nota: Todas las variables no definidas se tratan como variables reales. Consulte además **real**().

imag(1+2*i*) **[ENTER]**

2

imag(*z*) **[ENTER]**

0

imag(*x*+*iy*) **[ENTER]**

y

imag(*lista*) ⇒ *lista*

Devuelve una lista de las partes imaginarias de los elementos.

imag({-3.4-*i*,*h*}) **[ENTER]**

{0 -1 1}

imag(matriz) \Rightarrow matriz

Devuelve una matriz con las partes imaginarias de los elementos.

imag([a,b:ĉ.d]) **ENTER**

$\begin{bmatrix} 0 & 0 \\ c & d \end{bmatrix}$

ImpDif() Menú MATH/Calculus, CATALOG

ImpDif(ecuación, Varindependiente, Vardependiente,orden) \Rightarrow expresión

impDif(x^2+y^2=100.x.y)**ENTER**

-x/y

donde el orden predeterminado es 1.

Calcula la derivada implícita de ecuaciones en las que una variable resulta implícitamente definida por los términos de la otra.

Input CATALOG

Input

Interrumpe el programa momentáneamente, presenta la pantalla Graph actual, y permite actualizar las variables x_c e y_c (además de r_c y θ_c en el modo de coordenadas polares), con el cursor gráfico.

Al pulsar **ENTER**, se reanuda el programa.

Parte de un programa:

```
⋮  
:● Get 10 points from the Graph  
: Screen  
:For i,1,10  
: Input  
: xc→XLISTA[i]  
: yc→YLISTA[i]  
:EndFor  
⋮
```

Input [promptCadena,] var

Input [promptCadena], var interrumpe el programa momentáneamente, muestra *promptCadena* en la pantalla Program I/O, espera a que se introduzca una expresión, y almacena dicha expresión en *var*.

Si omite *promptCadena*, aparece el indicador "?".

Parte de un programa:

```
⋮  
:For i,1,9,1  
: "Enter x" & string(i)→str1  
: Input str1.#(right(str1,2))  
:EndFor  
⋮
```

InputStr CATALOG

InputStr [promptCadena,] var

Interrumpe el programa momentáneamente, presenta *promptCadena* en la pantalla Program I/O, espera a que se introduzca una respuesta, y la almacena en forma de cadena en *var*.

Si omite *promptCadena*, aparece el indicador "?".

Nota: La diferencia entre **Input** e **InputStr** es que **InputStr** siempre almacena el resultado como un cadena, por lo que no se necesitan las comillas (" ").

Parte de un programa:

```
⋮  
:InputStr "Enter Your Name".str1  
⋮
```

inString() Menú MATH/String

inString(srcCadena, subCadena, inicio) \Rightarrow entero

Devuelve la posición del carácter en la cadena *srcCadena* con el que empieza la cadena *subCadena*.

El *inicio*, si se incluye, especifica la posición del carácter en *srcCadena* en que comenzará la búsqueda. El valor por omisión = 1 (el primer carácter de *srcCadena*).

Si *srcCadena* no contiene *subCadena* o si *inicio* es mayor que *srcCadena*, devuelve un cero.

inString("Hello there","the")
ENTER

7

```
"ABCEFG"→s1:If inString(s1,  
"D")=0:Disp "D not found."ENTER  
D not found.
```

int()**CATALOG****int**(*expresión*) ⇒ entero**int**(*lista1*) ⇒ lista**int**(*matriz1*) ⇒ matrizint(-2.5) **ENTER** -3.int([-1.234,0.0.37]) **ENTER** [-2. 0 0.]

Devuelve el mayor número entero menor o igual que un argumento. Esta función es idéntica a **floor()**.

El argumento puede ser un número real o complejo.

En una lista o matriz, devuelve el mayor entero de cada uno de los elementos.

intDiv()**CATALOG****intDiv**(*número1*, *número2*) ⇒ entero**intDiv**(*lista1*, *lista2*) ⇒ lista**intDiv**(*matriz1*, *matriz2*) ⇒ matrizintDiv(-7,2) **ENTER** -3intDiv(4,5) **ENTER** 0intDiv({12,-14,-16},{5,4,-3}) **ENTER** {2 -3 5}

Devuelve el número entero correspondiente a argumento 1 dividido entre argumento 2.

En listas y matrices, devuelve el número entero correspondiente a argumento 1 dividido entre argumento 2, para cada par de elementos.

integrateConsulte **f()**, página 209.**iPart()****Menú MATH/Number****iPart**(*número*) ⇒ entero**iPart**(*lista1*) ⇒ lista**iPart**(*matriz1*) ⇒ matriziPart(-1.234) **ENTER** -1.iPart({3/2,-2.3,7.003}) **ENTER** {1 -2. 7.}

Devuelve el número entero de un argumento.

En listas y matrices, devuelve el número entero de cada elemento.

El argumento puede ser un número real o complejo.

isArchiv()**CATALOG****isArchiv**(*nombre_de_variable*) ⇒ true,falseisArchiv(PROG1) **ENTER** True

Determina si el *nombre_de_variable* está archivado o no. Devuelve true si *nombre_de_variable* está archivado. Devuelve false si *nombre_de_variable* no está archivado.

isClkOn()**CATALOG****isClkOn**() ⇒ true,false

Determina si el reloj está activado o desactivado. Devuelve true si el reloj está activado (ON). Devuelve false si el reloj está desactivado (OFF).

isLocked() CATALOG

isLocked(*nombre_de_variable*) ⇒ *true,false*

isLocked(PROG1)

False

Determina si *nombre_de_variable* está bloqueado o no. Devuelve true si *nombre_de_variable* está bloqueado. Devuelve false si *nombre_de_variable* no está bloqueado ni archivado.

isPrime() Menú MATH/Test

isPrime(*número*) ⇒ *Expresión booleana constante*

isPrime(5)

true

Devuelve verdadero o falso para indicar si *número* es un número primo ≥ 2.

isPrime(6)

false

Si *número* es mayor de aproximadamente 306 dígitos y no tiene factores ≤ 1021, **isPrime**(*número*) muestra un mensaje de error.

Función para hallar el siguiente número primo posterior al número especificado:

Si sólo desea determinar si *número* es primo, utilice **isPrime()** en vez de **factor()**. Es mucho más rápido, en particular si *número* no es primo y tiene un segundo factor mayor que es mayor de aproximadamente cinco dígitos.

```
Define nextPrim(n)=Func:Loop:  
n+1→n:if isPrime(n):return n:  
EndLoop:EndFunc  Done  
nextPrim(7)  11
```

isVar() CATALOG

isVar(*nombre_de_variable*) ⇒ *true,false*

isArchiv(PROG1)

True

Determina si *nombre_de_variable* se ha utilizado. Devuelve true si *nombre_de_variable* ya existe. Devuelve false si *nombre_de_variable* no existe.

Item CATALOG

Item *elementoNombreCadena*

Consulte el ejemplo con **Custom**.

Item *elementoNombreCadena, etiqueta*

Sólo es válida dentro de un bloque **Custom...EndCustom** o **ToolBar...EndTBar**. Configura un elemento de un menú desplegable para poder pegar texto en la posición del cursor (**Custom**) o pegar una ramificación en una etiqueta (**ToolBar**).

Nota: La ramificación de una etiqueta no está permitida dentro de un bloque **Custom**.

Lbl CATALOG

Lbl *Nombre de etiqueta*

Parte de un programa:

Define en un programa una etiqueta con el nombre *Nombre de etiqueta*.

```
:  
:  
:Lbl lb11  
:InputStr "Enter password", str1  
:If str1≠password  
: Goto lb11  
:Disp "Welcome to ..."  
:  
:
```

Puede utilizar la instrucción **Goto** *Nombre de etiqueta* para transferir el control del programa a la instrucción situada justo después de la etiqueta.

Nombre de etiqueta debe cumplir los mismos requisitos que el nombre de una variable.

lcm() Menú MATH/Number**lcm**(*número1*, *número2*) ⇒ *expresión*

lcm(6.9) [ENTER] 18

lcm(*lista1*, *lista2*) ⇒ *lista*lcm({1/3, -14.16}, {2/15.7.5}) [ENTER]
{2/3 14 80}**lcm**(*matriz1*, *matriz2*) ⇒ *matriz*

Devuelve el mínimo común múltiplo de dos argumentos. La función **lcm** de dos fracciones es la **lcm** de sus numeradores dividido entre la **gcd** de sus denominadores. La función **lcm** de números fraccionarios en coma flotante es su producto.

En el caso de dos listas o matrices, devuelve el mínimo común múltiplo de los elementos correspondientes.

left() Menú MATH/String**left**(*Cadena origen* [, *num*]) ⇒ *cadena*

left("Hello".2) [ENTER] "He"

Devuelve el número de caracteres *num* más a la izquierda contenidos en la *Cadena origen*.

Si se omite *num*, devuelve la *Cadena origen* completa.

left(*lista1* [, *num*]) ⇒ *lista*

left({1.3, -2.4}.3) [ENTER] {1 3 -2}

Devuelve el número de elementos *num* más a la izquierda contenidos en la *lista1*.

Si se omite *num*, devuelve la *lista1* completa.

left(*comparación*) ⇒ *expresión*

left(x<3) [ENTER] x

Devuelve la parte izquierda de una ecuación o una desigualdad.

limit() Menú MATH/Calculus**limit**(*expresión1*, *var*, *punto*, *dirección*) ⇒ *expresión*

limit(2x+3.x.5) [ENTER] 13

limit(*lista1*, *var*, *punto*, *dirección*) ⇒ *lista*

limit(1/x.x.0.1) [ENTER] ∞

limit(*matriz1*, *var*, *punto*, *dirección*) ⇒ *matriz*

limit(sin(x)/x.x.0) [ENTER] 1

Devuelve el límite pedido.

dirección. negativa=por la izquierda, positiva=por la derecha, de otra manera =por ambos lados. Si se omite, la *dirección* es en ambos sentidos.

limit((sin(x+h)-sin(x))/h.h.0) [ENTER] cos(x)

limit((1+1/n)^n.n.∞) [ENTER] e

Los límites en $+\infty$ y $-\infty$ se toman como el límite lateral por la parte finita.

Según las circunstancias, **limit()** se devuelve sin calcular o devuelve undef cuando no puede calcular. Si determinar un único valor. Esto no significa que no existe el límite. undef significa que el resultado es un número no conocido finito o infinito, o un conjunto de números no conocidos.

limit() utiliza, por ejemplo, la regla de L'Hopital, por lo que hay límites que no puede calcular. Si *expresión1* contiene variables no definidas que no sean *var*, quizá sea necesario restringirlas para obtener un resultado más conciso.

limit(a^x.x.∞) [ENTER] undef

limit(a^x.x.∞)|a>1 [ENTER] ∞

limit(a^x.x.∞)|a>0 and a<1 [ENTER] 0

Los límites son muy sensibles a errores de redondeo. Evite el estado APPROX del modo Exact/Approx, y los números aproximados, al calcular los límites. De lo contrario, los límites igual a cero o infinito tomarían otro valor, al igual que los límites finitos y distintos de cero.

Line CATALOG

Line $xInicio, yInicio, xFin, yFin, modoDraw$

Presenta la pantalla Graph y dibuja, borra o invierte un segmento entre las coordenadas de ventana ($xInicio, yInicio$) y ($xFin, yFin$), incluyendo ambos extremos.

Si $modoDraw = 1$, dibuja la recta (por omisión).

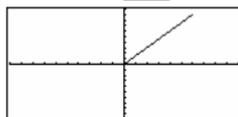
Si $modoDraw = 0$, desactiva la recta.

Si $modoDraw = -1$, desactiva la recta activada y viceversa (invierte los pixels de la misma).

Nota: Al dibujar la gráfica otra vez, se borran todos los elementos dibujados. Consulte además **PxlLine**.

Dibuje una recta y después bórrela en una ventana **ZoomStd**.

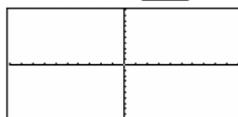
Line 0.0.6.9 **ENTER**



HOME

2nd **DEL** **2nd** **ENTER** [CALC HOME]

Line 0.0.6.9.0 **ENTER**



LineHorz CATALOG

LineHorz $y[, modoDraw]$

Presenta la pantalla Graph y dibuja, borra o invierte una recta horizontal de ordenada y .

Si $modoDraw = 1$, dibuja la recta (por omisión).

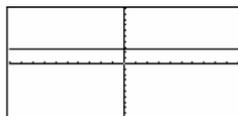
Si $modoDraw = 0$, desactiva la recta.

Si $modoDraw = -1$, desactiva la recta activada y viceversa (invierte los pixels de la misma).

Nota: Al dibujar la gráfica otra vez, se borran todos los elementos dibujados. Consulte además **PxlHorz**.

En la ventana **ZoomStd**:

LineHorz 2.5 **ENTER**



LineTan CATALOG

LineTan $expresión1, expresión2$

Presenta la pantalla Graph y dibuja una recta tangente a $expresión1$ en un punto determinado.

La $expresión1$ es una expresión o el nombre de una función en la que x es la variable independiente, mientras que la $expresión2$ es el valor de x en el punto de tangencia.

Nota: En el ejemplo, la gráfica de la $expresión1$ se dibuja por separado. **LineTan** no realiza la gráfica de la $expresión1$.

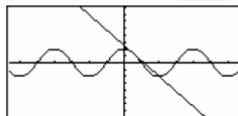
En el modo de gráficas de función y con la ventana **ZoomTrig**:

Graph cos(x)

HOME

2nd **DEL** **2nd** **ENTER** [CALC HOME]

LineTan cos(x). $\pi/4$ **ENTER**



LineVert CATALOG

LineVert $x[, modoDraw]$

Presenta la pantalla Graph y dibuja, borra o invierte una recta vertical de abscisa x .

Si $modoDraw = 1$, dibuja la recta (por omisión).

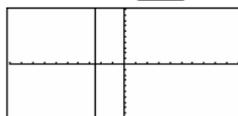
Si $modoDraw = 0$, desactiva la recta.

Si $modoDraw = -1$, desactiva la recta activada y viceversa (invierte los pixels de la misma).

Nota: Al volver a dibujar la gráfica, se borran todos los elementos dibujados. Consulte además **PxlVert**.

En la ventana **ZoomStd**:

LineVert -2.5 **ENTER**



LinReg Menú MATH/Statistics/Regressions

LinReg *lista1*, *lista2*, [*lista3*], [*lista4*, *lista5*]

Calcula la regresión lineal y actualiza todas las variables estadísticas del sistema.

Todas las listas deben tener el mismo tamaño, excepto la *lista5*.

La *lista1* representa *x*lista.

La *lista2* representa *y*lista.

La *lista3* representa la frecuencia.

La *lista4* representa códigos de categoría.

La *lista5* representa la lista de categorías.

Nota: Desde la *lista1* hasta la *lista4* deben ser un nombre de variable o c1–c99 (columnas de la última variable de datos mostrada en Data/Matrix Editor). La *lista5* no tiene que ser un nombre de variable y no puede ser c1–c99.

En el modo de gráficas de función:

{0,1,2,3,4,5,6} → L1 [ENTER] {0 1 2 ...}

{0,2,3,4,3,4,6} → L2 [ENTER] {0 2 3 ...}

LinReg L1,L2 [ENTER] Done

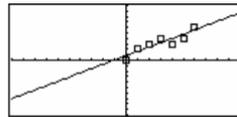
ShowStat [ENTER]



[ENTER] Regeq(x) → y1(x) [ENTER] Done

NewPlot 1,1,L1,L2 [ENTER] Done

[GRAPH]



Δlist() MATH/List menu

list(*lista1*) ⇒ *lista*

Devuelve una lista con las diferencias entre elementos consecutivos de la *lista1*. Cada elemento de la *lista1* se sustrae del siguiente elemento de la *lista1*. La lista resultante siempre tiene un elemento menos que la *lista1* original.

Δlist({20,30,45,70}) [ENTER] {10,15,25}

list→mat() Menú MATH/List

list→mat(*lista*, *elementosPorFila*) ⇒ *matriz*

Devuelve una matriz constituida fila por fila con los elementos de la *lista*.

elementosPorFila, si se incluye, especifica el número de elementos en cada fila. Por omisión, es el número de elementos en la *lista* (una fila).

Si la *lista* no llena por completo la matriz resultante, se añaden ceros.

list→mat({1,2,3}) [ENTER] [1 2 3]

list→mat({1,2,3,4,5},2) [ENTER] [1 2
3 4
5 0]

ln Menú MATH/String

ln *expresión* ⇒ *expresión*

Convierte la expresión de entrada en una expresión que sólo contiene logaritmos naturales (ln).

Log(x) ln [ENTER]

ln(x)
ln(10)

In()Tecla **2nd** [LN]

Tecla [LN]

In(expresión) ⇒ expresión**In**(lista) ⇒ lista

Devuelve el logaritmo neperiano de un argumento.

En una lista, devuelve los logaritmos neperianos de los elementos.

ln(2.0) **ENTER** .693...Si el modo Complex Format es **REAL**:ln({-3.1.2.5}) **ENTER**
Error: Non-real result

Si el modo Complex Format es

RECTANGULAR:ln({-3.1.2.5}) **ENTER**
{ln(3) + π • i .182... ln(5)}**In**(Matriz cuadrada1) ⇒ Matriz cuadradaDevuelve la matriz logaritmo neperiano de la *Matriz cuadrada1*. Esto *no* es lo mismo que calcular el logaritmo neperiano de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.*Matriz cuadrada1* debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

En el modo Angle en radianes, y en el modo de formato complejo rectangular:

ln([1.5;3;4;2.1;6; -2.1]) **ENTER**
$$\begin{bmatrix} 1.831...+1.734... \cdot i & .009...-1.490... \cdot i & ... \\ .448...-.725... \cdot i & 1.064...+.623 \cdot i & ... \\ -.266...-2.083... \cdot i & 1.124...+1.790... \cdot i & ... \end{bmatrix}$$
LnRegMenú **MATH/Statistics/Regressions****LnReg** lista1, lista2 [, lista3] [, lista4, lista5]

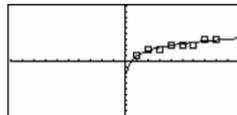
Calcula la regresión logarítmica y actualiza todas las variables estadísticas del sistema.

Todas las listas deben tener el mismo tamaño, excepto la *lista5*.La *lista1* representa xlista.La *lista2* representa ylista.La *lista3* representa la frecuencia.La *lista4* representa códigos de categoría.La *lista5* representa la lista de categorías.**Nota:** Desde la *lista1* hasta la *lista4* deben ser un nombre de variable o c1–c99 (columnas en la última variable de datos mostrada en Data/Matrix Editor). La *lista5* no tiene que ser un nombre de variable y no puede ser c1–c99.

En el modo de gráficas de función:

{1.2.3.4.5.6.7.8} → L1 **ENTER** {1 2 3 ...}{1.2.2.3.3.3.4.4} → L2 **ENTER** {1 2 2 ...}LnReg L1.L2 **ENTER** DoneShowStat **ENTER****ENTER** DoneRegeq(x) → y1(x) **ENTER** DoneNewPlot 1.1.L1.L2 **ENTER** Done

▶ [GRAPH]

**Local****CATALOG****Local** var1 [, var2] [, var3] ...Establece las variables *var* como variables locales. Estas variables existen sólo durante la operación de un programa o una función, y se borran cuando terminan de ejecutarse.**Nota:** Las variables locales ahorran memoria debido a que existen sólo temporalmente. Además, no interfieren en ningún valor existente en las variables globales. Las variables locales deben utilizarse para bucles **For** y para almacenar valores temporalmente en una función de varias líneas, ya que una función no permite modificaciones en variables globales.

Listado del programa:

:prgname()
:Prgm
:Local x,y
:Input "Enter x".x
:Input "Enter y".y
:Disp x*y
:EndPrgmNota: *x* e *y* no existen una vez ejecutado el programa.

Lock	CATALOG		
Lock $var1[, var2] \dots$		{1,2,3,4}►L1 ENTER	{1,2,3,4}
Bloquea las variables. Esto impide borrar o cambiar por equivocación una variable sin emplear primero la instrucción para desbloquearla.		Lock L1 ENTER	Done
En el ejemplo, la variable L1 está bloqueada y no puede ser borrada ni modificada.		DelVar L1 ENTER	Error: Variable is locked or protected
Nota: Las variables pueden desbloquearse con la orden Unlock .			

log()	CATALOG		
log (<i>expresión1</i> [, <i>expresión2</i>]) ⇒ <i>expresión</i>		log(2.0) ENTER	.301...
log (<i>lista1</i> [, <i>expresión2</i>]) ⇒ <i>lista</i>		Si el modo Complex Format es REAL :	
Devuelve el logaritmo de <i>expresión2</i> en la base del argumento.		log({-3.1,2,5}) ENTER	Error: Non-real result
Para las listas, devuelve el logaritmo de <i>expresión2</i> en la base de los elementos.		Si el modo Complex Format es RECTANGULAR :	
Si se omite la expresión 2, se utiliza la base 10.		log({-3.1,2,5}) ENTER	$\left\{ \frac{\ln(3)}{\ln(10)} + \frac{\pi}{\ln(10)} \cdot i \cdot .079... \frac{\ln(5)}{\ln(10)} \right\}$
log (<i>Matriz cuadrada1</i>) ⇒ <i>Matriz cuadrada</i>		En el modo Angle en radianes, y en el modo de formato complejo rectangular:	
Devuelve el logaritmo de <i>expresión2</i> en la base de la <i>Matriz cuadrada1</i> . El resultado no es igual que calcular el logaritmo de <i>expresión2</i> en la base de cada elemento. Para obtener más información sobre el método de cálculo, consulte cos() .		log([1,5,3;4,2,1;6,-2,1]) ENTER	$\begin{bmatrix} .795...+.753... \cdot i & .003...-.647... \cdot i & \dots \\ .194...-.315... \cdot i & .462...+.270 \cdot i & \dots \\ -.115...-.904... \cdot i & .488...+.777... \cdot i & \dots \end{bmatrix}$
<i>Matriz cuadrada1</i> debe ser diagonalizable. El resultado contiene siempre números en coma flotante.			
log (<i>x.b</i>) ⇒ <i>expresión</i>		Log(10,3) - log(5,3) ENTER	Log ₅ (2)
log (<i>Matriz cuadrada1</i>) ⇒ <i>Matriz cuadrada</i>		Log(2,0,4) ENTER	.5
En una lista, devuelve el logaritmo de <i>expresión2</i> en la base de los elementos.			

logbase	Menú MATH/String		
<i>expresión</i> ► logbase (<i>expresión1</i>) ⇒ <i>expresión</i>		Log(10,3) - log(5,5)►logbase(5) ENTER	
Simplifica la expresión de entrada a una expresión según la base utilizada por <i>expresión1</i> .			$\frac{\log_5(30)}{\log_5(3)}$

Logistic Menú MATH/Statistics/Regressions

Logistic *lista1, lista2* [, *iteraciones*] [, *lista3*] [, *lista4, lista5*]

Calcula la regresión logística y actualiza todas las variables estadísticas del sistema.

Todas las listas deben tener el mismo tamaño excepto la *lista5*.

La *lista1* representa *x*lista.
La *lista2* representa *y*lista.
La *lista3* representa la frecuencia.
La *lista4* representa códigos de categoría.
La *lista5* representa la lista de categorías.

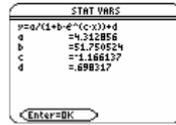
iteraciones especifica el número máximo de veces que se intenta obtener una solución. En caso de omitirse, se utiliza 64. Normalmente, los valores más grandes logran mayor precisión pero necesitan más tiempo de ejecución, y viceversa.

Nota: Desde la *lista1* hasta la *lista4* deben ser un nombre de variable o *c1*–*c99* (columnas de la última variable de datos mostrada en Data/Matrix Editor). La *lista5* no tiene que ser un nombre de variable y no puede ser *c1*–*c99*.

En el modo de gráficas de función:

```
{1.2,3.4,5.6}→L1 [ENTER] {1 2 3 ...}
{1.1,3.2,5.3,5.4,5.4,8}→L2 [ENTER]
{1 1.3 2.5 ...} Done
```

Logistic L1,L2 [ENTER]
ShowStat [ENTER]

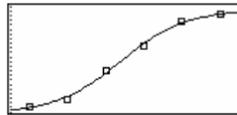


[ENTER]
regeq(x)→y1(x) [ENTER]
NewPlot 1,1,L1,L2 [ENTER]

Done
Done

• [GRAPH]

[F2] 9



Loop CATALOG

Loop
bloque
EndLoop

Ejecuta repetidamente los enunciados de *bloque*. Téngase en cuenta que el bucle se ejecuta indefinidamente, a menos que se ejecuten las instrucciones **Goto** o **Exit** en *bloque*.

bloque es una sucesión de enunciados separados por el carácter ":".

Parte de un programa:

```
:
:
:1→i
:Loop
: Rand(6)→die1
: Rand(6)→die2
: If die1=6 and die2=6
:   Goto End
: i+1→i
:EndLoop
:Lbl End
:Disp "The number of rolls is", i
:
```

LU Menú MATH/Matrix

LU *matriz, lMatNombre, uMatNombre, pMatNombre, tol*

Calcula la descomposición LU (inferior-superior) de Doolittle de una *matriz* real o compleja. La matriz triangular inferior se almacena en *lMatNombre*, la matriz triangular superior en *uMatNombre* y la matriz de permutación (que describe los intercambios de filas efectuadas durante el cálculo) en *pMatNombre*.

lMatNombre * *uMatNombre* = *pMatNombre* * *matriz*

De forma opcional, cualquier elemento de la matriz se considera cero si su valor absoluto es menor que *tol*. Esta tolerancia se utiliza sólo si la matriz tiene entradas de coma flotante y no contiene ninguna variable simbólica sin valor asignado. De no ser así, *tol* se ignora.

[6.12,18:5.14,31:3.8,18]→m1 [ENTER]

$$\begin{bmatrix} 6 & 12 & 18 \\ 5 & 14 & 31 \\ 3 & 8 & 18 \end{bmatrix}$$

LU m1,lower,upper,perm [ENTER] Done

lower [ENTER] $\begin{bmatrix} 1 & 0 & 0 \\ 5/6 & 1 & 0 \\ 1/2 & 1/2 & 1 \end{bmatrix}$

upper [ENTER] $\begin{bmatrix} 6 & 12 & 18 \\ 0 & 4 & 16 \\ 0 & 0 & 1 \end{bmatrix}$

perm [ENTER] $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

- Si se utiliza \square [ENTER] o se ajusta el modo a Exact/Approx=APPROXIMATE, los cálculos se llevan a cabo con aritmética de coma flotante.

- Si *tol* se omite o no se utiliza, la tolerancia por omisión se calcula como:

$$5e^{-14} * \max(\text{dim}(\text{matriz})) * \text{rowNorm}(\text{matriz})$$

El algoritmo de descomposición LU utiliza pivotación parcial con intercambios de filas.

$[m, n: o, p] \rightarrow m1$ [ENTER]	$\begin{bmatrix} m & n \\ o & p \end{bmatrix}$
LU m1, lower, upper, perm [ENTER]	Done
lower [ENTER]	$\begin{bmatrix} 1 & 0 \\ m & 1 \\ o & \end{bmatrix}$
upper [ENTER]	$\begin{bmatrix} o & p \\ 0 & n - \frac{m \cdot p}{o} \end{bmatrix}$
perm [ENTER]	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

mat▶data Menú MATH/List

mat▶data *mat, daos[, fila1][, col1][, fila2][, col2]*

Convierte una matriz en datos.

Cada argumento *[, fila1][, col1][, fila2][, col2]* se puede omitir de forma individual. Si se omite *fila1*, el valor predeterminado es 1. Si se omite *col1*, el valor predeterminado es 1. Si se omite *fila2*, el valor predeterminado es "fila máxima". Si el argumento omitido es *col2*, el valor predeterminado es "columna máxima".

mat▶data, m1, d1, 1, 1, 1 [ENTER]
Done

mat▶list() Menú MATH/List

mat▶list(*matriz*) \Rightarrow *lista*

Devuelve una lista constituida con los elementos de *matriz*. Los elementos se copian de la *matriz* fila por fila.

mat▶list([1, 2, 3]) [ENTER] (1 2 3)
[1, 2, 3:4, 5, 6]▶M1 [ENTER] $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$
mat▶list(M1) [ENTER] {1 2 3 4 5 6}

max() Menú MATH/List

max(*expresión1, expresión2*) \Rightarrow *expresión*

max(*lista1, lista2*) \Rightarrow *lista*

max(*matriz1, matriz2*) \Rightarrow *matriz*

Devuelve el máximo de dos argumentos. Si ambos argumentos son dos listas o matrices, devuelve una lista o matriz que contiene el valor máximo de cada par de elementos correspondientes.

max(2.3, 1.4) [ENTER] 2.3
max({1, 2}, {-4, 3}) [ENTER] {1 3}

max(*lista*) \Rightarrow *expresión*

Devuelve el elemento con el valor máximo que hay en la *lista*.

max({0, 1, -7, 1, 3, .5}) [ENTER] 1.3

max(*matriz1*) \Rightarrow *matriz*

Devuelve un vector fila que contiene el elemento máximo de cada columna de la *matriz1*.

max([1, -3, 7; -4, 0, .3]) [ENTER] \$
[1 0 7]

Nota: Consulte además **fMax()** y **min()**.

mean() Menú MATH/Statistics

mean(*lista*, *freclista*) \Rightarrow *expresión*

Devuelve la media de los elementos de la *lista*.

Cada elemento *freclista* cuenta el número de apariciones consecutivas del elemento correspondiente en la *lista*.

mean({.2,0.1,-.3,.4}) **[ENTER]** .26

mean({1,2,3},{3,2,1}) **[ENTER]** 5/3

mean(*matriz1*, *frecmatriz*) \Rightarrow *matriz*

Devuelve un vector fila con las medias de todas las columnas de la *matriz1*.

Cada elemento *frecmatriz* cuenta el número de apariciones consecutivas del elemento correspondiente en la *matriz1*.

En el modo de formato rectangular de vector:

mean([.2,0;-1,3;-.4,-.5]) **[ENTER]**
[-.133... .833...]

mean([1/5,0;-1,3;2/5,-1/2]) **[ENTER]**
[-2/15 5/6]

mean([1,2;3,4;5,6],[5,3;4,1;6,2]) **[ENTER]**
[47/15, 11/3]

median() Menú MATH/Statistics

median(*lista*) \Rightarrow *expresión*

Devuelve la mediana de los elementos de la *lista1*.

median({.2,0.1,-.3,.4}) **[ENTER]** .2

median(*matriz1*) \Rightarrow *matriz*

Devuelve un vector fila con las medianas de las columnas de *matriz1*.

median([.2,0;1,-.3;-.4,-.5]) **[ENTER]**
[.4 -.3]

Nota: Todas las entradas en la lista o matriz deben simplificarse a números.

MedMed Menú MATH/Statistics/Regressions

MedMed *lista1*, *lista2*, [*lista3*], [*lista4*, *lista5*]

Calcula la recta mediana-mediana y actualiza todas las variables estadísticas del sistema.

Todas las listas deben tener el mismo tamaño, excepto la *lista5*.

La *lista1* representa *xlista*.

La *lista2* representa *ylista*.

La *lista3* representa la frecuencia.

La *lista4* representa códigos de categoría.

La *lista5* representa la lista de categorías.

Nota: Desde la *lista1* hasta la *lista4* deben ser un nombre de variable o c1-c99 (columnas en la última variable de datos mostrada en Data/Matrix Editor). La *lista5* no tiene que ser un nombre de variable y no puede ser c1-c99.

En el modo de gráficas de función:

{0,1,2,3,4,5,6}**[ENTER]** L1 **[ENTER]** {0 1 2 ...}

{0,2,3,4,3,4,6}**[ENTER]** L2 **[ENTER]** {0 2 3 ...}

MedMed L1,L2 **[ENTER]** Done

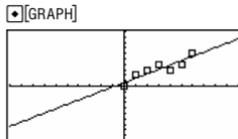
ShowStat **[ENTER]**



[ENTER] Regeq(x)**[ENTER]** y1(x) **[ENTER]** Done

NewPlot 1,1,L1,L2 **[ENTER]** Done

[GRAPH]



mid() Menú MATH/String**mid**(Cadena origen, inicio [, conteo]) ⇒ cadena

Devuelve *conteo* caracteres de la cadena de caracteres *Cadena origen*, comenzando en el número del carácter de *inicio*.

Si el *conteo* se omite o es mayor que la *Cadena origen*, devuelve todos los caracteres de la *Cadena origen*, comenzando en el número del carácter de *inicio*.

El *conteo* debe ser ≥ 0 . Si *conteo* = 0, devuelve una cadena vacía.

```
mid("Hello there".2) [ENTER] "ello there"
mid("Hello there".7.3) [ENTER] "the"
mid("Hello there".1.5) [ENTER] "Hello"
mid("Hello there".1.0) [ENTER] ""
```

mid(Lista origen, inicio [, conteo]) ⇒ lista

Devuelve *conteo* elementos de la *Lista origen*, comenzando en el número del elemento de *inicio*.

Si se omite el *conteo* o es mayor que la *Lista origen*, devuelve todos los elementos de *Lista origen*, comenzando en el número del elemento de *inicio*.

El *conteo* debe ser ≥ 0 . Si el *conteo* = 0, devuelve una lista vacía.

```
mid({9.8.7.6}.3) [ENTER] {7 6}
mid({9.8.7.6}.2.2) [ENTER] {8 7}
mid({9.8.7.6}.1.2) [ENTER] {9 8}
mid({9.8.7.6}.1.0) [ENTER] {}
```

mid(CadenaLista origen, inicio[, conteo]) ⇒ lista

Devuelve *conteo* cadenas de la lista *CadenaLista origen*, comenzando en el número del elemento de *inicio*.

```
mid({"A"."B"."C"."D"}.2.2) [ENTER] {"B" "C"}
```

min() Menú MATH/List**min**(expresión1, expresión2) ⇒ expresión**min**(lista1, lista2) ⇒ lista**min**(matriz1, matriz2) ⇒ matriz

Devuelve el mínimo de dos argumentos. Si los argumentos son dos listas o matrices, devuelve una lista o matriz que contiene el valor mínimo de cada par de elementos.

```
min(2.3.1.4) [ENTER] 1.4
min({1.2},{-4.3}) [ENTER] {-4 2}
```

min(lista) ⇒ expresión

Devuelve el elemento mínimo de la *lista*.

```
min({0.1.-7.1.3..5}) [ENTER] -7
```

min(matriz) ⇒ matriz

Devuelve un vector fila que contiene el elemento mínimo de cada columna en la *matriz1*.

```
min([1.-3.7;-4.0..3]) [ENTER] [-4 -3 .3]
```

Nota: Consulte además **fMin()** y **max()**.

mod() Menú MATH/Number**mod**(expresión1, expresión2) ⇒ expresión**mod**(lista1, lista2) ⇒ lista**mod**(matriz1, matriz2) ⇒ matriz

Devuelve el primer argumento con respecto al módulo de segundo argumento, según las identidades:

$$\text{mod}(x,0) = x$$

$$\text{mod}(x,y) = x - y \text{ floor}(x/y)$$

Cuando el segundo argumento no es cero, el resultado es periódico en dicho argumento. El resultado de esta función será cero o tendrá el mismo signo que el segundo argumento.

Si los argumentos son dos listas o dos matrices, devuelve una lista o matriz que contiene el módulo de cada par de elementos correspondientes.

Nota: Consulte además **remain()**.

mod(7.0) **[ENTER]** 7
 mod(7.3) **[ENTER]** 1
 mod(-7.3) **[ENTER]** 2
 mod(7.-3) **[ENTER]** -2
 mod(-7.-3) **[ENTER]** -1
 mod({12.-14.16}·{9.7.-5}) **[ENTER]**
 {3 0 -4}

MoveVar CATALOG**MoveVar** var, Carpeta antigua, Carpeta nueva

Mueve la variable *var* de *Carpeta antigua* a *Carpeta nueva*. Si *Carpeta nueva* no existe, **MoveVar** la crea.

{1.2.3.4} > L1 **[ENTER]** {1 2 3 4}
 MoveVar L1.Main.Games **[ENTER]** Done

mRow() Menú MATH/Matrix/Row ops**mRow**(expresión, matriz1, índice) ⇒ matriz

Devuelve una copia de la *matriz1* con cada elemento en la fila *índice* de *matriz1* multiplicado por *expresión*.

mRow(-1/3.[1.2:3.4].2) **[ENTER]**

$$\begin{bmatrix} 1 & 2 \\ -1 & -4/3 \end{bmatrix}$$

mRowAdd() Menú MATH/Matrix/Row ops**mRowAdd**(expresión, matriz1, índice1, índice2) ⇒ matriz

Devuelve una copia de la *matriz1* con cada elemento en la fila *índice2* de la *matriz1* sustituido por:

$$\text{expresión} \times \text{fila } \textit{índice1} + \text{fila } \textit{índice2}$$

mRowAdd(-3.[1.2:3.4].1.2) **[ENTER]**

$$\begin{bmatrix} 1 & 2 \\ 0 & -2 \end{bmatrix}$$

 mRowAdd(n.[a.b:c.d].1.2) **[ENTER]**

$$\begin{bmatrix} a & b \\ a \cdot n + c & b \cdot n + d \end{bmatrix}$$

nCr() Menú MATH/Probability**nCr**(expresión1, expresión2) ⇒ expresión

Siendo *expresión1* y *expresión2* números enteros con *expresión1* ≥ *expresión2* ≥ 0, **nCr()** es el número de combinaciones de los elementos de la *expresión1* tomados de *expresión2* en *expresión2*. También se denomina coeficiente binomial. Ambos argumentos pueden ser números enteros o expresiones simbólicas.

nCr(z.3) $\frac{z \cdot (z-2) \cdot (z-1)}{6}$
 ans(1)|z=5 10
 nCr(z.c) $\frac{z!}{c!(z-c)!}$
 ans(1)/nPr(z.c) $\frac{1}{c!}$

nCr(expresión, 0) ⇒ 1**nCr**(expresión, Entero neg) ⇒ 0

nCr(expresión, Entero pos) ⇒

$$\text{expresión} \cdot (\text{expresión} - 1) \dots (\text{expresión} - \text{Entero} + 1) / \text{Entero pos!}$$

nCr(expresión, no Entero) ⇒ $\text{expresión!} / ((\text{expresión} - \text{Entero})! \cdot \text{no Entero!})$

nCr(*lista1*, *lista2*) ⇒ *lista*

Devuelve una lista de combinaciones basada en los correspondientes pares de elementos de las dos listas. Los argumentos deben pertenecer a listas del mismo tamaño.

nCr({5.4.3}.{2.4.2}) **[ENTER]**

{10 1 3}

nCr(*matriz1*, *matriz2*) ⇒ *matriz*

Devuelve una matriz de combinaciones basada en los pares de elementos correspondientes de dos matrices. Los argumentos deben pertenecer a matrices del mismo tamaño.

nCr({6.5:4.3}.{2.2:2.2}) **[ENTER]**

$\begin{bmatrix} 15 & 10 \\ 6 & 3 \end{bmatrix}$

nDeriv() Menú MATH/Calculus

nDeriv(*expresión1*, *var*, *h*) ⇒ *expresión*

nDeriv(*expresión1*, *var*, *lista*) ⇒ *lista*

nDeriv(*lista*, *var*, *h*) ⇒ *lista*

nDeriv(*matriz*, *var*, *h*) ⇒ *matriz*

Devuelve la derivada numérica como una expresión. Utiliza la fórmula del cociente de diferencia central.

h es el valor del incremento. Si se omite, *h* es 0.001.

Cuando se usa una *lista* o *matriz*, se obtienen las expresiones correspondientes a cada uno de los elementos de la lista o matriz.

Nota: Consulte además **avgRC()** y **d()**.

nDeriv(cos(x).x.h) **[ENTER]**

$\frac{-\cos(x-h) - \cos(x+h)}{2h}$

limit(**nDeriv**(cos(x).x.h).h.0) **[ENTER]**

-sin(x)

nDeriv(x^3.x.0.01) **[ENTER]**

3. * (x^2 + .000033)

nDeriv(cos(x).x)|x=π/2 **[ENTER]**

-1.

nDeriv(x^2.x.{.01..1}) **[ENTER]**

{2. * x 2. * x}

NewData CATALOG

NewData *dataVar*, *lista1*[, *lista2*] [, *lista3*]...

Crema la variable de datos *Var datos*, en la que las columnas son las listas ordenadas.

Debe incluir al menos una lista.

lista1, *lista2*, ..., *listan* pueden ser listas como las mostradas en el ejemplo, expresiones que se transforman en listas o nombres de vector lista.

NewData hace que la nueva variable sea la actual de Data/Matrix Editor.

NewData mydata.{1.2.3}.{4.5.6} **[ENTER]**

Done

(Vaya a Data/Matrix Editor y abra *var* mydata para mostrar la variable de datos mostrada a continuación).

DATA	c1	c2	c3
1	1	4	
2	2	5	
3	3	6	
4			

NewData *Var datos*, *matriz*

Crema la variable de datos *Var datos* basada en *matriz*.

NewData *sysData*, *matriz*

Carga el contenido de *matriz* en la variable de datos del sistema *sysData*.

NewFold CATALOG

NewFold *Nombre de carpeta*

Crema una carpeta con el nombre *Nombre de carpeta*, y establece como carpeta actual dicha carpeta. Después de ejecutarse esta instrucción, se situará en la nueva carpeta.

NewFold games **[ENTER]**

Done

newList() CATALOG

newList(*númElementos*) ⇒ *lista*

Devuelve una lista de dimensión *númElementos*. Cada elemento es cero.

newList(4) **[ENTER]**

{0 0 0 0}

newMat() CATALOG

newMat(*númFilas*, *númColumnas*) ⇒ *matriz*

Devuelve una matriz de ceros de dimensión *númFilas* por *númColumnas*.

newMat(2,3) **ENTER** $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

NewPic CATALOG

NewPic *matriz*, *picVar* [, *máxFila*] [, *máxCol*]

Crema una variable *pic* *picVar* basada en la *matriz*. La *matriz* debe ser una matriz $n \times 2$ en la que cada fila represente un pixel. Las coordenadas del pixel comienzan en 0,0. Si *picVar* ya existe, **NewPic** la sustituye.

El valor por omisión de *picVar* es el área mínima requerida por los valores de la matriz. Los argumentos opcionales, *máxFila* y *máxCol*, determinan los límites máximos de *picVar*.

NewPic [1.1:2.2:3.3:4.4:5.5:5.1:4.2:2.4:1.5].xpic **ENTER** Done

Rc1Pic xpic **ENTER**



NewPlot CATALOG

NewPlot *n*, *tipo*, *xLista* [, *yLista*] [, *frecLista*] [, *catLista*] [, *incluir catLista*] [, *marca*] [, *Tamaño de cubo*]

Crema una nueva definición para el número de gráfico *n*.

tipo determina el tipo de gráfico.

- 1 = nube de puntos
- 2 = recta xy
- 3 = caja
- 4 = histograma
- 5 = gráfico modificado de caja

marca establece el tipo de marca mostrada.

- 1 = □ (caja)
- 2 = × (cruz)
- 3 = + (signo más)
- 4 = ■ (cuadrado)
- 5 = • (punto)

El *Tamaño de cubo* es el ancho de cada "barra" del histograma (*tipo* = 4), y varía según las variables de ventana *xmin* y *xmax*. *Tamaño de cubo* debe ser >0. Por omisión = 1.

Nota: *n* puede ser 1–9. Las listas deben ser nombres de variables o c1–c99 (columnas en la última variable de datos mostrada en Data/Matrix Editor), excepto *incluir catLista*, que no tiene que ser un nombre de variable y no puede ser c1–c99.

FnOff **ENTER** Done

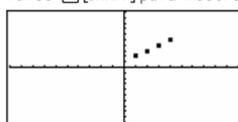
PlotsOff **ENTER** Done

{1,2,3,4} > L1 **ENTER** {1 2 3 4}

{2,3,4,5} > L2 **ENTER** {2 3 4 5}

NewPlot 1.1.L1.L2...4 **ENTER** Done

Pulse [GRAPH] para mostrar:



NewProb CATALOG

NewProb

NewProb

Done

Ejecuta diversas operaciones que permiten comenzar un nuevo problema después de un vaciado sin tener que reiniciar la memoria.

- Borra todos los nombres de variables de un solo carácter (Clear a-z) en la carpeta actual, a menos que las variables estén bloqueadas o archivadas.
- Desactiva todas las funciones y los gráficos estadísticos (**FnOff** y **PlotsOff**) en el modo gráfico actual.
- Ejecuta **ClrDraw**, **ClrErr**, **ClrGraph**, **ClrHome**, **ClrIO** y **ClrTable**.

nInt()

Menú MATH/Calculus

nInt(*expresión1*, *var*, *inferior*, *superior*) \Rightarrow *expresión*

Si la *expresión1* del integrando no contiene más variables que *var*, e *inferior* y *superior* son constantes, $+\infty$ o $-\infty$, **nInt()** devuelve un valor aproximado de $\int(\text{expresión1}, \text{var}, \text{inferior}, \text{superior})$. Este valor aproximado es un promedio ponderado de valores del integrando en el intervalo $\text{inferior} < \text{var} < \text{superior}$.

Se trata de conseguir que el resultado tenga, al menos, 6 dígitos significativos. El algoritmo termina cuando parece haberse obtenido el resultado o cuando parece que los valores adicionales no proporcionarán una mejora significativa.

Se presenta una advertencia ("Questionable accuracy") cuando no se ha obtenido el resultado.

Utilice **nInt()** anidados para realizar una integración numérica múltiple. Los límites de integración pueden depender de las variables de integración no incluidos en éstos.

Nota: Consulte además $\int()$.

nInt($e^{(-x^2)}.x$, -1.1)

1.493...

nInt($\cos(x)$, x , $-\pi$, $\pi+1$)

-1.041...E-12

$\int(\cos(x)$, x , $-\pi$, $\pi+10^{(-12)}$)

$-\sin\left(\frac{1}{1000000000000}\right)$

ans(1)

-1.E-12

nInt(**nInt**($e^{(-x*y)}/\sqrt{(x^2-y^2)}$, y , $-x$, x), x , 0.1)

3.304...

norm()

Menú MATH/Matrix/Norms

norm(*matriz*) \Rightarrow *expresión*

Devuelve la norma de un vector o matriz.

norm([a;b;c;d])

$\sqrt{a^2+b^2+c^2+d^2}$

norm([1.2;3.4])

$\sqrt{30}$

not Menú MATH/Test**not** *expresión booleana1* ⇒ *expresión booleana*Devuelve true, false o la *expresión booleana1* simplificada.

not 2>=3 [ENTER] true

not x<2 [ENTER] x ≥ 2

not not innocent [ENTER] innocent

not *entero1* ⇒ *entero*Devuelve el complemento a uno de un número entero real. De forma interna, *entero1* se convierte a un número binario de 32 bits con su correspondiente signo. El valor de cada bit se cambia (0 se convierte en 1 y viceversa) para el complemento a uno. Los resultados se presentan de acuerdo con el estado del modo Base.

Es posible introducir el entero en cualquier base de numeración. Para una entrada binaria o hexadecimal es necesario utilizar el prefijo 0b ó 0h, respectivamente. Sin un prefijo, el entero se trata como decimal (base 10).

Si se introduce un entero decimal demasiado grande para una forma binaria de 32 bits con su correspondiente signo, se utiliza una operación de módulos simétricos para llevar el valor al rango apropiado.

En el modo de base Hex:

not 0h7AC36 [ENTER] 0hFFF853C9

↳ **Importante:** Cero, no la letra O.

En el modo de base Bin:

0b100101 ▶dec [ENTER] 37

not 0b100101 [ENTER]

0b1111111111111111111111111111101010

ans(1) ▶dec [ENTER] -38

Nota: Las entradas binarias pueden tener hasta 32 dígitos (sin contar el prefijo 0b); las hexadecimales, un máximo de 8.**Nota:** Para escribir el operador de conversión ▶, pulse [2nd] [↵]. También puede seleccionar conversiones de base en el menú MATH/Base.**nPr()** Menú MATH/Probability**nPr**(*expresión1*, *expresión2*) ⇒ *expresión*Siendo *expresión1* y *expresión2* números enteros con *expresión1* ≥ *expresión2* ≥ 0, **nPr**() es el número de variaciones de los elementos de *expresión1* tomados de *expresión2* en *expresión2*. Ambos argumentos pueden ser números enteros o expresiones simbólicas.

nPr(z,3) [ENTER] z • (z-2) • (z-1)

ans(1) | z=5 [ENTER] 60

nPr(z, -3) [ENTER] $\frac{1}{(z+1) \cdot (z+2) \cdot (z+3)}$ nPr(z,c) [ENTER] $\frac{z!}{(z-c)!}$

ans(1) * nPr(z-c, -c) [ENTER] 1

nPr(*expresión*, 0) ⇒ 1**nPr**(*expresión*, *Entero neg*) ⇒ $\frac{1}{(expresión+1) \cdot (expresión+2) \dots (expresión - Entero neg)}$ **nPr**(*expresión*, *Entero pos*) ⇒ $\frac{expresión \cdot (expresión-1) \dots (expresión - Entero pos+1)}$ **nPr**(*expresión*, *no Entero*) ⇒ *expresión!* (*expresión* - *no Entero*)!**nPr**(*lista1*, *lista2*) ⇒ *lista*

Devuelve una lista de variaciones basada en los pares de elementos correspondientes de dos listas. Los argumentos deben pertenecer a listas del mismo tamaño.

nPr({5,4,3}, {2,4,2}) [ENTER] {20 24 6}

nPr(*matriz1*, *matriz2*) ⇒ *matriz*

Devuelve una matriz de variaciones basada en los pares de elementos correspondientes de dos matrices. Los argumentos deben pertenecer a matrices del mismo tamaño.

nPr([6,5,4,3], [2,2,2,2]) [ENTER] $\begin{bmatrix} 30 & 20 \\ 12 & 6 \end{bmatrix}$

nSolve() Menú MATH/Algebra

nSolve(*ecuación, varOEstim*) \Rightarrow *número de cadena_error*

Busca mediante iteraciones una única solución numérica real aproximada a la *ecuación* para su única variable. Especifique *varOGuess* como:

variable

- 0 -

variable = *número real*

Por ejemplo, tanto x como $x=3$ son válidos.

nSolve() suele resultar mucho más rápido que **solve()** o **zeros()**, sobre todo si se usa el operador "|" para restringir la búsqueda a un intervalo pequeño que contenga exactamente una solución simple.

nSolve() intenta determinar un punto donde el residuo sea cero o dos puntos relativamente cercanos en que el residuo tenga signos opuestos y su magnitud no sea excesiva. Si no puede alcanzarlo con un número modesto de puntos de muestra, devuelve el mensaje "no solution found."

Si utiliza **nSolve()** en un programa, puede usar **getType()** para comprobar un resultado numérico antes de usarlo en una expresión algebraica.

Nota: Véase también **cSolve()**, **cZeros()**, **solve()** y **zeros()**.

`nSolve(x^2+5x-25=9,x)` **[ENTER]**

3.844...

`nSolve(x^2=4,x=-1)` **[ENTER]**

- 2.

`nSolve(x^2=4,x=1)` **[ENTER]**

2.

Nota: Si hay varias soluciones, puede usar una estimación para encontrar una solución específica.

`nSolve(x^2+5x-25=9,x)|x<0` **[ENTER]**

- 8.844...

`nSolve(((1+r)^24-1)/r=26,r)|r>0 and r<.25` **[ENTER]**

.0068...

`nSolve(x^2=-1,x)` **[ENTER]**

"no solution found"

OneVar Menú MATH/Statistics

OneVar *lista1* [, *lista2*] [, *lista3*] [, *lista4*]

Calcula las estadísticas para una única variable y y actualiza todas las variables estadísticas del sistema.

Todas las *lista* deben tener el mismo tamaño, excepto la *lista4*.

La *lista1* representa x lista.

La *lista2* representa la frecuencia.

La *lista3* representa códigos de categoría.

La *lista4* representa la lista de categorías.

Nota: Desde la *lista1* hasta la *lista3* debe ser un nombre de variable o c1–c99 (columnas en la última variable de datos mostrada en Data/Matrix Editor). La *lista4* no tiene que ser un nombre de variable y no puede ser c1–c99.

`{0,2,3,4,3,4,6}` \rightarrow L1 **[ENTER]**

OneVar L1 **[ENTER]**

Done

ShowStat **[ENTER]**

STAT VARS	
\bar{x}	=3.142857
Σx	=22.
Σx^2	=98.
S_x	=1.864454
nStat	=7.
min	=0.
\bar{y}	=2.
maxStat	=3.

[<Enter=BK]

or Menú MATH/Test

expresión booleana1 **or** *expresión booleana2* \Rightarrow
expresión booleana

Devuelve true, false o la entrada simplificada.

Devuelve true si una o ambas expresiones son verdaderas. Devuelve false si ambas expresiones son falsas.

Nota: Consulte **xor**.

`x>3 or x>4` **[ENTER]**

$x \geq 3$

Parte de un programa:

```

:
:
If x<0 or x>5
  Goto END
:
:

```

```

If choice=1 or choice=2
  Disp "Wrong choice"
:
:

```

entero1 **or** *entero2* ⇒ *entero*

Compara dos números enteros reales bit a bit mediante una operación **or**. Internamente, ambos enteros se convierten a números binarios de 32 bits con su correspondiente signo. Cuando se comparan los bits correspondientes, el resultado es 1 si cualquier bit es 1; el resultado es 0 sólo si ambos bits son 0. El valor devuelto representa los bits que resultan y se presenta de acuerdo con el estado del modo Base.

Es posible introducir los números enteros en cualquier base de numeración. Para entradas binarias o hexadecimales, debe utilizarse el prefijo 0b ó 0h, respectivamente. Sin un prefijo, los enteros se tratan como decimales (base 10).

Si se introduce un entero decimal demasiado grande para una forma binaria de 32 bits con su correspondiente signo, se utiliza una operación de módulos simétricos para llevar el valor al rango apropiado.

Nota: Consulte **xor**.

En el modo de base Hex:

0h7AC36 or 0h3D5F **ENTER** 0h7BD7F

Importante: Cero, no la letra O.

En el modo de base Bin:

0b100101 or 0b100 **ENTER** 0b100101

Nota: Las entradas binarias pueden tener hasta 32 dígitos (sin contar el prefijo 0b); las hexadecimales, un máximo de 8.

ord() Menú MATH/String

ord(cadena) ⇒ *entero*

ord(lista) ⇒ *lista*

Devuelve el código numérico del primer carácter de *cadena*, o la lista con los primeros caracteres de cada elemento de la lista.

Consulte el anexo B para ver una lista de todos los códigos de caracteres.

ord("hello") **ENTER** 104

char(104) **ENTER** "h"

ord(char(24)) **ENTER** 24

ord({"alpha", "beta"}) **ENTER** {97 98}

Output CATALOG

Output *fila*, *columna*, *exprOCadena*

Presenta *exprOCadena* (una expresión o cadena de caracteres) en la pantalla Program I/O en las coordenadas (*fila*, *columna*).

Una expresión puede incluir operaciones de conversión tales como **DD** y **Rect**. También se puede utilizar el operador **▶** para ejecutar conversiones de bases de numeración y de unidades.

Si Pretty Print = ON, *exprOCadena* aparece en "pretty print".

En la pantalla Program I/O, se puede pulsar **F5** para mostrar la pantalla Home; un programa puede utilizar **DispHome**.

Parte de un programa:

```
:  
:RandSeed 1147  
:ClrIO  
:For i,1,90,10  
: Output i, rand(100)."Hello"  
:EndFor  
:
```

Resultado después de la ejecución:

```
      Hello  
Hello  Hello  
      Hello  Hello  
Hello  Hello  
      Hello
```

P>Rx() Menú MATH/Angle

P>Rx(*r*Expresión, *θ*Expresión) ⇒ expresión

P>Rx(*r*Lista, *θ*Lista) ⇒ lista

P>Rx(*r*Matriz, *θ*Matriz) ⇒ matriz

Devuelve la abscisa correspondiente al par (*r*, *θ*).

Nota: El argumento *θ* se interpreta como un ángulo expresado en grados, grados centesimales o radianes, según el modo de ángulo actual. Si el argumento es una expresión, puede utilizar $^{\circ}$, $^{\prime}$ o $^{\prime\prime}$ para anular temporalmente el modo de ángulo.

En el modo Angle, en radianes:

P>Rx(*r*, *θ*) **[ENTER]** $\cos(\theta) \cdot r$

P>Rx(4,60°) **[ENTER]** 2

P>Rx({-3,10,1.3},{π/3,-π/4.0})

[ENTER]

{-3/2 5·√2 1.3}

P>Ry() Menú MATH/Angle

P>Ry(*r*Expresión, *θ*Expresión) ⇒ expresión

P>Ry(*r*Lista, *θ*Lista) ⇒ lista

P>Ry(*r*Matriz, *θ*Matriz) ⇒ matriz

Devuelve la ordenada correspondiente al par (*r*, *θ*).

Nota: El argumento *θ* se interpreta como un ángulo expresado en grados, grados centesimales o radianes, según el modo de ángulo actual. Si el argumento es una expresión, puede utilizar $^{\circ}$, $^{\prime}$ o $^{\prime\prime}$ para anular temporalmente el modo de ángulo actual.

En el modo Angle, en radianes:

P>Ry(*r*, *θ*) **[ENTER]** $\sin(\theta) \cdot r$

P>Ry(4,60°) **[ENTER]** 2·√3

P>Ry({-3,10,1.3},{π/3,-π/4.0})

[ENTER]

{ $\frac{-3 \cdot \sqrt{3}}{2}$ -5·√2 0.}

part() CATALOG

part(*expresión1*, *Entero no negativo*)

Esta función de programación avanzada permite identificar y extraer todas las subexpresiones en el resultado simplificado de *expresión1*.

Por ejemplo, si la *expresión1* se simplifica a $\cos(\pi * x + 3)$:

- La función **cos()** tiene un argumento: $(\pi * x + 3)$.
- La suma de $(\pi * x + 3)$ tiene dos operandos: $\pi * x$ y 3.
- El número 3 no tiene argumentos u operandos.
- El producto $\pi * x$ tiene dos operandos: π y x .
- La variable x y la constante simbólica π no tiene argumentos u operandos.

Si x tiene un valor numérico y se pulsa **[]** **[ENTER]**, se calcula el valor numérico de $\pi * x$, el resultado se suma a 3 y, a continuación, se calcula el coseno. **cos()** es el operador de **nivel superior** debido a que es el **último** en aplicarse.

part(*expresión1*) ⇒ número

Simplifica la *expresión1* y devuelve el número de los argumentos u operandos de nivel superior. Devuelve 0 si la *expresión1* es un número, una variable o una constante simbólica tal como π , e , i , ∞ .

part($\cos(\pi * x + 3)$) **[ENTER]**

1

Nota: $\cos(\pi * x + 3)$ tiene un argumento.

part(*expresión1*, 0) ⇒ *cadena*

Simplifica la *expresión1* y devuelve una cadena que contiene el nombre de la función u operador de nivel superior. Devuelve la **string**(*expresión1*) si *expresión1* es un número, una variable o una constante simbólica tal como π , e , i , ∞ .

part(*expresión1*, *n*) ⇒ *expresión*

Simplifica la *expresión1* y devuelve el argumento u operando *n*-simo, donde $n > 0$ y \leq que el número de argumentos u operandos de nivel superior devueltos por **part**(*expresión1*). De no ser así, se obtiene un error.

`part(cos(π *x+3).0)` [ENTER] "cos"

`part(cos(π *x+3).1)` [ENTER] $3+\pi \cdot x$

Nota: La simplificación ha variado el orden del argumento.

Mediante la combinación de las variaciones de **part**(), se puede extraer todas las subexpresiones en el resultado simplificado de *expresión1*. Como se muestra en el ejemplo de la derecha, se puede almacenar un argumento u operando y, a continuación, utilizar **part**() para extraer más subexpresiones.

Nota: Cuando utilice **part**(), no confíe en ningún orden particular en sumas y en productos.

Expresiones tales como $(x+y+z)$ y $(x-y-z)$ se representan internamente como $(x+y)+z$ y $(x-y)-z$, lo que afecta a los valores devueltos por los argumentos primero y segundo. Existen razones técnicas por las que **part**($x+y+z,1$) devuelve $y+x$ en vez de $x+y$.

De forma similar, $x*y*z$ se representan internamente como $(x*y)*z$. De nuevo, existen razones técnicas por las que el primer argumento se devuelve como $y \cdot x$ en vez de $x \cdot y$.

Al extraer expresiones de una matriz debe recordar que las matrices se almacenan como listas de listas, como se muestra en el ejemplo de la derecha.

```
part(cos( $\pi$ *x+3)) [ENTER] 1
part(cos( $\pi$ *x+3).0) [ENTER] "cos"
part(cos( $\pi$ *x+3).1)→temp [ENTER]  $3+\pi \cdot x$ 
temp [ENTER]  $\pi \cdot x+3$ 
part(temp.0) [ENTER] "+"
part(temp) [ENTER] 2
part(temp.2) [ENTER] 3
part(temp.1)→temp [ENTER]  $\pi \cdot x$ 
part(temp.0) [ENTER] "*"
part(temp) [ENTER] 2
part(temp.1) [ENTER]  $\pi$ 
part(temp.2) [ENTER] x
```

```
part(x+y+z) [ENTER] 2
part(x+y+z.2) [ENTER] z
part(x+y+z.1) [ENTER] y+x
```

```
part(x*y*z) [ENTER] 2
part(x*y*z.2) [ENTER] z
part(x*y*z.1) [ENTER] y*x
```

```
part([a,b,c:x,y,z].0) [ENTER] "{"
part([a,b,c:x,y,z]) [ENTER] 2
part([a,b,c:x,y,z].2)→temp [ENTER]
temp [ENTER] {x y z}
part(temp.0) [ENTER] "{"
part(temp) [ENTER] 3
part(temp.3) [ENTER] z
delVar temp [ENTER] Done
```

En el programa de la derecha se usa **getType()** y **part()** para implementar parcialmente una diferenciación simbólica. El estudio y terminación de esta función puede ayudarle a aprender cómo se diferencia a mano. También puede incluir funciones que la TI-89 Titanium / Voyage™ 200 no puede diferenciar, como las funciones Bessel.

```
:d(y,x)
:Func
:Local f
:If getType(y)="VAR"
:  Return when(y=x,1,0,0)
:If part(y)=0
:  Return 0 Ⓣ y=π,∞,i,numbers
:part(y,0)→f
:If f="-" Ⓣ if negate
:  Return -d(part(y,1),x)
:If f="-" Ⓣ if minus
:  Return d(part(y,1),x)
:    -d(part(y,2),x)
:If f="+"
:  Return d(part(y,1),x)
:    +d(part(y,2),x)
:If f="*"
:  Return part(y,1)*d(part(y,2),x)
:    +part(y,2)*d(part(y,1),x)
:If f="{"
:  Return seq(d(part(y,k),x),
:    k,1,part(y))
:Return undef
:EndFunc
```

PassErr CATALOG

PassErr

Pasa un error al siguiente nivel.

Si "errornum" es cero, **PassErr** no realiza ninguna operación.

La cláusula **Else** del programa debe utilizar **ClrErr** o **PassErr**. Si se desea ignorar o procesar el error, debe utilizarse **ClrErr**. Si no sabe qué debe hacerse con el error, utilice **PassErr** para enviarlo al siguiente gestor de errores. Consulte además **ClrErr**.

Consulte el programa ejemplo de **ClrErr**

Pause CATALOG

Pause [expresión]

Suspende la ejecución de un programa. Si se incluye *expresión*, ésta se presenta en la pantalla Program I/O.

La *expresión* puede incluir operaciones de conversión tales como **DD** y **Rect**. También se puede utilizar el operador **▶** para ejecutar conversiones de bases de numeración y unidades.

Si el resultado de la *expresión* es demasiado grande como para caber en la pantalla, se puede utilizar la tecla del cursor para desplazarse por ésta.

La ejecución del programa se reanuda al pulsar **ENTER**.

Parte de un Programa:

```
:
:DelVar temp
:1→temp[1]
:1→temp[2]
:Disp temp[2]
:Ⓣ Guess the Pattern
:For i,3,20
:  temp[i-2]+temp[i-1]→temp[i]
:  Disp temp[i]
:  Disp temp. "Can you guess
:    the next number?"
:  Pause
:EndFor
:
```

PlotsOff CATALOG

PlotsOff [1] [, 2] [, 3] ... [, 9]

Desactiva la representación de las gráficas que se determinen. En el modo Two Graph, esto sólo afecta a la gráfica activa.

Si no hay parámetros, desactiva todas las gráficas.

PlotsOff 1.2.5 **ENTER**

Done

PlotsOff **ENTER**

Done

PlotsOn CATALOG

PlotsOn [1] [, 2] [, 3] ... [, 9]

Activa la representación de las gráficas que se determinen. En el modo Two Graph esto sólo afecta a la gráfica activa.

Si no incluye argumentos, activa todas las gráficas.

PlotsOn 2.4.5 **ENTER**

Done

PlotsOn **ENTER**

Done

►Polar Menú MATH/Matrix/Vector ops

vector ►Polar

Presenta el *vector* en forma polar [$r \angle \theta$]. El vector debe tener dos dimensiones y puede ser una lista o una matriz.

Nota: ►Polar es una instrucción del formato de visualización, no una función de conversión. Puede utilizarla sólo al final de una línea de entrada, y no actualiza ans.

Nota: Consulte además ►Rect.

[1.3.] ►Polar **ENTER**

[x.y] ►Polar **ENTER**

```

■ [1 3.]►Polar
  [3.16228 ∠ 1.24905]
■ [x y]►Polar
  [√(x² + y²) ∠ (π · sign(y) - tan⁻¹(x/y))]

```

Valor complejo ►Polar

Presenta el *Vector complejo* en forma polar.

- El modo Angle, en grados, devuelve ($r \angle \theta$).
- El modo Angle, en radianes, devuelve $re^{i\theta}$.

El *Valor complejo* puede tener cualquier forma compleja. No obstante, una entrada $e^{i\theta}$ causa error en el modo Angle en grados.

Nota: Para una entrada polar ($r \angle \theta$) debe utilizar paréntesis.

En el modo Angle, en radianes:

3+4i ►Polar **ENTER**

$e^{i \cdot (\frac{\pi}{2} - \tan^{-1}(3/4))} \cdot 5$

(4∠π/3) ►Polar **ENTER**

$e^{i \cdot \frac{\pi}{3}} \cdot 4$

En el modo Angle, en grados centesimales:

4i ►Polar **ENTER**

(4∠100)

En el modo Angle en grados:

3+4i ►Polar **ENTER**

(5∠90 - tan⁻¹(3/4))

polyEval() Menú MATH/List

polyEval(*lista1*, *expresión1*) ⇒ *expresión*

polyEval(*lista1*, *lista2*) ⇒ *expresión*

Obtiene el valor numérico del polinomio de coeficiente *lista1* para la indeterminada igual a *expresión1*.

polyEval({a,b,c}.x) **ENTER** $a \cdot x^2 + b \cdot x + c$

polyEval({1,2,3,4}.2) **ENTER** 26

polyEval({1,2,3,4},{2,-7})
ENTER {26 -262}

PopUp CATALOG

PopUp *elementoLista*, *var*

Presenta un menú desplegable que contiene las cadenas de caracteres de *elementoLista*, espera a que se seleccione un elemento, y almacena el número seleccionado en *var*.

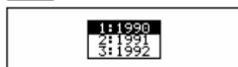
Los elementos de *elementoLista* deben ser cadenas de caracteres: {*elemento1Cadena*, *elemento2Cadena*, *elemento3Cadena*, ...}

Si *var* ya existe y tiene un número de elemento válido, dicho elemento se muestra como la opción por omisión.

elementoLista debe contener al menos una opción.

PopUp {"1990"."1991"."1992"}.var1

ENTER



PowerReg Menú MATH/Statistics/Regressions

PowerReg *lista1*, *lista2* [, *lista3*] [, *lista4*, *lista5*]

Calcula la regresión potencial y actualiza todas las variables estadísticas del sistema.

Todas las listas deben tener las mismas dimensiones excepto la *lista5*.

La *lista1* representa *x*lista.

La *lista2* representa *y*lista.

La *lista3* representa la frecuencia.

La *lista4* representa códigos de categoría.

La *lista5* representa la lista de categorías.

Nota: Desde la *lista1* hasta la *lista4* deben ser un nombre de variable o c1–c99 (columnas en la última variable de datos mostrada en Data/Matrix Editor). La *lista5* no tiene que ser un nombre de variable y no puede ser c1–c99.

En el modo de gráficas de función:

{1,2,3,4,5,6,7} → L1 **ENTER**

{1 2 3 ...}

{1,2,3,4,3,4,6} → L2 **ENTER**

{1 2 3 ...}

PowerReg L1,L2 **ENTER**

ShowStat **ENTER**

Done



ENTER

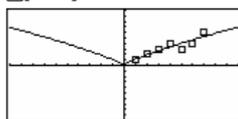
Regeq(x) → y1(x) **ENTER**

Done

NewPlot 1,1,L1,L2 **ENTER**

Done

▀ [GRAPH]



Prgm CATALOG

Prgm

⋮

EndPrgm

Instrucción requerida para identificar el comienzo de un programa. La última línea del programa debe ser **EndPrgm**.

Parte de un programa:

:prgname()

:Prgm

:

:EndPrgm

product() Menú MATH/List

product(*lista*[, *primero*[, *último*]]) \Rightarrow *expresión*

Devuelve el producto de los elementos contenidos en la *lista*.

product({1.2,3,4}) **[ENTER]** 24

product({2.x,y}) **[ENTER]** $2 \cdot x \cdot y$

product({4.5,8.9},2,3) **[ENTER]** 40

product(*matriz*[, *primero*[, *último*]]) \Rightarrow *matriz*

Devuelve un vector fila que contiene los productos de los elementos en la columna de la *matriz*1. *Primero* y *último* son opcionales, y especifican un rango de filas.

product([1.2,3,4,5,6,7,8,9]) **[ENTER]** [28

product([1.2,3,4,5,6,7,8,9],
1,2) **[ENTER]** [4,10,18]

Producto() Consulte #(), página 209.

Prompt CATALOG

Prompt *var*1[, *var*2] [, *var*3] ...

Presenta el indicador *var*1? en la pantalla Program I/O para cada variable de la lista de argumentos. Almacena la expresión que se introduzca en la variable correspondiente.

Prompt debe tener al menos un argumento.

Parte de un programa:

⋮
Prompt A,B,C
⋮
EndPrgm

propFrac() Menú MATH/Algebra

propFrac(*expresión*[, *var*]) \Rightarrow *expresión*

propFrac(*número_racional*) devuelve *número_racional* como la suma de un entero y una fracción irreducible con el mismo signo.

propFrac(*expresión_racional*, *var*) devuelve la suma de fracciones propias y un polinomio respecto a *var*. En *var*, el grado del denominador es superior al numerador en cada fracción propia. Se agrupan las potencias similares de *var*. Los términos y sus factores se clasifican con *var* como la variable principal.

Si se omite *var*, se realiza un desarrollo de las fracciones propias respecto a la variable principal. Los coeficientes de la parte polinómica se convierten en propios primero respecto a su variable principal, y así sucesivamente.

En expresiones racionales, **propFrac()** es más rápida pero menos exacta que **expand()**.

propFrac(4/3) **[ENTER]** $1 + 1/3$

propFrac(-4/3) **[ENTER]** $-1 - 1/3$

propFrac((x^2+x+1)/(x+1)+
(y^2+y+1)/(y+1),x) **[ENTER]**

$$\text{propFrac}\left(\frac{x^2+x+1}{x+1} + \frac{y^2+y+1}{y+1}\right)$$

propFrac(ans(1))

$$\text{propFrac}\left(\frac{1}{x+1} + x + \frac{y^2+y}{y+1}\right)$$

PtChg CATALOG

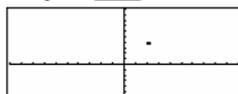
PtChg *x*, *y*

PtChg *x*Lista, *y*Lista

Presenta la pantalla Graph e invierte el pixel de la pantalla que está más cerca de las coordenadas (*x*, *y*).

Nota: **PtChg** hasta **PtText** muestran ejemplos similares continuos.

PtChg 2,4 **[ENTER]**

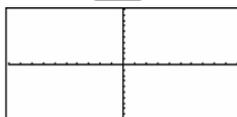


PtOff CATALOG

PtOff x, y
PtOff $xLista, yLista$

Presenta la pantalla Graph y desactiva el pixel en la pantalla que está más cerca de las coordenadas (x, y) .

PtOff 2.4 **ENTER**

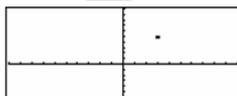


PtOn CATALOG

PtOn x, y
PtOn $xLista, yLista$

Presenta la pantalla Graph y activa el pixel en la pantalla que está más cerca de las coordenadas (x, y) .

PtOn 3.5 **ENTER**



ptTest() CATALOG

ptTest $(x, y) \Rightarrow$ expresión
ptTest $(xLista, yLista) \Rightarrow$ expresión booleana de constante

Devuelve true o false. Sólo devuelve true si está activado el pixel de la pantalla más cercano a las coordenadas (x, y) .

ptTest(3.5) **ENTER**

true

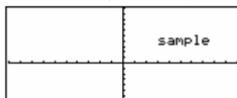
PtText CATALOG

PtText cadena, x, y

Presenta la pantalla Graph y coloca la cadena de caracteres *cadena* en el pixel de la pantalla más cercana a las coordenadas (x, y) especificadas.

La *cadena* se sitúa de forma que la esquina superior izquierda de su primer carácter se encuentre sobre las coordenadas.

PtText "sample".3.5 **ENTER**



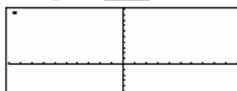
PxlChg CATALOG

PxlChg fila, col
PxlChg filaLista, colLista

Presenta la pantalla Graph e invierte el pixel en las coordenadas $(fila, col)$ del mismo.

Nota: Al volver a efectuar una representación gráfica, se borran todos los elementos dibujados.

PxlChg 2.4 **ENTER**



PxlCrc1 CATALOG

PxlCrc1 fila, col, r , modoDraw]

Presenta la pantalla Graph y dibuja una circunferencia centrada en las coordenadas $(fila, col)$ del pixel, con un radio de r pixels.

Si *modoDraw* = 1, dibuja la circunferencia (por omisión).

Si *modoDraw* = 0, desactiva la circunferencia.

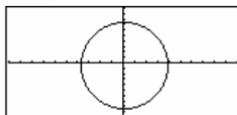
Si *modoDraw* = -1, invierte los pixels de la circunferencia.

Nota: Al volver a efectuar una representación gráfica, se borran todos los elementos dibujados. Consulte además **Circle**.

 PxlCrc1 40.80.30.1 **ENTER**

 PxlCrc1 50.125.40.1

ENTER



PxlHorz CATALOG

PxlHorz *fila* [, *modoDraw*]

Presenta la pantalla Graph y dibuja una recta horizontal en la posición del pixel de *fila*.

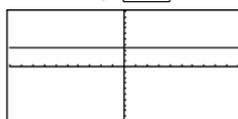
Si *modoDraw* = 1, dibuja la recta (por omisión).

Si *modoDraw* = 0, desactiva la recta.

Si *modoDraw* = -1, desactiva la recta activada y viceversa (invierte los pixels de la misma).

Nota: Al volver a efectuar una representación gráfica, se borran todos los elementos dibujados. Consulte además **LineHorz**.

PxlHorz 25.1 **[ENTER]**



PxlLine CATALOG

PxlLine *filalncio*, *collncio*, *filaFin*, *colFin* [, *modoDraw*]

Presenta la pantalla Graph y dibuja el segmento entre las coordenadas del pixel (*filalncio*, *collncio*) y (*filaFin*, *colFin*), incluyendo ambos extremos.

Si *modoDraw* = 1, dibuja la recta (por omisión).

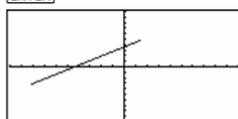
Si *modoDraw* = 0, desactiva la recta.

Si *modoDraw* = -1, desactiva la recta activada y viceversa (invierte los pixels de la misma).

Nota: Al volver a efectuar una representación gráfica se borran todos los elementos dibujados. Consulte además **Line**.

PxlLine 50.15.20.90.1 **[ENTER]**

PxlLine 80.20.30.150.1 **[ENTER]**



PxlOff CATALOG

PxlOff *fila*, *col*

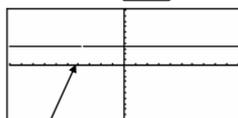
PxlOff *filaLista*, *colLista*

Presenta la pantalla Graph y desactiva el pixel de coordenadas (*fila*, *col*).

Nota: Al volver a efectuar una representación gráfica, se borran todos los elementos dibujados.

PxlHorz 25.1 **[ENTER]**

PxlOff 25.50 **[ENTER]**



25,50

PxlOn CATALOG

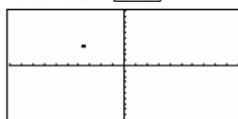
PxlOn *fila*, *col*

PxlOn *filaLista*, *colLista*

Presenta la pantalla Graph y activa el pixel de coordenadas (*fila*, *col*).

Nota: Al volver a efectuar una representación gráfica, se borran todos los elementos dibujados.

PxlOn 25.50 **[ENTER]**



pxlTest() CATALOG

pxlTest (*fila*, *col*) \Rightarrow *expresión booleana*

pxlTest (*filaLista*, *colLista*) \Rightarrow *expresión booleana*

Devuelve true si está activado el pixel de coordenadas (*fila*, *col*). Devuelve false si el pixel está desactivado.

Nota: Al volver a efectuar una representación gráfica, se borran todos los elementos dibujados.

PxlOn 25.50 **[ENTER]**

[HOME]

[CALC HOME]

PxlTest(25.50) **[ENTER]**

true

PxlOff 25.50 **[ENTER]**

[HOME]

[CALC HOME]

PxlTest(25.50) **[ENTER]**

false

PxlText CATALOG

PxlText *cadena, fila, col*

Presenta la pantalla Graph y coloca la cadena de caracteres *cadena* en la pantalla, empezando en las coordenadas de pixel (*fila, col*).

La *cadena* se sitúa con la esquina superior izquierda de su primer carácter en dichas coordenadas.

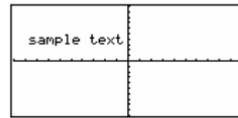
Nota: Al volver a efectuar una representación gráfica, se borran todos los elementos dibujados.

 PxlText "sample text".20.10



 PxlText "sample text".20.50





PxlVert CATALOG

PxlVert *col[, modoDraw]*

Dibuja una recta vertical en la posición *col* del pixel.

Si *modoDraw* = 1, dibuja la recta (por omisión).

Si *modoDraw* = 0, desactiva la recta.

Si *modoDraw* = -1, desactiva la recta que está activada y viceversa (invierte los pixels de la misma).

Nota: Al volver a efectuar una representación gráfica, se borran todos los elementos dibujados. Consulte además **LineVert**.

PxlVert 50.1 



QR Menú MATH/Matrix

QR *matriz, qMatNombre, rMatNombre[, tol]*

Calcula la factorización QR de la *matriz* real o compleja. Las matrices Q y R resultantes se almacenan en los *MatNombres* especificados. La matriz Q es unitaria. La matriz R es triangular superior.

De forma opcional, cualquier elemento de matriz se considera como cero si su valor absoluto es menor que *tol*. Esta tolerancia se utiliza sólo si la matriz tiene entradas de coma flotante y no contiene ninguna variable simbólica sin valor asignado. De no ser así, *tol* se ignora.

- Si se utiliza   o se ajusta el modo a Exact/Approx=APPROXIMATE, los cálculos se realizan mediante aritmética de coma flotante.

- Si *tol* se omite o no se utiliza, la tolerancia por omisión se calcula como:

$$5 \times 10^{-14} * \max(\dim(\text{matriz})) * \text{rowNorm}(\text{matriz})$$

La factorización QR se obtiene numéricamente con transformaciones Householder. La solución simbólica se obtiene mediante Gram-Schmidt. Las columnas de *qMatNombre* son los vectores de base ortonormal que abarcan el espacio definido por *matriz*.

El número de coma flotante (9.) en *m*1 ocasiona que los resultados se calculen en forma de coma flotante.

[1.2.3:4.5.6:7.8.9.]>m1 

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

QR m1,qm.rm 

Done

qm 

$$\begin{bmatrix} .123... & .904... & .408... \\ .492... & .301... & -.816... \\ .861... & -.301... & .408... \end{bmatrix}$$

rm 

$$\begin{bmatrix} 8.124... & 9.601... & 11.078... \\ 0. & .904... & 1.809... \\ 0. & 0. & 0. \end{bmatrix}$$

[m,n:o,p]>m1 

$$\begin{bmatrix} m & n \\ 0 & p \end{bmatrix}$$

QR m1,qm.rm 

Done

qm 

$$\begin{bmatrix} \frac{m}{\sqrt{m^2 + o^2}} & \frac{-\text{sign}(m \cdot p - n \cdot o) \cdot o}{\sqrt{m^2 + o^2}} \\ 0 & \frac{m \cdot \text{sign}(m \cdot p - n \cdot o)}{\sqrt{m^2 + o^2}} \end{bmatrix}$$

rm 

$$\begin{bmatrix} \sqrt{m^2 + o^2} & \frac{m \cdot n + o \cdot p}{\sqrt{m^2 + o^2}} \\ 0 & \frac{lm \cdot p - n \cdot ol}{\sqrt{m^2 + o^2}} \end{bmatrix}$$

QuadReg Menú MATH/Statistics/Regressions

QuadReg *lista1, lista2*, [*lista3*], [*lista4, lista5*]

Calcula una regresión polinómica de segundo grado y actualiza las variables estadísticas del sistema.

Todas las listas deben tener el mismo tamaño, excepto la *lista5*.

La *lista1* representa *x*lista.

La *lista2* representa *y*lista.

La *lista3* representa la frecuencia.

La *lista4* representa códigos de categoría.

La *lista5* representa la lista de categorías.

Nota: Desde la *lista1* hasta la *lista4* deben ser un nombre de variable o c1–c99 (columnas en la última variable de datos mostrada en Data/Matrix Editor). La *lista5* no tiene que ser un nombre de variable y no puede ser c1–c99.

En el modo de gráficas de función:

{0.1.2.3.4.5.6.7} → L1 [ENTER] {1 2 3 ...}

{4.3.1.1.2.2.3.3} → L2 [ENTER] {4 3 1 ...} Done

QuadReg L1,L2 [ENTER]

ShowStat [ENTER]

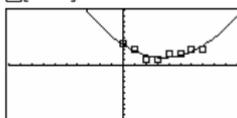


[ENTER]

Regeq(x) → y1(x) [ENTER] Done

NewPlot 1.1.L1,L2 [ENTER] Done

• [GRAPH]



QuartReg Menú MATH/Statistics/Regressions

QuartReg *lista1, lista2*, [*lista3*], [*lista4, lista5*]

Calcula una regresión polinómica de cuarto grado y actualiza las variables estadísticas del sistema.

Todas las listas deben tener el mismo tamaño, excepto la *lista5*.

La *lista1* representa *x*lista.

La *lista2* representa *y*lista.

La *lista3* representa la frecuencia.

La *lista4* representa códigos de categoría.

La *lista5* representa la lista de categorías.

Nota: Desde la *lista1* hasta la *lista4* deben ser un nombre de variable o c1–c99 (columnas en la última variable de datos mostrada en Data/Matrix Editor). La *lista5* no tiene que ser un nombre de variable y no puede ser c1–c99.

En el modo de gráficas de función:

{-2,-1,0.1.2.3.4.5.6} → L1 [ENTER] {-2 -1 0 ...}

{4.3.1.2.4.2.1.4.6} → L2 [ENTER] {4 3 1 ...} Done

QuartReg L1,L2 [ENTER]

ShowStat [ENTER]

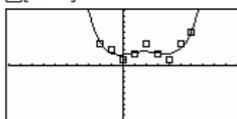


[ENTER]

Regeq(x) → y1(x) [ENTER] Done

NewPlot 1.1.L1,L2 [ENTER] Done

• [GRAPH]



R►Pθ() Menú MATH/Angle

R►Pθ (*x*Expresión, *y*Expresión) ⇒ expresión

R►Pθ (*x*Lista, *y*Lista) ⇒ lista

R►Pθ (*x*Matriz, *y*Matriz) ⇒ matriz

Devuelve la coordenada θ correspondiente al par (x , y).

Nota: El resultado se devuelve como un ángulo expresado en grados, grados centesimales o radianes, según el valor de modo de ángulo actual.

En el modo Angle, en grados:

R►Pθ(x , y) **ENTER**

```
■ R►Pθ(x,y)
  90·sign(y) - tan⁻¹( $\frac{x}{y}$ )
```

En el modo Angle, en grados centesimales:

R►Pθ(x , y) **ENTER**

```
■ R►Pθ(x,y)
  100·sign(y) - tan⁻¹( $\frac{x}{y}$ )
```

En el modo Angle, en radianes:

R►Pθ(3,2) **ENTER**

R►Pθ([3,-4.2],[0.π/4,1.5]) **ENTER**

```
■ R►Pθ(3,2)          tan⁻¹(2/3)
■ R►Pθ([3 -4 2],[θ  $\frac{\pi}{4}$  1.5])
  [θ tan⁻¹( $\frac{16}{\pi}$ ) +  $\frac{\pi}{2}$  .643501]
```

R►Pr() Menú MATH/Angle

R►Pr (*x*Expresión, *y*Expresión) ⇒ expresión

R►Pr (*x*Lista, *y*Lista) ⇒ lista

R►Pr (*x*Matriz, *y*Matriz) ⇒ matriz

Devuelve la coordenada r correspondiente al par (x , y).

En el modo Angle, en radianes:

R►Pr(3,2) **ENTER**

R►Pr(x , y) **ENTER**

R►Pr([3,-4.2],[0.π/4,1.5]) **ENTER**

```
■ R►Pr(3,2)          √13
■ R►Pr(x,y)          √x²+y²
■ R►Pr([3 -4 2],[θ  $\frac{\pi}{4}$  1.5])
  [3  $\frac{\sqrt{\pi^2+256}}{4}$  2.5]
```

►Rad Menú CATALOG/MATH/Angle

►Rad *expresión*

Convierte una expresión en una medida de ángulo en radianes.

En el modo Angle, en grados:

1.5 ►Rad **ENTER** .02618^R

En el modo Angle, en grados centesimales:

1.5 ►Rad **ENTER**
.023562^R

rand() Menú MATH/Probability

rand(n) ⇒ expresión

n es un entero ≠ cero.

Sin ningún parámetro, devuelve un número aleatorio entre 0 y 1. Cuando el argumento es positivo, devuelve un número entero aleatorio del intervalo [1, n]. Cuando el argumento es negativo, devuelve un número entero aleatorio del intervalo [$-n$, -1].

RandSeed 1147 **ENTER**

Done

↶ (Establece una nueva serie de números aleatorios).

rand() **ENTER** 0.158...

rand(6) **ENTER** 5

rand(-100) **ENTER** -49

randMat() Menú MATH/Probability

randMat(*númFilas*, *númColumnas*) ⇒ *matriz*

Devuelve una matriz de números enteros entre -9 y 9 del tamaño que se determine.

Ambos argumentos deben simplificarse en enteros.

RandSeed 1147

Done

randMat(3,3)

$$\begin{bmatrix} 8 & -3 & 6 \\ -2 & 3 & -6 \\ 0 & 4 & -6 \end{bmatrix}$$

Nota: Los valores de esta matriz cambian cada vez que pulsa .

randNorm() Menú MATH/Probability

randNorm(*media*, *sd*) ⇒ *expresión*

Devuelve un número decimal a partir de la distribución normal indicada. Puede ser cualquier número real, aunque estará distribuido, sobre todo, en el intervalo [*media*-3**sd*, *media*+3**sd*].

RandSeed 1147

Done

randNorm(0,1)

0.492...

randNorm(3,4.5)

-3.543...

randPoly() Menú MATH/Probability

randPoly(*var*, *orden*) ⇒ *expresión*

Devuelve un polinomio en *var* del orden que se determine. Los coeficientes son enteros aleatorios en el rango de -9 hasta 9. El coeficiente inicial no podrá ser cero.

El *orden* debe estar comprendido entre 0 y 99.

RandSeed 1147

Done

randPoly(x,5)

$-2 \cdot x^5 + 3 \cdot x^4 - 6 \cdot x^3 + 4 \cdot x - 6$

RandSeed Menú MATH/Probability

RandSeed *número*

Si *número* = 0, establece los orígenes en los valores por omisión del generador de número aleatorio. Si *número* ≠ 0, se utiliza para generar dos inicios que se almacenan en las variables del sistema seed1 y seed2.

RandSeed 1147

Done

rand()

0.158...

RclGDB CATALOG

RclGDB *GDBvar*

RclGDB *GDBvar*

Done

Restaura todos los estados almacenados en la variable de la base de datos gráfica *GDBvar*.

Para ver una lista de los estados, consulte **StoGDB**.

Nota: Es necesario haber guardado algo en *GDBvar* antes de restaurarlo.

RclPic CATALOG

RclPic *picVar* [, *fila*, *columna*]

Muestra la pantalla Graph y añade la imagen almacenada en *picVar* en las coordenadas del pixel de la esquina superior izquierda (*fila*, *columna*) usando lógica OR.

picVar debe ser un tipo de imagen.

Las coordenadas por omisión son (0, 0).

real() Menú MATH/Complex

real (<i>expresión</i>) \Rightarrow <i>expresión</i>	$\text{real}(2+3i)$ [ENTER]	2
Devuelve la parte real del argumento.	$\text{real}(z)$ [ENTER]	z
Nota: Todas las variables no definidas se tratan como variables reales. Consulte además imag() .	$\text{real}(x+iy)$ [ENTER]	x
<hr/>		
real (<i>lista</i>) \Rightarrow <i>lista</i>	$\text{real}(\{a+i\cdot b, 3, i\})$ [ENTER]	{a 3 0}
Devuelve la parte real de todos los elementos.		
<hr/>		
real (<i>matriz</i>) \Rightarrow <i>matriz</i>	$\text{real}([a+i\cdot b, 3:c, i])$ [ENTER]	$\begin{bmatrix} a & 3 \\ c & 0 \end{bmatrix}$

►Rect Menú MATH/Matrix/Vector ops

vector ► Rect	$[3, \angle\pi/4, \angle\pi/6]$ ► Rect [ENTER]	
Presenta <i>vector</i> en forma rectangular [x, y, z]. El vector puede ser de dimensión 2 o 3, y puede ser fila o columna.	$\begin{bmatrix} 3\cdot\sqrt{2} & 3\cdot\sqrt{2} & 3\cdot\sqrt{3} \\ 4 & 4 & 2 \end{bmatrix}$	
Nota: ► Rect es una instrucción del formato de visualización, no una función de conversión. Sólo puede utilizarla al final de una línea de entrada y no actualiza ans.	$[a, \angle b, \angle c]$ [ENTER]	$\begin{bmatrix} a\cdot\cos(b)\cdot\sin(c) \\ a\cdot\sin(b)\cdot\sin(c) \\ a\cdot\cos(c) \end{bmatrix}$
Nota: Consulte además ► Polar .		

Valor complejo ► Rect	En el modo Angle, en radianes:	
Presenta <i>Valor complejo</i> en la forma rectangular $a+bi$. El <i>Valor complejo</i> puede tener cualquier forma compleja. No obstante, una entrada e^{bi} causa un error en el modo Angle en radianes.	$4e^{(\pi/3)}$ ► Rect [ENTER]	$4\cdot e^{\frac{\pi}{3}}$
Nota: Para una entrada polar ($r\angle\theta$) debe utilizar paréntesis.	$(4\angle\pi/3)$ ► Rect [ENTER]	$2+2\cdot\sqrt{3}\cdot i$
	En el modo Angle, en grados centesimales:	
	$(1\angle 100)$ ► Rect [ENTER]	
	En el modo Angle en grados:	
	$(4\angle 60)$ ► Rect [ENTER]	$2+2\cdot\sqrt{3}\cdot i$

Nota: Para escribir ►**Rect** desde el teclado, pulse [2nd] [►] para el operador ►. Para escribir \angle , pulse [2nd] [\angle].

ref() Menú MATH/Matrix

ref (<i>matriz</i>) \Rightarrow <i>matriz</i>	$\text{ref}([-2, -2, 0, -6:1, -1.9, -9; -5, 2.4, -4])$ [ENTER]	
Devuelve la forma escalonada de <i>matriz</i> .	$\begin{bmatrix} 1 & -2/5 & -4/5 & 4/5 \\ 0 & 1 & 4/7 & 11/7 \\ 0 & 0 & 1 & -62/71 \end{bmatrix}$	
De forma opcional, cualquier elemento de matriz se trata como cero si su valor absoluto es menor que <i>tol</i> . Esta tolerancia se utiliza sólo si la matriz contiene entradas de coma flotante y no contiene ninguna variable simbólica sin valor asignado. De no ser así, <i>tol</i> se ignora.	$[a, b, c; e, f, g]$ ► m1 [ENTER]	$\begin{bmatrix} a & b & c \\ e & f & g \end{bmatrix}$
<ul style="list-style-type: none"> Si se utiliza \square [ENTER] o se ajusta el modo a Exact/Approx=APPROXIMATE, los cálculos se realizan mediante aritmética de coma flotante. Si <i>tol</i> se omite o no se utiliza, la tolerancia por omisión se calcula como: $5e-14 \cdot \max(\text{dim}(\text{matriz}))$ $\cdot \text{rowNorm}(\text{matriz})$. 	$\text{ref}(m1)$ [ENTER]	$\begin{bmatrix} 1 & \frac{f}{e} & \frac{g}{e} \\ 0 & 1 & \frac{a\cdot g - c\cdot e}{a\cdot f - b\cdot e} \end{bmatrix}$
Nota: Consulte además rref() .		

remain() Menú MATH/Number

remain(*expresión1*, *expresión2*) ⇒ *expresión*

remain(*lista1*, *lista2*) ⇒ *lista*

remain(*matriz1*, *matriz2*) ⇒ *matriz*

Devuelve el resto del primer argumento con respecto al segundo, según las siguientes identidades:

$$\begin{aligned} \text{remain}(x,0) &= x \\ \text{remain}(x,y) &= x - y * \text{iPart}(x/y) \end{aligned}$$

En consecuencia, tome en cuenta que **remain**(-x,y) = - **remain**(x,y). El resultado es cero o tiene el mismo signo que el primer argumento.

Nota: Consulte además **mod**().

```
remain(7.0) [ENTER] 7
remain(7.3) [ENTER] 1
remain(-7.3) [ENTER] -1
remain(7.-3) [ENTER] 1
remain(-7.-3) [ENTER] -1
remain({12.-14.16}.*{9.7.-5}) [ENTER]
                                     {3 0 1}
remain([9.-7:6.4].*[4.3:4.-3]) [ENTER]
                                     [ 1 -1
                                      2  1 ]
```

Rename CATALOG

Rename *NombreVar antiguo*, *NombreVar nuevo*

Cambia el nombre de la variable *NombreVar antiguo* por *NombreVar nuevo*.

```
{1.2.3.4} > L1 [ENTER] {1.2.3.4}
Rename L1. list1 [ENTER] Done
list1 [ENTER] {1.2.3.4}
```

Request CATALOG

Request *promptCadena*, *var*

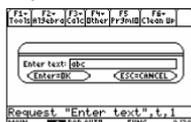
Si **Request** está dentro de un bloque **Dialog...EndDialog**, crea un cuadro de entrada para que el usuario escriba datos. Si es una instrucción única, crea un recuadro de diálogo para estos datos. En ambos casos, si *var* contiene una cadena, se muestra y resalta en el cuadro de entrada como la opción por omisión. *promptCadena* debe tener ≤ 20 caracteres.

Esta instrucción puede ser única o parte de un recuadro de diálogo.

El argumento opcional *alphaOn/Off* puede ser cualquier expresión. Si se le asigna un valor cero, *alpha-lock* se define en OFF. Si se le asigna un valor cualquiera distinto de cero, *alpha-lock* se define en ON. El valor predeterminado de bloqueo alfabético es ON cuando no se utiliza el argumento opcional.

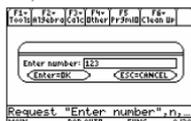
Si aparece más de una orden **Request** en un cuadro de diálogo **Dialog...EndDialog**, se utiliza el primer valor alfabético y se ignoran los restantes.

Request "Enter text".t.1 [ENTER]



El argumento con bloque alfabético activado del ejemplo anterior.

Request "Enter number".n.0



El argumento con bloque alfabético desactivado del ejemplo anterior.

Return CATALOG

Return [*expresión*]

Devuelve *expresión* como el resultado de la función. Se utiliza en un bloque **Func...EndFunc** o en un bloque **Prgm...EndPrgm**.

Nota: Use **Return** sin argumento para salir de un programa.

Nota: Introduzca el texto en una única línea en la pantalla Home.

```
Define factorial(nn)=Func
:local answer,count:1→answer
:For count,1,nn
:answer*count→answer:EndFor
:Return answer:EndFunc [ENTER] Done
factorial(3) [ENTER] 6
```

right() Menú MATH/List

right (<i>lista1</i> , <i>núm1</i>) ⇒ <i>lista</i>	<code>right({1.3, -2.4}, 3) [ENTER]</code>	{3 -2 4}
Devuelve los <i>núm</i> elementos situados a la derecha de la <i>lista1</i> . Si se omite <i>núm</i> , devuelve toda la <i>lista1</i> .		
right (<i>Cadena origen</i> [, <i>núm1</i>]) ⇒ <i>cadena</i>	<code>right("Hello", 2) [ENTER]</code>	"lo"
Devuelve los <i>núm</i> caracteres situados a la derecha de la cadena de caracteres <i>Cadena origen</i> . Si se omite <i>núm</i> , devuelve la <i>Cadena origen</i> en su totalidad.		
right (<i>comparación</i>) ⇒ <i>expresión</i>	<code>right(x<3) [ENTER]</code>	3
Devuelve el lado derecho de una ecuación o desigualdad.		

root() Menú CATALOG/MATH/Number

root (<i>expresión</i>) ⇒ <i>raíz</i>	<code>root(8, 3) [ENTER]</code>	2
Calcula una raíz <i>n</i> ésima de <i>x</i> donde <i>x</i> puede ser una constante de coma flotante real o compleja, una constante racional entera o compleja o una expresión simbólica general.	<code>root(3, 3) [ENTER]</code>	3 ^{1/3}
	<code>root(3, 0.3) [ENTER]</code>	1.442249570

rotate() Menú MATH/Base

rotate (<i>entero1</i> , <i>#Rotaciones</i>) ⇒ <i>entero</i>	En el modo de base Bin:	
Traslada los bits en un entero binario. Puede introducir el <i>entero1</i> en cualquier base de numeración; se convierte automáticamente a una forma binaria de 32 bits con signo. Si la magnitud de <i>entero1</i> es demasiado grande para esta forma, una operación de módulos simétricos la lleva dentro del rango.	<code>rotate(0b1111010110000110101) [ENTER]</code> <code>0b1000000000000111101011000011010</code>	
Si <i>#Rotaciones</i> es positivo, la traslación es hacia la izquierda. Si <i>#Rotaciones</i> es negativo, la traslación es hacia la derecha. El valor predeterminado es -1 (se traslada un bit a la derecha).	<code>rotate(256, 1) [ENTER]</code>	0b1000000000
Por ejemplo, en una traslación hacia la derecha:	En el modo de base Hex:	
→ Cada bit se traslada hacia la derecha.	<code>rotate(0h78E) [ENTER]</code>	0h3C7
0b00000000000001111010110000110101	<code>rotate(0h78E, -2) [ENTER]</code>	0h800001E3
↑ El bit más a la derecha se traslada al extremo izquierdo.	<code>rotate(0h78E, 2) [ENTER]</code>	0h1E38
se genera:	Importante: Para introducir un número binario o hexadecimal, utilice siempre el prefijo 0b ó 0h (cero, no la letra O).	
0b100000000000000111101011000011010		
El resultado se presenta de acuerdo con el estado del modo Base.		

rotate (<i>lista1</i> , <i>#Rotaciones</i>) ⇒ <i>lista</i>	En el modo de base Dec:	
Devuelve una copia de la <i>lista1</i> trasladada a izquierdas o derechas según los elementos de <i>#Rotaciones</i> . No modifica la <i>lista1</i> .	<code>rotate({1, 2, 3, 4}) [ENTER]</code>	{4 1 2 3}
Si el <i>#Rotaciones</i> es positivo, la traslación es a la izquierda. Si el <i>#Rotaciones</i> es negativo, la traslación es a la derecha. El valor predeterminado es -1 (traslada un elemento a la derecha).	<code>rotate({1, 2, 3, 4}, -2) [ENTER]</code>	{3 4 1 2}
	<code>rotate({1, 2, 3, 4}, 1) [ENTER]</code>	{2 3 4 1}

rotate (cadena1, #Rotaciones) ⇒ cadena	rotate("abcd") ENTER	"dabc"
Devuelve una copia de la <i>cadena1</i> trasladada a la derecha o a la izquierda según los caracteres del #Rotaciones. No modifica la <i>cadena1</i> .	rotate("abcd".-2) ENTER	"cdab"
	rotate("abcd".1) ENTER	"bcda"
Si el #Rotaciones es positivo, la traslación es a la izquierda. Si el #Rotaciones es negativo, la traslación es a la derecha. El valor predeterminado es -1 (traslada un carácter a la derecha).		

round() Menú MATH/Number

round (expresión1, dígitos) ⇒ expresión	round(1.234567,3) ENTER	1.235
Devuelve el argumento redondeado al número de dígitos decimales indicados por <i>dígitos</i> .		
El valor de <i>dígitos</i> debe ser un entero en el rango 0–12. Si no se incluye <i>dígitos</i> , devuelve el argumento redondeado a 12 dígitos significativos.		
Nota: El modo Display Digits puede influir en la presentación de este resultado.		

round (lista1, dígitos) ⇒ lista	round({π,√(2),ln(2)},4) ENTER	{3.1416 1.4142 .6931}
Devuelve la lista de los elementos redondeados de acuerdo con el número indicado de dígitos.		

round (matriz1, dígitos) ⇒ matriz	round([ln(5),ln(3);π,e^(1)],1) ENTER	$\begin{bmatrix} 1.6 & 1.1 \\ 3.1 & 2.7 \end{bmatrix}$
Devuelve la matriz de los elementos redondeados de acuerdo con el número indicado de dígitos.		

rowAdd() Menú MATH/Matrix/Row ops

rowAdd (matriz1, rÍndice1, rÍndice2) ⇒ matriz	rowAdd([3,4;-3,-2],1,2) ENTER	$\begin{bmatrix} 3 & 4 \\ 0 & 2 \end{bmatrix}$
Devuelve una copia de la <i>matriz1</i> con la <i>rÍndice2</i> sustituida por la suma de las filas <i>rÍndice1</i> y <i>rÍndice2</i> .	rowAdd([a,b;c,d],1,2) ENTER	$\begin{bmatrix} a & b \\ a+c & b+d \end{bmatrix}$

rowDim() Menú MATH/Matrix/Dimensions

rowDim (matriz) ⇒ expresión		$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$
Devuelve el número de filas de <i>matriz</i> .	[1,2;3,4;5,6]→M1 ENTER	
Nota: Consulte además colDim() .	rowdim(M1) ENTER	3

rowNorm() Menú MATH/Matrix/Norms

rowNorm (matriz) ⇒ expresión	rowNorm([-5,6,-7;3,4,9;9,-9,-7]) ENTER	25
Devuelve el valor máximo obtenido al sumar los valores absolutos de los elementos de filas de la <i>matriz</i> .		
Nota: Todos los elementos de matriz se deben simplificar a números. Consulte además colNorm() .		

rowSwap() Menú MATH/Matrix/Row ops

rowSwap(*matriz1*, *rñdice1*, *rñdice2*) ⇒ *matriz*

[1.2:3.4:5.6]→Mat [ENTER]

Devuelve la *matriz1* con las filas *rñdice1* y *rñdice2* intercambiadas.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

rowSwap(Mat.1.3) [ENTER]

$$\begin{bmatrix} 5 & 6 \\ 3 & 4 \\ 1 & 2 \end{bmatrix}$$

RplcPic CATALOG

RplcPic *picVar*[, *fila*][, *columna*]

Vacía la pantalla Graph y coloca la imagen *picVar* en las coordenadas del píxel (*fila*, *columna*). Si no desea vaciar la pantalla, utilice **RclPic**.

picVar debe ser una variable de tipo de imagen. La *fila* y la *columna*, si se incluyen, especifican las coordenadas del píxel situado en la esquina superior izquierda de la imagen. Las coordenadas predeterminadas son (0, 0).

Nota: En el caso de imágenes que ocupan menos de una pantalla, sólo se vacía el área que ocupa la nueva imagen.

rref() Menú MATH/Matrix

rref(*matriz1*, *tol*) ⇒ *matriz*

Devuelve la forma reducida escalonada de *matriz1*.

De forma opcional, cualquier elemento de matriz se considera como cero si su valor absoluto es menor que *tol*. Esta tolerancia se utiliza sólo si la matriz tiene entradas de coma flotante y no contiene ninguna variable simbólica sin valor asignado. De no ser así, *tol* se ignora.

- Si se utiliza \square [ENTER] o se ajusta el modo a Exact/Approx=APPROXIMATE, los cálculos se realizan mediante aritmética de coma flotante.
- Si *tol* se omite o no se utiliza, la tolerancia por omisión se calcula como:

$$5 \times 10^{-14} * \max(\dim(\text{matriz1})) * \text{rowNorm}(\text{matriz1})$$

Nota: Consulte también **ref()**.

rref([-2.-2.0.-6:1.-1.9.-9;
-5.2.4.-4]) [ENTER]

$$\begin{bmatrix} 1 & 0 & 0 & 66/71 \\ 0 & 1 & 0 & 147/71 \\ 0 & 0 & 1 & -62/71 \end{bmatrix}$$

rref([a.b.x:c.d.y]) [ENTER]

$$\begin{bmatrix} 1 & 0 & \frac{d \cdot x - b \cdot y}{a \cdot d - b \cdot c} \\ 0 & 1 & \frac{-(c \cdot x - a \cdot y)}{a \cdot d - b \cdot c} \end{bmatrix}$$

sec() Menú MATH/Trig (MATEMÁTICAS/Trigonometría)

sec(*expresión1*) ⇒ *expresión*

sec(*lista1*) ⇒ *lista*

Devuelve la secante de *expresión1* o una lista de las secantes de todos los elementos de *lista1*.

Nota: El resultado devuelve un ángulo expresado en grados, grados centesimales o radianes, según el valor de modo de ángulo actual.

En el modo Angle, en grados:

sec(45) [ENTER]

$$\sqrt{2}$$

sec({1.2.3.4}) [ENTER]

$$\frac{1}{\cos(1)} \quad 1.000\dots \quad \frac{1}{\cos(4)}$$

sec⁻¹() Menú MATH/Trig (MATEMÁTICAS/Trigonometría)

sec⁻¹(*expresión1*) ⇒ *expresión*

sec⁻¹(*lista1*) ⇒ *lista*

Devuelve el ángulo cuya secante es *expresión1* o una lista de las secantes inversas de todos los elementos de *lista1*.

Nota: El resultado devuelve un ángulo expresado en grados, grados centesimales o radianes, según el valor de modo de ángulo actual.

En el modo Angle, en grados:

$$\text{sec}^{-1}(1) \text{ [ENTER]} \quad 0$$

En el modo Angle, en grados centesimales:

$$\text{sec}^{-1}(\sqrt{2}) \text{ [ENTER]}$$

En el modo Angle, en radianes:

$$\text{sec}^{-1}(\{1, 2.5\}) \text{ [ENTER]} \quad 0 \frac{\pi}{3} \cos^{-1}(1/5)$$

sech() Menú MATH/Hyperbolic (MATEMÁTICAS/Hiperbólicas)

sech(*expresión1*) ⇒ *expresión*

sech(*lista1*) ⇒ *lista*

Devuelve la secante hiperbólica de *expresión1* o una lista de las secantes hiperbólicas de todos los elementos de *lista1*.

$$\text{sech}(3) \text{ [ENTER]} \quad \frac{1}{\cosh(3)}$$

$$\text{sech}(\{1, 2, 3, 4\}) \text{ [ENTER]} \quad \frac{1}{\cosh(1)} \cdot 198\ldots \frac{1}{\cosh(4)}$$

sech⁻¹() Menú MATH/Hyperbolic (MATEMÁTICAS/Hiperbólicas)

sech⁻¹(*expresión1*) ⇒ *expresión*

sech⁻¹(*lista1*) ⇒ *lista*

Devuelve la secante hiperbólica inversa de *expresión1* o una lista de las secantes hiperbólicas inversas de todos los elementos de *lista1*.

En el modo Angle en radianes y el modo complejo rectangular:

$$\text{sech}^{-1}(1) \text{ [ENTER]} \quad 0$$

$$\text{sech}^{-1}(\{1, -2, 2, 1\}) \text{ [ENTER]} \quad 0 \left(\frac{2 \cdot \pi}{3}\right) \cdot i \ 1.074\ldots \cdot i$$

Send CATALOG

Send *lista*

Instrucción del CBL™ (Calculator-Based Laboratory™) o CBR™ (Calculator-Based Ranger™). Envía la *lista* al puerto de conexión.

Parte de un programa:

```
⋮  
:Send {1.0}  
:Send {1.2,1}  
⋮
```

SendCalc CATALOG

SendCalc *var*

Envía la variable *var* a la puerta de enlace, donde otra unidad enlazada a esa puerta puede recibir su valor. La unidad receptora ha de encontrarse en la pantalla inicial o debe ejecutar **GetCalc** desde un programa.

Si envía desde una TI-89, TI-92 Plus o Voyage™ 200 a una TI-92, se produce un error si la TI-92 ejecuta **GetCalc** desde un programa. En este caso, la unidad de envío ha de usar **SendChat** en su lugar.

Parte de un programa:

```
⋮  
:a+b→x  
:SendCalc x  
⋮
```

SendCalc *var[,port]*

Envía el contenido de la variable *var* de una TI-89 Titanium a otra TI-89 Titanium.

Si no se ha especificado un puerto, o si el valor especificado es *port = 0*, la TI-89 Titanium envía los datos utilizando el puerto USB, si estuviera conectado; en caso contrario, utiliza el puerto E/S.

Si *port = 1*, la TI-89 Titanium envía los datos sólo a través del puerto USB.

Si *port = 2*, la TI-89 Titanium envía los datos sólo a través del puerto E/S.

SendChat CATALOG

SendChat *var*

Alternativa general a **SendCalc**, resulta útil si la unidad receptora es una TI-92 (o un programa de "charla" genérico que permita usar una TI-92, TI-92 Plus o Voyage 200). Véase **SendCalc** para más información.

SendChat envía una variable sólo si dicha variable es compatible con la TI-92, lo que suele ser cierto en programas de "charla". Sin embargo, **SendChat** no envía una variable archivada, una base de datos de gráfica TI-89, etc.

Parte de un programa:

```
:  
:  
:a+b→x  
:SendChat x  
:  
:
```

seq() Menú MATH/List

seq(*expresión, var, inferior, superior[, paso]*) ⇒ *lista*

Incrementa *var* de *inferior* hasta *superior* según el *paso*, calcula la *expresión*, y devuelve los resultados como una lista. El contenido primitivo de *var* no varía después de completarse **seq()**.

La *var* no puede ser una variable del sistema.

Valor por omisión de *paso* = 1.

seq($n^2, n, 1, 6$) {1 4 9 16 25 36}

seq($1/n, n, 1, 10, 2$)
{1 1/3 1/5 1/7 1/9}

sum(seq($1/n^2, n, 1, 10, 1$))
196...
127...

o pulse para obtener: 1.549..

setDate() CATALOG

setDate(*año, mes, día*) ⇒ *lista anterior*

Ajusta el reloj en la fecha indicada en el argumento y devuelve una lista. (**Nota:** El *año* debe hallarse en el rango 1997 - 2132.) La lista devuelta tiene el formato {*año anterior, mes anterior, día anterior*}. La fecha devuelta corresponde al valor anterior del reloj.

Introduzca el año como un número entero de cuatro cifras. El mes y el día pueden ser enteros de una o dos cifras.

setDate(2001,10,31)
{2001 11 1}

setDtFmt() CATALOG

setDtFmt(entero) ⇒ enteroanterior

Define el formato de fecha del escritorio de acuerdo con el argumento y devuelve el valor del formato de fecha anterior.

Valores enteros:

- 1 = MM/DD/AA
- 2 = DD/MM/AA
- 3 = MM.DD.AA
- 4 = DD.MM.AA
- 5 = AA.MM.DD
- 6 = MM-DD-AA
- 7 = DD-MM-AA
- 8 = AA-MM-DD

setFold() CATALOG

setFold(Nombre de carpeta nueva) ⇒ Cadena de carpeta antigua

Devuelve el nombre de la carpeta actual en una cadena y establece *Nombre de carpeta nueva* como la carpeta actual.

Es necesario que ya exista *Nombre de carpeta nueva*.

```
newFold chris [ENTER] Done
setFold(main) [ENTER] "chris"
setFold(chris)→oldfoldr [ENTER] "main"
l→a [ENTER] 1
setFold(#oldfoldr) [ENTER] "chris"
a [ENTER] a
chris\a [ENTER] 1
```

setGraph() CATALOG

setGraph(modoNombreCadena, estadoCadena) ⇒ cadena

Establece el modo Graph de *modoNombreCadena* en *estadoCadena*, y devuelve el estado previo del modo. El almacenamiento de los estados previos permite su recuperación posterior.

modoNombreCadena es una cadena de caracteres que especifica el modo que desea establecer. Debe ser uno de los modos de la siguiente tabla.

estadoCadena es una cadena de caracteres que especifica el nuevo estado del modo. Debe ser uno de los estados indicados abajo para el modo concreto en proceso de ajuste.

```
setGraph("Graph Order","Seq") [ENTER] "SEQ"
setGraph("Coordinates","Off") [ENTER] "RECT"
```

Nota: Al introducir nombres de modos, las mayúsculas y los espacios en blanco son opcionales.

Nombres de modos	Estados
"Coordinates"	"Rect", "Polar", "Off"
"Graph Order"	"Seq", "Simul" ¹
"Grid"	"Off", "On" ²
"Axes"	"Off", "On" (no en el modo de gráficas en 3D) "Box", "Axes", "Off" (modo de gráficas en 3D)
"Leading Cursor"	"Off", "On" ²
"Labels"	"Off", "On"
"Style"	"Wire Frame", "Hidden Surface", "Contour Levels", "Wire and Contour", "Implicit Plot" ³
"Seq Axes"	"Time", "Web", "U1-vs-U2" ⁴
"DE Axes"	"Time", "t-vs-y" ¹ , "y-vs-y" ¹ , "y1-vs-y2" ¹ , "y1-vs-y2" ¹ , "y1'-vs-y2'" ⁵
	Sugerencia: Para escribir un símbolo de número primo ('), pulse [2nd] ['].
"Solution Method"	"RK", "Euler" ⁵
"Fields"	"SlpFld", "DirFld", "FldOff" ⁵

"Discontinuity Detection" "Off", "On"⁶

¹No disponible en el modo de gráficas de sucesiones, de gráficas en 3D o de gráficas de ecuaciones diferenciales. No disponible en el modo de gráficas de función cuando "Discontinuity Detection" está definido en "On".

²No disponible en el modo de gráficas en 3D.

³Aplicable únicamente al modo de gráficas en 3D.

⁴Aplicable únicamente al modo de gráficas de sucesiones.

⁵Aplicable únicamente al modo de gráficas de ecuaciones diferenciales.

⁶Aplicable únicamente al modo de gráficas de función cuando "Graph Order" está definido en "Seq".

setMode() CATALOG

setMode(*modoNombreCadena*, *estadoCadena*) ⇒
cadena

setMode(*lista*) ⇒ *cadenaLista*

Establece el modo de *modoNombreCadena* en *estadoCadena*, y devuelve el estado actual de este modo.

modoNombreCadena es una cadena de caracteres que especifica el modo que desea configurar. Debe ser uno de los nombres de modo de la siguiente tabla.

estadoCadena es una cadena de caracteres que especifica el nuevo estado del modo. Debe ser uno de los estados indicados abajo para el modo concreto que se esté ajustando.

La *lista* contiene pares de cadenas de palabras clave y los ajusta todos a la vez. Se recomienda utilizarla en los cambios simultáneos de varios modos. El ejemplo no dará el resultado que se indica si cada uno de los pares se introduce a través de una orden **setMode()** independiente en el orden mostrado.

Utilice **setMode(*var*)** para restablecer los estados guardados con **getMode("ALL")> var**.

Nota: Para definir o devolver información sobre el modo Unit System, utilice **setUnits()** o **getUnits()** en vez de **setMode()** o **getMode()**.

```
setMode("Angle", "Degree")
[ENTER] RADIAN

sin(45) [ENTER]  $\frac{\sqrt{2}}{2}$ 

setMode("Angle", "Radian")
[ENTER] "DEGREE"

sin( $\pi/4$ ) [ENTER]  $\frac{\sqrt{2}}{2}$ 

setMode("Angle", "Gradian")
[ENTER] "RADIAN"

sin(50) [ENTER]  $\frac{\sqrt{2}}{2}$ 

setMode("Display Digits",
"Fix 2") [ENTER] "FLOAT"
 $\pi$  [ENTER] 3.14
setMode("Display Digits",
"Float") [ENTER] "FIX 2"
 $\pi$  [ENTER] 3.141...

setMode({"Split Screen",
"Left-Right", "Split 1 App",
"Graph", "Split 2 App", "Table"})
[ENTER] {"Split 2 App" "Graph"
"Split 1 App" "Home"
"Split Screen" "FULL"}
```

Nota: Las mayúsculas y los espacios en blanco son opcionales cuando se introducen nombres de modos. Además, los resultados de estos ejemplos pueden ser distintos en su unidad.

Nombres de modos Estados

"Graph"	"Function", "Parametric", "Polar", "Sequence", "3D", "Diff Equations"
"Display Digits"	"Fix 0", "Fix 1", ..., "Fix 12", "Float", "Float 1", ..., "Float 12"
"Angle"	"Radian", "Degree", "Gradian"
"Exponential Format"	"Normal", "Scientific", "Engineering"
"Complex Format"	"Real", "Rectangular", "Polar"

"Vector Format"	"Rectangular", "Cylindrical", "Spherical"
"Pretty Print"	"Off", "On"
"Split Screen"	"Full", "Top-Bottom", "Left-Right"
"Split 1 App"	"Home", "Y= Editor", "Window Editor", "Graph", "Table", "Data/Matrix Editor", "Program Editor", "Text Editor", "Numeric Solver", "Apl flash"
"Split 2 App"	"Home", "Y= Editor", "Window Editor", "Graph", "Table", "Data/Matrix Editor", "Program Editor", "Text Editor", "Numeric Solver", "Apl flash"
"Number of Graphs"	"1", "2"
"Graph2"	"Function", "Parametric", "Polar", "Sequence", "3D", "Diff Equations"
"Exact/Approx"	"Auto", "Exact", "Approximate"
"Base"	"Dec", "Hex", "Bin"
"Language"	"English", "Idioma alternativo"

setTable() CATALOG

setTable(*modoNombreCadena, estadoCadena*) ⇒ *cadena*

Establece el parámetro de la tabla *modoNombreCadena* en *estadoCadena*, y devuelve el estado previo de este parámetro. Al almacenar los estados previos, puede recuperarlos más adelante.

modoNombreCadena es una cadena de caracteres que especifica el parámetro que desea ajustar. Debe ser uno de los parámetros de la siguiente tabla.

estadoCadena es una cadena de caracteres que especifica el nuevo estado del parámetro. Debe ser uno de los estados indicados del parámetro que esté ajustando.

setTable("Graph <-> Table", "ON")

"OFF"

setTable("Independent", "AUTO")

"ASK"



Nota: La capitalización y los espacios en blanco son opcionales al introducir parámetros.

Nombres de parámetros	Estados
-----------------------	---------

"Graph <-> Table"	"Off", "On"
-------------------	-------------

"Independent"	"Auto", "Ask"
---------------	---------------

setTime() CATALOG

setTime(*hora,minuto,segundo*) ⇒ *listaanterior*

Ajusta el reloj en la hora indicada en el argumento y devuelve una lista. La lista tiene el formato {*horaanterior,minutoanterior,segundoanterior*}. La hora devuelta corresponde al valor anterior del reloj.

Escriba la hora en formato de 24 horas, en donde 13 = 1 P.M.

setTime(11,32,50)

{10 44 49}

setTmFmt() CATALOG

setTmFmt(*entero*) ⇒ *enteroanterior*

Define el formato de hora del escritorio de acuerdo con el argumento y devuelve el valor del formato de hora anterior.

Valores enteros:

12 = reloj de 12 horas

24 = reloj de 24 horas

setTmZn() CATALOG

setTmZn(entero) ⇒ enteroanterior

Define el formato de hora de acuerdo con el argumento y devuelve el valor de la zona horaria anterior.

La zona horaria se define mediante un entero que indica los minutos de diferencia respecto a la hora media de Greenwich (GMT), establecida en Greenwich, Inglaterra. Por ejemplo, si la zona horaria se diferencia dos horas de la GMT, el dispositivo devuelve 120 (minutos).

Los enteros correspondientes a las zonas situadas al oeste de la GMT son negativos.

Los enteros correspondientes a las zonas situadas al este de la GMT son positivos.

Si la hora media de Greenwich es 14:07:07, son las:

07:07:07 a.m. en Denver, Colorado (hora estándar de las Montañas Rocosas)
(-420 minutos respecto a GMT)

15:07:07 p.m. en Bruselas, Bélgica (hora estándar de Europa central)
(+60 minutos respecto a GMT)

setUnits() CATALOG

setUnits(lista) ⇒ lista

Ajusta las unidades por omisión en los valores especificados en la *lista1*, y devuelve una lista de los valores por omisión previos.

- Para especificar el sistema incorporado SI (métrico) o ENG/US, la *lista1* utiliza la forma: {"SI"} o {"ENG/US"}
- Para especificar un conjunto personalizado de unidades por omisión, la *lista1* utiliza la forma: {"CUSTOM", "cat1", "unidad1", "cat2", "unidad2", ...} donde cada par de *cat* y *unidad* especifica una categoría y su unidad por omisión (se pueden especificar sólo unidades incorporadas, no unidades definidas por el usuario). Cualquier categoría no especificada utilizará su unidad personalizada anterior.
- Para volver a las unidades por omisión personalizadas, la *lista1* utiliza la forma: {"CUSTOM"}

Si desea distintos valores predeterminados en función de la situación, cree listas independientes y guárdelas con nombres diferentes. Para utilizar un conjunto de valores por omisión, especifique ese nombre de lista en **setUnits()**.

Es posible utilizar **setUnits()** para restablecer los ajustes previamente guardados con **setUnits()** ⇒ *var* o con **getUnits()** ⇒ *var*.

Todos los nombres de unidad deben comenzar con un guión bajo _.

  [_]
  [2nd] [_]

También es posible seleccionar unidades en un menú pulsando:

  [2nd] [UNITS]
  [↓] [UNITS]

```
setUnits({"SI"}) [ENTER]
{"SI" "Area" "NONE"
"Capacitance" "_F" ...}
```

```
setUnits({"CUSTOM"."Length",
"_cm"."Mass","_gm"}) [ENTER]
{"SI" "Length" "_m"
"Mass" "_kg" ...}
```

Nota: Su pantalla puede mostrar unidades diferentes.

Shade CATALOG

Shade $expr1$, $expr2$, [$xinferior$], [$xsuperior$], [$modelo$], [$patRes$]

Presenta la pantalla Graph, dibuja $expr1$ y $expr2$, y sombrea las áreas en que $expr1$ es menor que $expr2$. ($expr1$ y $expr2$ deben ser expresiones que utilizan x como variable independiente).

Los valores de $xinferior$ y $xsuperior$, si se incluyen, especifican los límites izquierdo y derecho del sombreado. Los valores válidos están comprendidos entre $xmin$ y $xmax$. Por omisión, son $xmin$ y $xmax$.

El $modelo$ especifica uno de los cuatro tipos de sombreado:

- 1 = vertical (por omisión)
- 2 = horizontal
- 3 = pendiente negativa a 45°
- 4 = pendiente positiva a 45°

El valor de $patRes$ especifica la resolución de los tipos de sombreado:

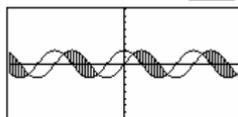
- 1 = sombreado continuo
- 2 = espaciado de 1 pixel (por omisión)
- 3 = espaciado de 2 pixels
- ⋮
- 10 = espaciado de 9 pixels

Nota: El sombreado interactivo está disponible en la pantalla Graph mediante la instrucción **Shade**. El sombreado automático de una función está disponible en la instrucción **Style**. **Shade** no es válida en el modo de gráficas en 3D.

En la ventana de visualización

ZoomTrig:

Shade $\cos(x) \cdot \sin(x)$ [ENTER]



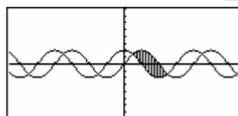
[HOME]

[CALC HOME]

ClrDraw [ENTER]

Shade $\cos(x) \cdot \sin(x) \cdot 0.5$ [ENTER]

Done



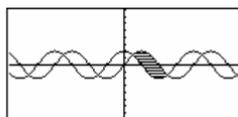
[HOME]

[CALC HOME]

ClrDraw [ENTER]

Shade $\cos(x) \cdot \sin(x) \cdot 0.5 \cdot 2$ [ENTER]

Done



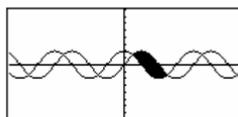
[HOME]

[CALC HOME]

ClrDraw [ENTER]

Shade $\cos(x) \cdot \sin(x) \cdot 0.5 \cdot 2 \cdot 1$ [ENTER]

Done



shift(cadena1[,#de desplazamientos]) \Rightarrow cadena

Devuelve una copia de la *cadena1* desplazada a la derecha o a la izquierda tantos caracteres como indica el *#de desplazamientos*. No altera la *cadena1*.

Si el *#de desplazamientos* es positivo, el desplazamiento es a la izquierda. Si el *#de desplazamientos* es negativo, el desplazamiento es a la derecha. El valor por omisión es -1 (desplazamiento a la derecha de un carácter).

Los caracteres introducidos al principio o al final de la *cadena* mediante el desplazamiento aparecen como un espacio.

shift("abcd") **ENTER** " abc"

shift("abcd", -2) **ENTER** " ab"

shift("abcd", 1) **ENTER** "bcd "

ShowStat CATALOG

ShowStat

Muestra un recuadro de diálogo que contiene los últimos resultados estadísticos calculados, si aún son válidos. Los resultados estadísticos se borran automáticamente si se modifican los datos con los que se calculan.

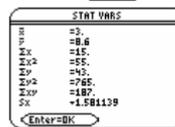
Utilice esta instrucción después de un cálculo estadístico, como por ejemplo, **LinReg**.

{1,2,3,4,5} \rightarrow L1 **ENTER** {1 2 3 4 5}

{0,2,6,10,25} \rightarrow L2 **ENTER** {0 2 6 10 25}

TwoVar L1,L2 **ENTER**

ShowStat **ENTER**



sign() Menú MATH/Number

sign(expresión1) \Rightarrow expresión

sign(lista1) \Rightarrow lista

sign(matriz1) \Rightarrow matriz

En el caso de una *expresión1* real o compleja, devuelve *expresión1* **abs**(*expresión1*) cuando *expresión1* \neq 0.

Devuelve 1 si la *expresión1* es positiva. Devuelve -1 si la *expresión1* es negativa.

sign(0) devuelve ± 1 si el modo de formato complejo es REAL; de no ser así, devuelve **sign**(0).

sign(0) representa la circunferencia de radio unidad en el dominio complejo.

En el caso de una lista o una matriz, devuelve los signos de todos los elementos.

sign(-3.2) **ENTER** -1.

sign({2,3,4,-5}) **ENTER** {1 1 1 -1}

sign(1+abs(x)) **ENTER** 1

Si el modo de formato complejo es REAL:

sign([-3,0.3]) **ENTER** [-1 \pm 1 1]

simult() Menú MATH/Matrix

simult(coefMatriz, constVector, tol) \Rightarrow matriz

Devuelve un vector columna que contiene las soluciones de un sistema de ecuaciones lineales.

coefMatriz debe ser una matriz cuadrada compuesta por los coeficientes de las ecuaciones.

El constVector debe tener idéntico número de filas (mismo tamaño) que la coefMatriz y debe contener las constantes.

De forma opcional, cualquier elemento de matriz se toma como cero si su valor absoluto es menor que tol. Esta tolerancia sólo se utiliza si la matriz tiene entradas de coma flotante y no contiene ninguna variable simbólica sin valor asignado. De no ser así, tol se ignora.

- Si se utiliza \square [ENTER] o se ajusta el modo a Exact/Approx=APPROXIMATE, los cálculos se realizan mediante aritmética de coma flotante.

- Si tol se omite o no se utiliza, la tolerancia por omisión se calcula como:

$$5E-14 * \max(\dim(\text{coefMatriz})) * \text{rowNorm}(\text{coefMatriz})$$

Hallar x e y:
$$\begin{aligned} x + 2y &= 1 \\ 3x + 4y &= -1 \end{aligned}$$

simult([1,2;3,4],[1;-1]) [ENTER]

$$\begin{bmatrix} -3 \\ 2 \end{bmatrix}$$

La solución es $x = -3$ e $y = 2$.

Hallar:
$$\begin{aligned} ax + by &= 1 \\ cx + dy &= 2 \end{aligned}$$

[a,b;c,d] \rightarrow matx1 [ENTER]

simult(matx1,[1;2]) [ENTER]

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

$$\begin{bmatrix} -(2 \cdot b - d) \\ a \cdot d - b \cdot c \\ 2 \cdot a - c \\ a \cdot d - b \cdot c \end{bmatrix}$$

simult(coefMatriz, constMatriz, tol) \Rightarrow matriz

Resuelve varios sistemas de ecuaciones lineales, teniendo cada sistema los mismos coeficientes o términos independientes pero distintas constantes.

Cada columna en la constMatriz debe contener las constantes para un sistema de ecuaciones. Cada columna en la matriz resultante contiene la solución para el sistema correspondiente.

Hallar:
$$\begin{aligned} x + 2y &= 1 & x + 2y &= 2 \\ 3x + 4y &= -1 & 3x + 4y &= -3 \end{aligned}$$

simult([1,2;3,4],[1,2;-1,-3]) [ENTER]

$$\begin{bmatrix} -3 & -7 \\ 2 & 9/2 \end{bmatrix}$$

Para el primer sistema, $x = -3$ e $y = 2$. Para el segundo sistema, $x = -7$ e $y = 9/2$.

sin() Tecla [2nd] [SIN] Tecla [SIN]

sin(expresión) \Rightarrow expresión

sin(lista) \Rightarrow lista

sin(expresión) devuelve el seno del argumento.

sin(lista) devuelve una lista de senos de todos los elementos de la lista.

Nota: El argumento se interpreta como un ángulo expresado en grados, grados centesimales o radianes, según el modo de ángulo actual. Puede utilizar $^{\circ}$, $^{\prime}$ o $^{\prime\prime}$ para anular temporalmente el valor de modo de ángulo actual.

En el modo Angle, en grados:

sin($(\pi/4)^{\circ}$) [ENTER]

$$\frac{\sqrt{2}}{2}$$

sin(45) [ENTER]

$$\frac{\sqrt{2}}{2}$$

sin({0.60,90}) [ENTER]

$$\{0 \frac{\sqrt{3}}{2} 1\}$$

En el modo Angle, en grados centesimales:

sin(50) [ENTER]

$$\frac{\sqrt{2}}{2}$$

En el modo Angle, en radianes:

sin($\pi/4$) [ENTER]

$$\frac{\sqrt{2}}{2}$$

sin(45°) [ENTER]

$$\frac{\sqrt{2}}{2}$$

sin(Matriz cuadrada) \Rightarrow *Matriz cuadrada*

Devuelve la matriz seno de *Matriz cuadrada1*. Esto *no* es lo mismo que calcular el seno de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.

La *Matriz cuadrada1* debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

En el modo Angle, en radianes:

`sin([1.5,3;4.2,1:6.-2.1])` **[ENTER]**

.942...	-.045...	-.031...
-.045...	.949...	-.020...
-.048...	-.005...	.961...

sin⁻¹()

Tecla **[SIN⁻¹]**

Tecla **[2nd]** **[SIN⁻¹]**

sin⁻¹(expresión) \Rightarrow *expresión*

sin⁻¹(lista) \Rightarrow *lista*

sin⁻¹(expresión) devuelve el ángulo cuyo seno es *expresión1*.

sin⁻¹(lista) devuelve una lista de los senos inversos de cada elemento de la *lista1*.

Nota: El resultado devuelve un ángulo expresado en grados, grados centesimales o radianes, según el valor de modo de ángulo actual.

En el modo Angle, en grados:

`sin-1(1)` **[ENTER]**

90

En el modo Angle, en grados centesimales:

`sin-1(1)` **[ENTER]**

En el modo Angle, en radianes:

`sin-1({0..2..5})` **[ENTER]**

{0 .201... .523...}

sin⁻¹(Matriz cuadrada) \Rightarrow *Matriz cuadrada*

Devuelve la matriz arcoseno de *Matriz cuadrada1*. Esto *no* es lo mismo que calcular el arcoseno de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.

La *Matriz cuadrada1* debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

En el modo Angle en radianes y en el modo de formato complejo rectangular:

`sin-1([1.5,3:4.2,1:6.-2.1])` **[ENTER]**

- .164...- .064...•i	1.490...-2.105...•i	...
.725...-1.515...•i	.947...- .778...•i	...
2.083...-2.632...•i	-1.790...+1.271...•i	...

sinh()

Menú MATH/Hyperbolic

sinh(expresión) \Rightarrow *expresión*

sinh(lista) \Rightarrow *lista*

sinh(expresión) devuelve el seno hiperbólico del argumento.

sinh(lista) devuelve una lista de los senos hiperbólicos de los elementos de la *lista1*.

`sinh(1.2)` **[ENTER]**

1.509...

`sinh({0,1,2,3.})` **[ENTER]**

{0 1.509... 10.017...}

sinh(Matriz cuadrada) \Rightarrow *Matriz cuadrada*

Devuelve la matriz seno hiperbólico de la *Matriz cuadrada1*. Esto *no* es lo mismo que calcular el seno hiperbólico de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.

La *Matriz cuadrada1* debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

En el modo Angle en radianes:

`sinh([1.5,3:4.2,1:6.-2.1])` **[ENTER]**

360.954	305.708	239.604
352.912	233.495	193.564
298.632	154.599	140.251

sinh⁻¹()

Menú MATH/Hyperbolic

sinh⁻¹(expresión) \Rightarrow *expresión*

sinh⁻¹(lista) \Rightarrow *lista*

sinh⁻¹(expresión) devuelve el seno hiperbólico inverso del argumento como una expresión.

sinh⁻¹(lista) devuelve una lista de los senos hiperbólicos inversos de los elementos de la *lista1*.

`sinh-1(0)` **[ENTER]**

0

`sinh-1({0,2,1,3})` **[ENTER]**

{0 1.487... sinh⁻¹(3)}

$\sinh^{-1}(\text{Matriz cuadrada1}) \Rightarrow \text{Matriz cuadrada}$

Devuelve la matriz de los senos hiperbólicos inversos de la *Matriz cuadrada1*. Esto *no* es lo mismo que calcular el seno hiperbólico inverso de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.

La *Matriz cuadrada1* debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

En el modo Angle en radianes:

$$\sinh^{-1}([1.5, 3; 4.2, 1; 6. -2, 1]) \text{ [ENTER]}$$

$$\begin{bmatrix} .041... & 2.155... & 1.158... \\ 1.463... & .926... & .112... \\ 2.750... & -1.528... & .572... \end{bmatrix}$$

SinReg Menú MATH/Statistics/Regressions

SinReg *lista1, lista2* [, *iteraciones*] [, *periodo*] [, *lista3, lista4*]

Calcula la regresión sinusoidal y actualiza todas las variables estadísticas del sistema.

Todas las listas deben tener el mismo tamaño excepto la *lista4*.

La *lista1* representa *x*lista.

La *lista2* representa *y*lista.

La *lista3* representa códigos de categoría.

La *lista4* representa la lista de categorías.

iteraciones especifica el número máximo de veces (1 a 16) que se intentará obtener una solución. Si se omite, se utiliza 8. Habitualmente, los valores grandes obtienen una mayor precisión pero requieren tiempos de ejecución más largos, y viceversa.

periodo especifica un periodo estimado. Si se omite, la diferencia entre los valores de la *lista1* debe ser igual y en orden secuencial. Si se especifica *periodo*, las diferencias entre los valores *x* pueden ser desiguales.

Nota: Desde la *lista1* hasta la *lista3* deben ser un nombre de variable o c1–c99 (columnas de la última variable de datos mostrada en Data/Matrix Editor). La *lista4* no tiene que ser un nombre de variable y no puede ser c1–c99.

El resultado de **SinReg** siempre se expresa en radianes, independientemente del estado del modo Angle.

En el modo de gráficas de funciones:

seq(x, x, 1, 361, 30) → L1 [ENTER] {1 31 61 ...}

{5.5, 8, 11, 13.5, 16.5, 19, 19.5, 17, 14.5, 12.5, 8.5, 6.5, 5.5} → L2 [ENTER] {5.5 8 11 ...}

SinReg L1, L2 [ENTER] Done

ShowStat [ENTER]

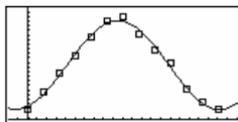


[ENTER] regeq(x) → y1(x) [ENTER] Done

NewPlot 1, 1, L1, L2 [ENTER] Done

[GRAPH]

[F2] 9



solve() Menú MATH/Algebra

solve(*ecuación, var*) ⇒ *expresión booleana*

solve(*desigualdad, var*) ⇒ *expresión booleana*

Devuelve las posibles soluciones reales, de una ecuación o inecuación, para *var*. Su objetivo es devolver todas las soluciones posibles. Sin embargo, puede haber ecuaciones o desigualdades en las que el número de soluciones sea infinito.

Las soluciones pueden no ser reales y finitas en algunos casos.

En el estado AUTO del modo Exact/Approx, el propósito es producir soluciones exactas cuando sean concretas, acompañadas de búsquedas iterativas con aritmética aproximada cuando las soluciones exactas no sean posibles.

Debido a la cancelación por omisión del máximo común divisor del numerador y denominador de fracciones, las soluciones pueden corresponder a sólo uno de los límites laterales.

solve($a \cdot x^2 + b \cdot x + c = 0, x$) [ENTER]

$$x = \frac{\sqrt{-(4 \cdot a \cdot c - b^2)} - b}{2 \cdot a}$$

$$\text{or } x = \frac{-\sqrt{-(4 \cdot a \cdot c - b^2)} + b}{2 \cdot a}$$

ans(1) | a=1 and b=1 and c=1 [ENTER]

Error: Non-real result

solve(($x - a$) $e^x = -x \cdot (x - a), x$) [ENTER]

$x = a$ or $x = -.567...$

($x+1$)($x-1$)/($x-1$)+ $x-3$ [ENTER]

solve(entry(1)=0, x) [ENTER]

entry(2) | ans(1) [ENTER]

limit(entry(3), x, 1) [ENTER]

2 · $x - 2$

$x = 1$

undef

0

Para las desigualdades del tipo $\geq, \leq, < >$, las soluciones explícitas son poco probables, a menos que la desigualdad sea lineal y sólo contenga *var*.

En el estado EXACT del modo Exact/Approx, las partes que no se pueden resolver se devuelven en forma de ecuación o inecuación implícita.

Utilice el operador "!" para restringir el intervalo de la solución y/u otras variables que están en la ecuación o desigualdad. Cuando se halla una solución en un intervalo, puede utilizar los operadores de desigualdad para excluir dicho intervalo en búsquedas posteriores.

Se devuelve false cuando no se obtiene ninguna solución real. Se devuelve true si **solve()** puede determinar que cualquier valor finito real de *var* sirve para la ecuación o desigualdad.

solve() siempre devuelve un resultado booleano, por lo que puede utilizar "and", "or" y "not" para combinar los resultados de **solve()** consigo mismos o con otras expresiones booleanas.

Las soluciones pueden contener una nueva variable no definida en la forma @n*j*, en la que *j* es un número entero comprendido entre 1 y 255. Dichas variables designan un entero arbitrario.

En el modo "Real", las potencias fraccionarias con denominadores impares sólo utilizan la raíz real. Por el contrario, las expresiones con varias raíces, tales como potencias fraccionarias, logaritmos, y funciones trigonométricas inversas, sólo utilizan la raíz principal. En consecuencia, **solve()** sólo halla soluciones que corresponden a dicha raíz real o principal.

Nota: Consulte además **cSolve()**, **cZeros()**, **nSolve()** y **zeros()**.

solve(ecuación1 and ecuación2 [and ...],

{var0Aproximación1, var0Aproximación2 [, ...]}) ⇒ Expresión booleana

Devuelve posibles soluciones reales del sistema de ecuaciones algebraicas, donde cada *var0Aproximación* especifica una incógnita que se desea calcular.

De forma opcional, se puede especificar una aproximación inicial para una incógnita. Cada *var0Aproximación* debe tener la forma:

variable

- 0 -

variable = número real o no real

Por ejemplo, x es válido, lo mismo que x=3.

Si todas las ecuaciones son polinómicas y NO se desea especificar ninguna aproximación inicial,

solve() utiliza el método de eliminación léxica de Gröbner/Buchberger para intentar determinar **todas** las soluciones reales.

Por ejemplo, suponga que tiene una circunferencia de radio *r* centrada en el origen y otra circunferencia de radio *r* de centro el origen en el que la primera circunferencia corta el eje *x* positivo. Utilice **solve()** para hallar las intersecciones.

solve(5x-2 ≥ 2x.x) [ENTER] x ≥ 2/3

exact(solve((x-a)e^x(x)=-x*(x-a),x)) [ENTER] e^x + x = 0 or x = a

En el modo Angle, en radianes:
solve(tan(x)=1/x.x)|x>0 and x<1 [ENTER] x = .860...

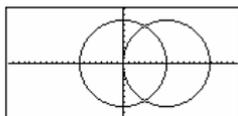
solve(x=x+1,x) [ENTER] false
solve(x=x,x) [ENTER] true

2x-1 ≤ 1 and solve(x^2 ≠ 9,x) [ENTER] x ≤ 1 and x ≠ -3

En el modo Angle, en radianes:
solve(sin(x)=0,x) [ENTER] x = @n1 · π

solve(x^(1/3)=-1,x) [ENTER] x = -1
solve(√(x)=-2,x) [ENTER] false
solve(-√(x)=-2,x) [ENTER] x = 4

solve(y=x^2-2 and x+2y=-1,{x,y}) [ENTER] x=1 and y=-1 or x=-3/2 and y=1/4



Como muestra r en el ejemplo de la derecha, las ecuaciones polinómicas simultáneas pueden tener variables extra que no tengan valores, pero representen valores numéricos dados que puedan sustituirse más adelante.

$$\begin{aligned} &\text{solve}(x^2+y^2=r^2 \text{ and } \\ &(x-r)^2+y^2=r^2, \{x,y\}) \text{ [ENTER]} \\ &x = \frac{r}{2} \text{ and } y = \frac{\sqrt{3}\cdot r}{2} \\ &\text{or } x = \frac{r}{2} \text{ and } y = \frac{-\sqrt{3}\cdot r}{2} \end{aligned}$$

Además, es posible incluir incógnitas que no aparezcan en la ecuación. Por ejemplo, puede incluir z como una incógnita para extender el ejemplo anterior a dos cilindros paralelos de radio r que se cortan.

$$\begin{aligned} &\text{solve}(x^2+y^2=r^2 \text{ and } \\ &(x-r)^2+y^2=r^2, \{x,y,z\}) \text{ [ENTER]} \\ &x = \frac{r}{2} \text{ and } y = \frac{\sqrt{3}\cdot r}{2} \text{ and } z=0 \\ &\text{or } x = \frac{r}{2} \text{ and } y = \frac{-\sqrt{3}\cdot r}{2} \text{ and } z=0 \end{aligned}$$

Las soluciones de los cilindros muestran cómo familias de soluciones pueden contener constantes arbitrarias de la forma $@k$, donde k es un parámetro entero desde 1 hasta 255. El parámetro toma el valor 1 al utilizar **ClrHome** o **[F1] 8:Clear Home**.

Para sistemas de polinomios, el tiempo de cálculo o el consumo de la memoria puede depender en gran medida del orden en el que se listen las variables de las soluciones. Si la primera opción consume la memoria o su paciencia, inténtelo de nuevo reordenando las variables en las ecuaciones y/o la lista de *varO*Aproximación.

Si no se incluye ninguna aproximación y hay alguna ecuación no polinómica en cualquier variable pero todas las ecuaciones son lineales en las incógnitas solve() utiliza el método de eliminación gaussiana para tratar de determinar todas las soluciones reales.

$$\begin{aligned} &\text{solve}(x+e^z)*y=1 \text{ and } \\ &x-y=\sin(z), \{x,y\}) \text{ [ENTER]} \\ &x = \frac{e^z \cdot \sin(z)+1}{e^z+1} \text{ and } y = \frac{-(\sin(z)-1)}{e^z+1} \end{aligned}$$

Si un sistema no es polinómico en todas sus variables ni lineal en sus incógnitas, solve() determina a lo sumo una solución mediante un método iterativo aproximado. Para ello, el número de incógnitas debe ser igual al número de ecuaciones, y todas las demás variables en las ecuaciones deben simplificarse a números.

$$\begin{aligned} &\text{solve}(e^z)*y=1 \text{ and } \\ &-y=\sin(z), \{y,z\}) \text{ [ENTER]} \\ &y=.041\dots \text{ and } z=3.183\dots \end{aligned}$$

Cada incógnita comienza tomando un valor aproximado, si es que existe; de lo contrario, comienza en 0,0.

Utilice aproximaciones para buscar más soluciones una a una. Para que converja, es posible que una aproximación tenga que ser bastante cercana a la solución.

$$\begin{aligned} &\text{solve}(e^z)*y=1 \text{ and } \\ &-y=\sin(z), \{y,z=2\pi\}) \text{ [ENTER]} \\ &y=.001\dots \text{ and } z=6.281\dots \end{aligned}$$

SortA Menú MATH/List

SortA listaNombre1[, listaNombre2] [, listaNombre3] ...
SortA vectorNombre1[, vectorNombre2]
 [, vectorNombre3] ...

Clasifica los elementos del primer argumento en orden ascendente.

Si se incluyen argumentos adicionales, clasifica los elementos de cada uno de forma que sus nuevas posiciones coincidan con las de los elementos del primer argumento.

Todos los argumentos deben ser nombres de listas o vectores. Además, deben tener el mismo tamaño.

```
{2,1,4,3} → list1 [ENTER]      {2,1,4,3}
SortA list1 [ENTER]           Done
list1 [ENTER]                 {1 2 3 4}
{4,3,2,1} → list2 [ENTER]     {4 3 2 1}
SortA list2, list1 [ENTER]     Done
list2 [ENTER]                 {1 2 3 4}
list1 [ENTER]                 {4 3 2 1}
```

SortD Menú MATH/List**SortD** *listaNombre1*, *listaNombre2* [, *listaNombre3*] ...**SortD** *vectorNombre1*, *vectorNombre2* [, *vectorNombre3*] ...Idéntica a **SortA**, excepto que **SortD** clasifica los elementos en orden descendente.{2.1,4.3} → list1 **ENTER**{1.2,3.4} → list2 **ENTER**SortD list1,list2 **ENTER**list1 **ENTER**list2 **ENTER**

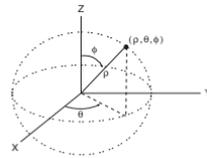
{2 1 4 3}

{1 2 3 4}

Done

{4 3 2 1}

{3 4 1 2}

Sphere Menú MATH/Matrix/Vector ops*vector* **Sphere**Presenta el vector fila o columna en forma esférica [ρ \angle θ \angle ϕ].El *vector* debe tener tres dimensiones y puede ser un vector fila o columna.**Nota:** **Sphere** es una instrucción de formato de visualización, no una función de conversión. Sólo puede utilizarla al final de una línea de entrada.[1.2,3] **Sphere**▾ **ENTER** [3.741... \angle 1.107... \angle .640...][2. \angle $\pi/4$, 3] **Sphere**▾ **ENTER** [3.605... \angle .785... \angle .588...]**ENTER** [$\sqrt{13}$ \angle $\frac{\pi}{4}$ \angle $\cos^{-1}(\frac{3 \cdot \sqrt{13}}{13})$]**startTmr()** CATALOG**startTmr()** ⇒ *entero*Devuelve el valor actual del reloj expresado con un número entero, proporcionando el *tiempo* de un temporizador. El *tiempo* puede introducirse como un argumento en **checkTmr()** para determinar cuántos segundos han transcurrido.

Es posible ejecutar varios temporizadores a la vez.

Nota: Consulte también **checkTmr()** y **timeCnv()**.startTmr() **ENTER**

148083315

checkTmr(148083315)

34

startTmr() → Tempor1

⋮

startTmr() → Tempor2

⋮

checkTmr(Tempor1) → ValorTempor1

⋮

checkTmr(Tempor2) → ValorTempor2

stdDev() Menú MATH/Statistics

stdDev(lista, freclista) \Rightarrow expresión

Devuelve la desviación estándar de los elementos de la *lista*.

Cada elemento *freclista* cuenta el número de apariciones consecutivas del elemento correspondiente en la *lista*.

Nota: La *lista* debe tener al menos dos elementos.

stdDev({a,b,c}) [ENTER]
stdDev({1,2,5,-6,3,-2}) [ENTER]

$$\sqrt{\frac{3 \cdot (a^2 - a \cdot (b+c) + b^2 - b \cdot c)}{2}}$$

stdDev({1.3,2.5,-6.4},{3,2,5}) [ENTER]4

stdDev(matriz1, frecmatriz) \Rightarrow matriz

Devuelve un vector fila de las desviaciones estándar de las columnas en la *matriz1*.

Cada elemento *frecmatriz* cuenta el número de apariciones consecutivas del elemento correspondiente en la *matriz1*.

Nota: La *matriz1* debe tener al menos dos filas.

stdDev([1,2,5;-3,0,1;.5,.7,3]) [ENTER]
[2.179... 1.014... 2]

stdDev([-1,2,5,3;2,5,7,3;6,-4],[4,2;3,3;1,7]) [ENTER]
[2.7005,5.44695]

stdDevPop() Menú MATH/Statistics

stdDevPop(lista, freclista) \Rightarrow expresión

Devuelve la desviación estándar de población de los elementos incluidos en la *lista*.

Cada elemento de *freclista* cuenta el número de veces consecutivas que el elemento correspondiente aparece en la *lista*.

Nota: *lista* debe tener dos elementos como mínimo.

En el modo Angle radianes y en el modo Auto:

stdDevPop({a,b,c}) [ENTER]

$$\sqrt{\frac{3 \cdot (a^2 - a \cdot (b+c) + b^2 - b \cdot c)}{6}}$$

stdDevPop({1,2,5,-6,3,-2}) [ENTER]

$$\sqrt{\frac{465}{6}}$$

stdDevPop({1.3,2.5,-6.4},{3,2,5}) [ENTER]

$$4.11107$$

stdDevPop(matriz1, frecmatriz) \Rightarrow matriz

Devuelve un vector de fila de las desviaciones estándar de población de las columnas incluidas en la *matriz1*.

Cada elemento de *frecmatriz* cuenta el número de veces consecutivas que el elemento correspondiente aparece en la *matriz1*.

Nota: *matriz1* debe tener dos filas como mínimo.

stdDevPop([[1,2,5]-3,0,1][.5,.7,3]]) [ENTER]

$$\left[1.77951 \quad .828654 \quad \frac{2\sqrt{6}}{3} \right]$$

stdDevPop([-1,2,5,3;2,5,7,3;6,-4],[2,3,3;1,7]) [ENTER]

$$\left[4 \quad ; \quad 2,5 \quad 7,3 \quad , \quad 3 \quad ; \quad 6 \quad -4 \quad , \quad 1 \quad ; \quad 2,52608 \quad 5,21506 \right]$$

StoGDB CATALOG

StoGDB *GDBvar*

Crea una variable de base de datos gráfica (GDB) que contiene lo siguiente:

- * Modo de representación gráfica
- * Funciones Y=
- * Variables de ventana
- * Estados del formato gráfico
- Estado de One o Two-Graph (pantalla dividida y ajuste de proporciones en el modo Two-Graph)
- Modo Angle
- Modo Real/Complex
- * Condiciones iniciales si se está en el modo Sequence o en el modo Diff Equations
- * Indicadores de tabla
- * tblStart, Δtbl, tblInput

Puede utilizar **RclGDB** *GDBvar* para restablecer el entorno gráfico.

***Nota:** Estos elementos se guardan para ambos gráficos en el modo Two-Graph.

Stop CATALOG

Stop

Se utiliza como instrucción de un programa para detener la ejecución del mismo.

Parte de un programa:

```
:  
:  
For i,1,10,1  
  If i=5  
    Stop  
  EndFor  
:  
:
```

StoPic CATALOG

StoPic *picVar* [, *pxlFila*, *pxlCol*] [, *ancho*, *superior*]

Presenta la pantalla Graph y copia un área rectangular de la pantalla en la variable *picVar*.

pxlFila y *pxlCol*, si se incluyen, especifican la esquina superior izquierda del área que se va a copiar (por omisión son 0, 0).

Los valores de *ancho* y *superior*, si se incluyen, especifican las dimensiones, en pixels, del área. Por omisión, son el ancho y la altura en pixels de la pantalla Graph actual.

Store Consulte → (almac.), página 209.

string() Menú MATH/String

string(*expresión*) ⇒ *cadena*

Simplifica la *expresión* y devuelve el resultado como una cadena de caracteres.

```
string(1.2345) [ENTER] "1.2345"  
string(1+2) [ENTER] "3"  
string(cos(x)+√(3)) [ENTER] "cos(x) + √(3)"
```

Style CATALOG

Style *numecua*, *CadenaPropiedadestilo*

Ajusta la función *numecua* del sistema en el modo gráfico actual para utilizar la propiedad *CadenaPropiedadestilo*.

numecua debe ser un número entero comprendido entre 1 y 99, debiendo estar la función previamente definida.

CadenaPropiedadestilo debe ser de uno de los siguientes tipos: "Line", "Dot", "Square", "Thick", "Animate", "Path", "Above" o "Below".

Tenga en cuenta que en las gráficas en paramétricas, sólo la parte *xt* del par contiene la información del estilo.

Nombres de estilos válidos para los modos de representación:

Function: todos los estilos
 Parametric/Polar: line, dot, square, thick, animate, path
 Sequence: line, dot, square, thick
 3D: ninguno
 Diff Equations: line, dot, square, thick, animate, path

Nota: Las mayúsculas y los espacios en blanco son opcionales al introducir nombres de *PropiedadesCadenasestilo*.

Style 1."thick" **[ENTER]**

Done

Style 10."path" **[ENTER]**

Done

Nota: En el modo de gráficas de función, estos ejemplos ajustan el estilo de $y1(x)$ en "Thick" e $y10(x)$ en "Path".

subMat() CATALOG

subMat(*matriz1*, *inicioFila* [, *inicioCo*] [, *finFila*] [, *finCo*]) \Rightarrow *matriz*

Devuelve la submatriz indicada de la *matriz1*.

Por omisión: *inicioFila*=1, *inicioCo*=1, *finFila*=última fila, *finCo*=última columna.

[1.2.3:4.5.6:7.8.9] \rightarrow m1 **[ENTER]**

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

subMat(m1,2.1.3.2) **[ENTER]**

$$\begin{bmatrix} 4 & 5 \\ 7 & 8 \end{bmatrix}$$

subMat(m1.2.2) **[ENTER]**

$$\begin{bmatrix} 5 & 6 \\ 8 & 9 \end{bmatrix}$$

sum() Menú MATH/List

sum(*lista*, *primero*, *último*) \Rightarrow *expresión*

Devuelve la suma de los elementos de la *lista*.

Primero y *último* son opcionales, y especifican un rango de elementos.

sum({1.2.3.4.5}) **[ENTER]**

15

sum({a.2a.3a}) **[ENTER]**

$6 \cdot a$

sum(seq(n.n.1.10)) **[ENTER]**

55

sum({1.3.5.7.9}.3) **[ENTER]**

21

sum(*matriz1*, *primero*, *último*) \Rightarrow *matriz*

Devuelve un vector fila que contiene las suma de todos los elementos de las columnas de la *matriz1*.

Primero y *último* son opcionales, y especifican un rango de filas.

sum([1.2.3;4.5.6]) **[ENTER]**

[5 7 9]

sum([1.2.3;4.5.6;7.8.9]) **[ENTER]**

[12 15 18]

sum([1.2.3;4.5.6;7.8.9].2.3)

[ENTER]

[11.13.15]

Suma() Consulte #(), página 209.

switch() CATALOG

switch([entero1]) ⇒ entero

Devuelve el número de la ventana activa.
También puede configurar la ventana activa.

Nota: Window 1 es la ventana izquierda o superior y Window 2 es la derecha o inferior.

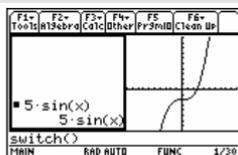
Si *entero1* = 0, devuelve el número de la ventana activa.

Si *entero1* = 1, activa la ventana 1 y devuelve el número de la ventana activa anterior.

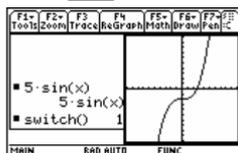
Si *entero1* = 2, activa la ventana 2 y devuelve el número de la ventana activa anterior.

Si se omite *entero1*, conmuta entre ventanas y devuelve el número de la ventana activa anterior.

entero1 se ignora si la TI-89 Titanium / Voyage™ 200 no presenta la pantalla dividida.



switch [ENTER]



T (trasp.)

Menú MATH/Matrix

matriz^T ⇒ matriz

Devuelve la matriz traspuesta de la dada.

[1.2.3:4.5.6:7.8.9] ⇒ mat1 [ENTER]

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

mat1^T [ENTER]

$$\begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$$

[a.b:c.d] ⇒ mat2 [ENTER]

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

mat2^T [ENTER]

$$\begin{bmatrix} a & c \\ b & d \end{bmatrix}$$

[1+i.2+i:3+i.4+i] ⇒ mat3 [ENTER]

$$\begin{bmatrix} 1+i & 2+i \\ 3+i & 4+i \end{bmatrix}$$

mat3^T [ENTER]

$$\begin{bmatrix} 1-i & 3-i \\ 2-i & 4-i \end{bmatrix}$$

Table CATALOG

Table *expresión1*, *expresión2* [, *var1*]

Crea la tabla de las expresiones o funciones que se indiquen.

Las expresiones de la tabla también se pueden representar gráficamente. Las expresiones introducidas con las órdenes **Table** o **Graph** reciben números de función que se incrementan empezando en 1. Las expresiones pueden modificarse o borrarse individualmente utilizando las funciones de edición disponibles al mostrar la tabla pulsando [F4] Header. Las funciones seleccionadas actualmente en Y= Editor se ignoran temporalmente.

Para borrar las funciones creadas mediante **Table** o **Graph**, ejecute la orden **ClrGraph** o presente Y= Editor.

Si se omite el parámetro de *var*, se utiliza la variable independiente del modo gráfico actual. Algunas variaciones válidas de esta instrucción son las siguientes:

Gráficas de función: **Table** *expr*, *x*

Gráficas en paramétricas: **Table** *xExpr*, *yExpr*, *t*

Gráficas en polares: **Table** *expr*, θ

Nota: La orden **Table** no es válida para la representación de gráficas en 3D, sucesiones o ecuaciones diferenciales. Como alternativa, es posible que desee utilizar **BldData**.

En el modo de gráficas de función:

Table 1.25x*cos(x) [ENTER]

x	1		
0.	0.		
1.	.67538		
2.	-1.04		
3.	-3.712		
4.	-3.268		

Table cos(time).time [ENTER]

x	1	2	3
0.	0.	1.	
1.	.67538	.5403	
2.	-1.04	-4.161	
3.	-3.712	-.99	
4.	-3.268	-.6536	

tan()

 **Tecla** [2nd] [TAN]

 **Tecla** [TAN]

tan(*expresión1*) \Rightarrow *expresión*

tan(*lista1*) \Rightarrow *lista*

tan(*expresión1*) devuelve la tangente del argumento.

tan(*lista1*) devuelve la lista de las tangentes de todos los elementos de *lista1*.

Nota: El argumento se interpreta como un ángulo expresado en grados, grados centesimales o radianes, según el modo de ángulo actual. Puede utilizar $^{\circ}$, $^{\circ}$ o $^{\circ}$ para anular temporalmente el valor de modo de ángulo actual.

En el modo Angle, en grados:

tan($(\pi/4)^{\circ}$) [ENTER] 1

tan(45) [ENTER] 1

tan({0.60,90}) [ENTER] {0 $\sqrt{3}$ undef}

En el modo Angle, en grados centesimales:

tan($(\pi/4)^{\circ}$) [ENTER] $\frac{200 \cdot \tan\left(\frac{\pi}{4}\right)}{\pi}$

tan(50) [ENTER] 1

tan({0.50,100}) [ENTER] {0 1 undef}

En el modo Angle, en radianes:

tan($\pi/4$) [ENTER] 1

tan(45 $^{\circ}$) [ENTER] 1

tan({ π , $\pi/3$, $-\pi$, $\pi/4$ }) [ENTER] {0 $\sqrt{3}$ 0 1}

tan(*Matriz cuadrada*) \Rightarrow *Matriz cuadrada*

Devuelve la matriz tangente de *Matriz cuadrada1*. Esto *no* es lo mismo que calcular la tangente de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.

La *Matriz cuadrada1* debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

En el modo Angle en radianes:

tan([1.5,3;4.2,1;6.-2,1]) **ENTER**

-28.291...	26.088...	11.114...
12.117...	-7.835...	-5.481...
36.818...	-32.806...	-10.459...

tan⁻¹()

Tecla [TAN⁻¹]

Tecla [2nd] [TAN⁻¹]

tan⁻¹(*expresión*) \Rightarrow *expresión*

tan⁻¹(*lista*) \Rightarrow *lista*

tan⁻¹(*expresión*) devuelve el ángulo cuya tangente es *expresión1*.

tan⁻¹(*lista*) devuelve la lista de los arcotangentes de los elementos de *lista1*.

Nota: El resultado devuelve un ángulo expresado en grados, grados centesimales o radianes, según el valor de modo de ángulo actual.

En el modo Angle, en grados:

tan⁻¹(1) **ENTER** 45

En el modo Angle, en grados centesimales:

tan⁻¹(1) **ENTER** 50

En el modo Angle, en radianes:

tan⁻¹({0.,.2.,.5}) **ENTER**
{0 .197... .463...}

tan⁻¹(*Matriz cuadrada*) \Rightarrow *Matriz cuadrada*

Devuelve la matriz arcotangente de *Matriz cuadrada1*. Esto *no* es lo mismo que calcular el arcotangente de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.

Matriz cuadrada1 debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

En el modo Angle en radianes:

tan⁻¹([1.5,3;4.2,1;6.-2,1]) **ENTER**

-.083...	1.266...	.622...
.748...	.630...	-.070...
1.686...	-1.182...	.455...

tanh()

Menú MATH/Hyperbolic

tanh(*expresión*) \Rightarrow *expresión*

tanh(*lista*) \Rightarrow *lista*

tanh(*expresión*) devuelve la tangente hiperbólica del argumento.

tanh(*lista*) devuelve la lista de las tangentes hiperbólicas de los elementos de *lista1*.

tanh(1.2) **ENTER** .833...

tanh({0,1}) **ENTER** {0 tanh(1)}

tanh(*Matriz cuadrada*) \Rightarrow *Matriz cuadrada*

Devuelve la matriz tangente hiperbólica de *Matriz cuadrada1*. Esto *no* es lo mismo que calcular la tangente hiperbólica de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.

La *Matriz cuadrada1* debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

En el modo Angle en radianes:

tanh([1.5,3;4.2,1;6.-2,1]) **ENTER**

-.097...	.933...	.425...
.488...	.538...	-.129...
1.282...	-1.034...	.428...

tanh⁻¹() Menú MATH/Hyperbolic

tanh⁻¹(expresión) ⇒ expresión

tanh⁻¹(lista) ⇒ lista

tanh⁻¹(expresión) devuelve la tangente hiperbólica inversa del argumento como una expresión.

tanh⁻¹(lista) devuelve la lista de las tangentes hiperbólicas inversas de los elementos de *lista*.

tanh⁻¹(Matriz cuadrada) ⇒ Matriz cuadrada

Devuelve la matriz tangente hiperbólica inversa de *Matriz cuadrada*. Esto *no* es lo mismo que calcular la tangente hiperbólica inversa de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.

La *Matriz cuadrada* debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

En el modo de formato complejo rectangular:

tanh⁻¹(0) [ENTER] 0

tanh⁻¹({1.2,1.3}) [ENTER]
{∞ .518... -1.570... · i $\frac{\ln(2)}{2} - \frac{\pi}{2} \cdot i$ }

En el modo Angle en radianes y en el modo de formato complejo rectangular:

tanh⁻¹([1.5,3;4,2,1.6,-2.1]) [ENTER]
$$\begin{bmatrix} -.099...+.164... \cdot i & .267...-1.490... \cdot i & \dots \\ -.087...-.725... \cdot i & .479...-.947... \cdot i & \dots \\ .511...-2.083... \cdot i & -.878...+1.790... \cdot i & \dots \end{bmatrix}$$

taylor() Menú MATH/Calculus

taylor(expresión, var, orden, punto) ⇒ expresión

Devuelve el polinomio de Taylor pedido. El polinomio está formado por los términos distintos de cero de grados comprendidos entre cero y *orden*, en un entorno de *var* menos *punto*. **taylor()** se devuelve sin cambios si no hay ninguna serie truncada de potencias de este orden, o si se requieren exponentes fraccionarios o negativos. Utilice una sustitución y/o multiplicación temporal por la potencia de (*var* menos *punto*) para determinar una serie de potencias más genéricas.

El valor de *punto* es cero por omisión, y es el centro del desarrollo.

taylor(e^{√(x)},x,2) [ENTER]

taylor(e^t,t,4)|t=√(x) [ENTER]

■ taylor(e^{√x},x,2)
taylor(e^{√x},x,2,θ)
■ taylor(e^t,t,4)|t=√x
 $\frac{x^2}{24} + \frac{x^{3/2}}{6} + \frac{x}{2} + \sqrt{x} + 1$

taylor(1/(x*(x-1)),x,3) [ENTER]

■ taylor($\frac{1}{x \cdot (x-1)}$,x,3)
taylor($\frac{1}{x \cdot (x-1)}$,x,3,θ)

expand(taylor(x/(x*(x-1)),x,4)/x,x)
[ENTER]

■ expand($\frac{\text{taylor}(\frac{x}{x \cdot (x-1)},x,4)}{x}$,x)
 $-x^3 - x^2 - x - \frac{1}{x} - 1$

tCollect() Menú MATH/AlgebraTrig

tCollect(expresión) ⇒ expresión

Devuelve una expresión en la que los productos y potencias enteras de senos y cosenos se convierten en una combinación lineal de senos y cosenos de varios ángulos, sumas de ángulos o restas de ángulos. La transformación convierte los polinomios trigonométricos en una combinación lineal de sus valores armónicos.

Algunas veces **tCollect()** cumple los objetivos cuando la simplificación trigonométrica por omisión no lo permite. **tCollect()** tiende a invertir las transformaciones efectuadas con **tExpand()**. Algunas veces, si se aplica **tExpand()** a un resultado de **tCollect()** o viceversa, en dos pasos separados, se simplifica una expresión.

tCollect((cos(α))^2) [ENTER]
 $\frac{\cos(2 \cdot \alpha) + 1}{2}$

tCollect(sin(α)cos(β)) [ENTER]
 $\frac{\sin(\alpha - \beta) + \sin(\alpha + \beta)}{2}$

tExpand() Menú MATHAlgebraTrig

tExpand(*expresión1*) ⇒ *expresión*

Devuelve una expresión en la que se desarrollan los senos y cosenos de varios ángulos enteros, sumas de ángulos o restas de ángulos. Debido a la identidad $(\sin(x))^2 + (\cos(x))^2 = 1$, hay muchos resultados equivalentes posibles. En consecuencia, los resultados pueden variar de unas publicaciones a otras.

Algunas veces, **tExpand()** cumple los objetivos cuando la simplificación trigonométrica por omisión no lo permite. **tExpand()** tiende a invertir las transformaciones realizadas con **tCollect()**. A veces, al aplicar **tCollect()** a un resultado de **tExpand()**, o viceversa, en dos pasos separados, se simplifica una expresión.

Nota: La conversión por $\pi/180$ en el modo de grados interfiere con la capacidad de **tExpand()** para reconocer todas las formas desarrollables. Para obtener los mejores resultados, **tExpand()** debe utilizarse en el modo de radianes.

tExpand($\sin(3\phi)$) **ENTER**

$$4 \cdot \sin(\phi) \cdot (\cos(\phi))^2 - \sin(\phi)$$

tExpand($\cos(\alpha - \beta)$) **ENTER**

$$\cos(\alpha) \cdot \cos(\beta) + \sin(\alpha) \cdot \sin(\beta)$$

Text CATALOG

Text *promptCadena*

Presenta el recuadro de diálogo de la cadena de caracteres *promptCadena*.

Si se utiliza como parte de un bloque **Dialog...EndDialog**, la cadena *promptCadena* se presenta dentro del recuadro de diálogo. Si se utiliza como instrucción separada, **Text** crea un recuadro de diálogo para presentar la cadena.

Text "Have a nice day." **ENTER**

Done



Then Consulte **If**, página 209.

timeCnv() CATALOG

timeCnv(*segundos*) ⇒ *lista*

Convierte segundos a unidades de tiempo que pueden ser más fáciles de comprender al evaluar. La lista tiene el formato {*días, horas, minutos, segundos*}.

Nota: Consulte también **checkTmr()** y **startTmr()**.

timeCnv(152442117)

{1764 9 1 57}

Title CATALOG

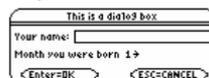
Title *títuloCadena*, [*Lb*]

Crea el título de un menú desplegable o recuadro de diálogo cuando se utiliza dentro de una construcción **Toolbar** o **Custom**, o en un bloque **Dialog...EndDialog**.

Nota: *Lb* sólo es válido en una construcción **Toolbar**. Cuando está presente, permite que la opción de menú se traslade a una etiqueta dentro del programa.

Parte de un programa:

```
:  
:Dialog  
:Title "This is a dialog box"  
:Request "Your name".Str1  
:Dropdown "Month you were born",  
seq(string(i),i,1,12).Var1  
:EndDialog  
:
```



tmpCnv() CATALOG

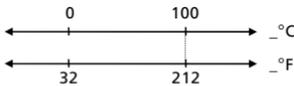
tmpCnv(expresión1_°tempUnidad1,_°tempUnidad2)
⇒ expresión_°tempUnidad2

Convierte el valor de temperatura especificado por *expresión1* de una unidad a otra. Las unidades de temperatura válidas son:

_°C Celsius
_°F Fahrenheit
_°K Kelvin
_°R Rankine

Para °, pulse [2nd] [°].
Para _, pulse [↓] [°].
Para _, pulse [2nd] [°].

Por ejemplo, 100_°C se convierte a 212_°F:



Para convertir un rango de temperaturas, utilice **ΔtmpCnv()**.

tmpCnv(100_°c._°f) [ENTER] 212.°_°F
tmpCnv(32_°f._°c) [ENTER] 0.°_°C
tmpCnv(0_°c._°k) [ENTER] 273.15_°°K
tmpCnv(0_°f._°r) [ENTER] 459.67_°_°R

Nota: Para seleccionar unidades de temperatura en un menú, pulse:

[↓] [2nd] [UNITS]
[↓] [UNITS]

ΔtmpCnv() CATALOG

ΔtmpCnv(expresión1_°tempUnidad1,_°tempUnidad2)
⇒ expresión_°tempUnidad2

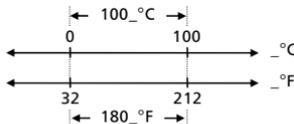
Convierte un rango de temperaturas (la diferencia entre dos valores de temperatura) especificada por *expresión1* de una unidad a otra. Las unidades de temperatura válidas son:

_°C Celsius
_°F Fahrenheit
_°K Kelvin
_°R Rankine

Para °, pulse [2nd] [°].
Para _, pulse [↓] [°].
Para _, pulse [2nd] [°].

1_°C y 1_°K tienen la misma magnitud, al igual que 1_°F y 1_°R. No obstante, 1_°C equivale a 9/5 de 1_°F.

Por ejemplo, un rango de 100_°C (desde 0_°C a 100_°C) equivale a un rango de 180_°F:



Para convertir un valor de temperatura concreto en vez de un rango, utilice **tmpCnv()**.

Para obtener Δ, puede pulsar [↓] [↑] [D] (o [2nd] [CHAR] 1 5).

ΔtmpCnv(100_°c._°f) [ENTER] 180.°_°F
ΔtmpCnv(180_°f._°c) [ENTER] 100.°_°C
ΔtmpCnv(100_°c._°k) [ENTER] 100.°_°K
ΔtmpCnv(100_°f._°r) [ENTER] 100.°_°R
ΔtmpCnv(1_°c._°f) [ENTER] 1.8_°_°F

Nota: Para seleccionar unidades de temperatura en un menú, pulse:

[↓] [2nd] [UNITS]
[↓] [UNITS]

Toolbar CATALOG

Toolbar
bloque
EndTBar

Crea un menú en la barra de herramientas.

El *bloque* puede ser un sólo enunciado o una sucesión de enunciados separados por el carácter ":". Los enunciados pueden ser Title o Item.

Item debe tener etiquetas. Title también debe tener una etiqueta si no contiene un Item.

```
Parte de un programa:  
:  
:Toolbar  
: Title "Examples"  
: Item "Trig", t  
: Item "Calc", c  
: Item "Stop", Pexit  
:EndTbar  
:
```

Nota: Cuando se ejecuta en un programa, se crea un menú con tres opciones que dirigen a tres sitios del programa.

Trace CATALOG

Trace

Dibuja un Smart Graph y sitúa el cursor Traza en la primera función definida de $Y=$, en la posición anterior del cursor o en la posición de reinicio, si se volvió a dibujar la gráfica.

Permite el funcionamiento del cursor y de la mayoría de las teclas cuando se editan valores de coordenadas. Algunas teclas, como las teclas de función **[APPS]** y **[MODE]**, no están activas durante el trazado.

Nota: Pulse **[ENTER]** para reanudar la operación.

Try CATALOG

Try
bloque1
Else
bloque2
EndTry

Ejecuta *bloque1* a menos que ocurra un error. La ejecución del programa se transfiere a *bloque2* si se produce un error en *bloque1*. La variable *errornum* contiene el número de error que permite al programa realizar su recuperación.

El *bloque1* y el *bloque2* pueden ser un único enunciado o una serie de varios enunciados separados por el carácter ":".

```
Parte de un programa:  
:  
:Try  
: NewFold(temp)  
: Else  
: Already exists  
: ClrErr  
:EndTry  
:
```

Nota: Consulte **ClrErr** y **PassErr**.

TwoVar Menú MATH/Statistics

TwoVar *lista1, lista2, [lista3], [lista4, lista5]*

Calcula las estadísticas de **TwoVar** y actualiza todas las variables estadísticas del sistema.

Todas las listas deben tener el mismo tamaño, excepto la *lista5*.

La *lista1* representa *x*lista.

La *lista2* representa *y*lista.

La *lista3* representa la frecuencia.

La *lista4* representa códigos de categoría.

La *lista5* representa la lista de categorías.

Nota: Desde la *lista1* hasta la *lista4* deben ser un nombre de variable o c1–c99 (columnas de la última variable de datos mostrada en Data/Matrix Editor). La *lista5* no tiene que ser un nombre de variable y no puede ser c1–c99.

{0.1.2.3.4.5.6} → L1 **[ENTER]**

{0 1 2 ...}

{0.2.3.4.3.4.6} → L2 **[ENTER]**

{0 2 3 ...}

TwoVar L1,L2 **[ENTER]**

Done

ShowStat **[ENTER]**

STAT VARS	
\bar{x}	=3.
\bar{y}	=3.142857
Σx	=21.
Σx^2	=91.
Σy	=22.
Σy^2	=86.
Σxy	=88.
S_x	=2.160247
S_y	=1.869454
r	=.7.
r^2	=.49.
r^3	=.343.
r^4	=.2401.
r^5	=.16807.
r^6	=.117649.
r^7	=.0823543.
r^8	=.057648.
r^9	=.0403506.
r^{10}	=.028243.
r^{11}	=.0197706.
r^{12}	=.0139314.
r^{13}	=.0097478.
r^{14}	=.00684257.
r^{15}	=.00472496.
r^{16}	=.00328171.
r^{17}	=.00228768.
r^{18}	=.00158543.
r^{19}	=.00108986.
r^{20}	=.000746496.
r^{21}	=.000516547.
r^{22}	=.000347882.
r^{23}	=.000237517.
r^{24}	=.000160262.
r^{25}	=.000105185.
r^{26}	=.0000697706.
r^{27}	=.000045843.
r^{28}	=.0000299081.
r^{29}	=.0000196518.
r^{30}	=.0000129543.
r^{31}	=.86E-05.
r^{32}	=.57E-06.
r^{33}	=.36E-07.
r^{34}	=.23E-08.
r^{35}	=.15E-09.
r^{36}	=.95E-11.
r^{37}	=.61E-12.
r^{38}	=.38E-13.
r^{39}	=.24E-14.
r^{40}	=.15E-15.
r^{41}	=.95E-17.
r^{42}	=.61E-18.
r^{43}	=.38E-19.
r^{44}	=.24E-20.
r^{45}	=.15E-21.
r^{46}	=.95E-23.
r^{47}	=.61E-24.
r^{48}	=.38E-25.
r^{49}	=.24E-26.
r^{50}	=.15E-27.
r^{51}	=.95E-29.
r^{52}	=.61E-30.
r^{53}	=.38E-31.
r^{54}	=.24E-32.
r^{55}	=.15E-33.
r^{56}	=.95E-35.
r^{57}	=.61E-36.
r^{58}	=.38E-37.
r^{59}	=.24E-38.
r^{60}	=.15E-39.
r^{61}	=.95E-41.
r^{62}	=.61E-42.
r^{63}	=.38E-43.
r^{64}	=.24E-44.
r^{65}	=.15E-45.
r^{66}	=.95E-47.
r^{67}	=.61E-48.
r^{68}	=.38E-49.
r^{69}	=.24E-50.
r^{70}	=.15E-51.
r^{71}	=.95E-53.
r^{72}	=.61E-54.
r^{73}	=.38E-55.
r^{74}	=.24E-56.
r^{75}	=.15E-57.
r^{76}	=.95E-59.
r^{77}	=.61E-60.
r^{78}	=.38E-61.
r^{79}	=.24E-62.
r^{80}	=.15E-63.
r^{81}	=.95E-65.
r^{82}	=.61E-66.
r^{83}	=.38E-67.
r^{84}	=.24E-68.
r^{85}	=.15E-69.
r^{86}	=.95E-71.
r^{87}	=.61E-72.
r^{88}	=.38E-73.
r^{89}	=.24E-74.
r^{90}	=.15E-75.
r^{91}	=.95E-77.
r^{92}	=.61E-78.
r^{93}	=.38E-79.
r^{94}	=.24E-80.
r^{95}	=.15E-81.
r^{96}	=.95E-83.
r^{97}	=.61E-84.
r^{98}	=.38E-85.
r^{99}	=.24E-86.
r^{100}	=.15E-87.

Unarchiv CATALOG

Unarchiv *var1* [, *var2*] [, *var3*] ...

Desplaza las variables especificadas desde la memoria de archivos de datos del usuario hasta la RAM.

Puede acceder a una variable archivada del mismo modo que lo haría con una variable en la RAM. No obstante, no es posible borrar, renombrar o almacenar una variable archivada debido a que se bloquea de forma automática.

Para archivar variables, utilice **Archive**.

10→arctest [ENTER] 10
 Archive arctest [ENTER] Done
 5*arctest [ENTER] 50
 15→arctest [ENTER]



[ESC]
 Unarchiv arctest [ENTER] Done
 15→arctest [ENTER] 15

unitV() Menú MATH/Matrix/Vector ops

unitV(*vector1*) ⇒ *vector*

Devuelve un vector fila o columna unitario, dependiendo de la forma del *vector1*.

El *vector1* debe ser una matriz de fila única o una matriz de columna única.

unitV([a,b,c]) [ENTER]

$$\left[\frac{a}{\sqrt{a^2+b^2+c^2}} \quad \frac{b}{\sqrt{a^2+b^2+c^2}} \quad \frac{c}{\sqrt{a^2+b^2+c^2}} \right]$$

unitV([1,2,1]) [ENTER]

$$\left[\frac{\sqrt{6}}{6} \quad \frac{\sqrt{6}}{3} \quad \frac{\sqrt{6}}{6} \right]$$

unitV([1:2:3]) [ENTER]

$$\begin{bmatrix} \frac{\sqrt{14}}{14} \\ \frac{\sqrt{14}}{7} \\ \frac{3 \cdot \sqrt{14}}{14} \end{bmatrix}$$

Unlock CATALOG

Unlock *var1*, *var2* [, *var3*]...

Desbloquea las variables especificadas.

Nota: Las variables se pueden bloquear utilizando la orden **Lock**.

variance() Menú MATH/Statistics

variance(*lista*, *freclista*) ⇒ *expresión*

Devuelve la varianza de la *lista*.

Cada elemento *freclista* cuenta el número de apariciones consecutivas del elemento correspondiente en la *lista*.

Nota: La *lista* debe contener al menos dos elementos.

variance({a,b,c}) [ENTER]

$$\frac{a^2 - a \cdot (b+c) + b^2 - b \cdot c + c^2}{3}$$

variance({1,2,5,-6,3,-2}) [ENTER]

31/2

variance({1,3,5},{4,6,2}) [ENTER]

68/33

variance(*matriz1* [, *frecmatriz2*]) ⇒ *matriz*

Devuelve un vector fila que contiene la varianza de cada columna de la *matriz1*.

Cada elemento *frecmatriz2* cuenta el número de apariciones consecutivas del elemento correspondiente en la *matriz1*.

Nota: La *matriz1* debe contener al menos dos filas.

variance([1,2,5;-3,0,1;

.5;.7,3]) [ENTER] [4.75 1.03 4]

variance([-1,1,2,2;3,4,5,1;

-2,3,4,3],[6,3;2,4;5,1]) [ENTER] [3.91731,2.08411]

when() CATALOG

when(condición, verdaderoResultado [, falsoResultado] [, desconocidoResultado]) ⇒ expresión

Devuelve *verdaderoResultado*, *falsoResultado* o *desconocidoResultado*, dependiendo de si la *condición* es verdadera, falsa o desconocida. Devuelve la entrada si no hay argumentos suficientes para especificar el resultado.

Omite tanto *falsoResultado* como *desconocidoResultado* para que una expresión sólo esté definida en la región en que la *condición* es verdadera.

Utilice undef *falsoResultado* para definir una expresión cuya gráfica sólo se representa en un intervalo.

Omíta sólo *desconocidoResultado* para definir una expresión de dos partes.

Encadene **when()** para definir expresiones que tienen más de dos partes.

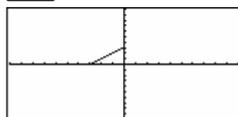
when() es útil para definir funciones recursivas.

when(x<0, x+3) | x=5 **[ENTER]**

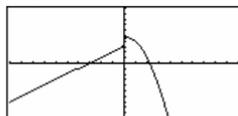
when(x<0, 3+x)

C1rGraph **[ENTER]**

Graph when(x≥-π and x<0, x+3, undef) **[ENTER]**



Graph when(x<0, x+3, 5-x^2) **[ENTER]**



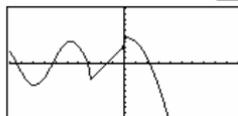
[HOME]

[CALC HOME]

C1rGraph **[ENTER]**

Graph when(x<0, when(x<-π, 4* sin(x), 2x+3), 5-x^2) **[ENTER]**

Done



when(n>0, n* factorial(n-1), 1)

→ factorial(n) **[ENTER]**

Done

factorial(3) **[ENTER]**

6

3! **[ENTER]**

6

While CATALOG

While condición

bloque

EndWhile

Ejecuta los enunciados de *bloque* siempre que la *condición* sea verdadera.

El *bloque* puede ser un solo enunciado o una sucesión de varios enunciados separados por el carácter ":".

Parte de un programa:

```

:
:1→i
:0→temp
:While i<=20
: temp+1/i→temp
: i+1→i
:EndWhile
:Disp "sum of reciprocals up to
20", temp
:

```

"With"

Consulte | página 209.

XOR **Menú MATH/Test**

expresión booleana1 **xor** *expresión booleana2* ⇒
expresión booleana

true xor true **[ENTER]** false
(5>3) xor (3>5) **[ENTER]** true

Devuelve true si la *expresión booleana1* es verdadera y la *expresión booleana2* es falsa, o viceversa.
Devuelve false si la *expresión booleana1* y la *expresión booleana2* son verdaderas o falsas.
Devuelve una expresión booleana simplificada si alguna de las expresiones booleanas originales no puede resolverse en true o false.

Nota: Consulte **or**.

entero1 **xor** *entero2* ⇒ *entero*

En el modo de base Hex:

Compara dos números enteros reales bit a bit mediante una operación **xor**. Internamente, ambos enteros se convierten en números binarios de 32 bits con su signo correspondiente. Cuando se comparan bits correspondientes, el resultado es 1 si uno de los dos bits (no ambos) es 1; el resultado es 0 si ambos bits son 0 o ambos bits son 1. El valor devuelto representa los resultados de bits, y se presenta de acuerdo con el estado del modo Base.

0h7AC36 xor 0h3D5F **[ENTER]** 0h79169
↳ **Importante:** Cero, no la letra O.

En el modo de base Bin:

Los números enteros pueden introducirse en cualquier base de numeración. Para una entrada binaria o hexadecimal, debe utilizarse el prefijo 0b ó 0h, respectivamente. Sin prefijo, los enteros se tratan como decimales (base 10).

0b100101 xor 0b100 **[ENTER]** 0b100001

Nota: Las entradas binarias pueden tener hasta 32 dígitos (sin contar el prefijo 0b); las hexadecimales, un máximo 8 dígitos.

Si se introduce un entero decimal demasiado grande para una forma binaria de 32 bits con su signo correspondiente, se utiliza una operación de módulos simétricos para llevar el valor al rango apropiado.

Nota: Consulte **or**.

XorPic **CATALOG**

XorPic *picVar*, *fila* [, *columna*]

Presenta en la pantalla Graph actual la imagen almacenada en *picVar*.

Utiliza lógica **xor** para cada pixel. Sólo se activan los pixels en las posiciones no exclusivas de la pantalla o la imagen. Esta instrucción desactiva los pixels que están activados en ambas imágenes.

La variable *picVar* debe contener un tipo de datos "pic".

La *fila* y la *columna*, si se incluyen, especifican las coordenadas del pixel en la esquina superior izquierda de la imagen. Los valores por omisión son (0, 0).

zeros() **Menú MATH/Algebra**

zeros(*expresión*, *var*) ⇒ *lista*

Devuelve una lista de posibles valores reales de *var* que hacen *expresión*=0. **zeros()** lo realiza calculando **exp|list(solve(expresión=0, var), var)**.

zeros($a \cdot x^2 + b \cdot x + c$) **[ENTER]**

$$\left\{ \frac{-\sqrt{b^2 - 4 \cdot a \cdot c} - b}{2 \cdot a}, \frac{\sqrt{b^2 - 4 \cdot a \cdot c} - b}{2 \cdot a} \right\}$$

$a \cdot x^2 + b \cdot x + c | x = \text{ans}(1)[2]$ **[ENTER]**

0

En algunos casos, la forma de resultados de **zeros()** es más conveniente que la de **solve()**. Sin embargo, la forma de resultados de **zeros()** no puede expresar soluciones implícitas, soluciones que requieren desigualdades o soluciones que no utilizan *var*.

Nota: Consulte además **cSolve()**, **cZeros()** y **solve()**.

```
exact(zeros(a*(e^(x)+x)(sign
(x)-1).x)) [ENTER] {}
exact(solve(a*(e^(x)+x)(sign
(x)-1)=0.x)) [ENTER]
e^x + x = 0 or x > 0 or a = 0
```

zeros({*expresión1*, *expresión2*}, {*varOAproximación1*, *varOAproximación2*, ...}) ⇒ *matriz*

Devuelve los posibles ceros reales del sistema de *expresiones* algebraicas, donde cada *varOAproximación* especifica una incógnita cuyo valor se busca.

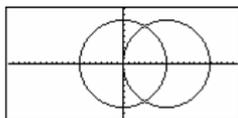
De forma opcional, puede especificar una aproximación inicial para una variable. Cada *varOAproximación* debe tener la forma:

variable
- 0 -
variable = número real o no real

Por ejemplo, *x* es válido, lo mismo que *x=3*.

Si todas las expresiones son polinómicas y NO se especifica ninguna aproximación inicial, zeros() utiliza el método de eliminación léxica de Gröbner/Buchberger para intentar determinar **todos** los ceros reales.

Por ejemplo, suponga que tiene una circunferencia de radio *r* centrada en el origen y otra circunferencia de radio *r* de centro el punto donde la primera circunferencia corta el eje positivo. Utilice **zeros()** para hallar las intersecciones.



Como se ve para *r* en el ejemplo de la derecha, las expresiones polinómicas simultáneas pueden tener variables extra que no contengan valores, pero representen valores numéricos dados que puedan sustituirse más adelante.

```
zeros({x^2+y^2- r^2,
(x- r)^2+y^2- r^2},{x,y,z}) [ENTER]
[ r  sqrt(3)*r ]
[ 2      2     ]
[ r -sqrt(3)*r ]
[ 2      2     ]
```

Cada fila de la matriz resultante representa un cero alternativo, con los componentes ordenados igual que en la lista de *varOAproximación*. Para extraer una fila, índice la matriz por [*fila*].

Extracción de la fila 2:
ans(1)[2] [ENTER] [r -sqrt(3)*r]
[2 2]

Además (o en su lugar) puede incluir incógnitas que no aparezcan en las expresiones. Por ejemplo, puede incluir *z* como una incógnita para ampliar el ejemplo anterior a dos cilindros intersectantes paralelos de radio *r* que se cortan. Los ceros para los cilindros muestran cómo las familias de ceros pueden contener constantes arbitrarias en la forma @*k*, donde *k* es un parámetro entero comprendido entre 1 y 255. El parámetro toma el valor 1 al utilizar **ClrHome** o **[F1] 8:Clear Home**.

```
zeros({x^2+y^2- r^2,
(x- r)^2+y^2- r^2},{x,y,z}) [ENTER]
[ r  sqrt(3)*r  @1 ]
[ 2      2     ]
[ r -sqrt(3)*r  @1 ]
[ 2      2     ]
```

Para sistemas de polinomios, el tiempo de cálculo y el consumo de la memoria dependen en gran medida del orden en que se listen las incógnitas. Si la opción inicial agota la memoria o su paciencia, intente reordenar las variables en las expresiones *y/o* en la lista de *varOAproximación*.

Si no se incluye ninguna aproximación y si ninguna expresión es no polinómica en cualquier variable pero todas las expresiones son lineales en las incógnitas, **zeros()** utiliza el método de eliminación gaussiana para intentar determinar todos los ceros reales.

Si un sistema no es polinómico en todas sus variables ni lineal en sus incógnitas, **zeros()** determina a lo sumo un cero mediante un método iterativo aproximado. Para ello, el número de incógnitas debe ser idéntico al número de expresiones, y todas las demás variables en las expresiones deben simplificarse a números.

Cada incógnita comienza en su valor aproximado, si es que existe; de no ser así, comienza en 0,0.

Utilice aproximaciones para obtener ceros adicionales uno a uno. Para que converja, es posible que una aproximación tenga que ser bastante cercana a una solución.

zeros({x+e^z(z)*y-1,x-y-sin(z)},
{x,y}) [ENTER]

$$\left[\frac{e^z \sin(z) + 1}{e^z + 1} \quad \frac{-(\sin(z) - 1)}{e^z + 1} \right]$$

zeros({e^z(z)*y-1,-y-sin(z)},
{y,z}) [ENTER]

[.041... 3.183...]

zeros({e^z(z)*y-1,-y-sin(z)},
{y,z=2π}) [ENTER]

[.001... 6.281...]

ZoomBox CATALOG

ZoomBox

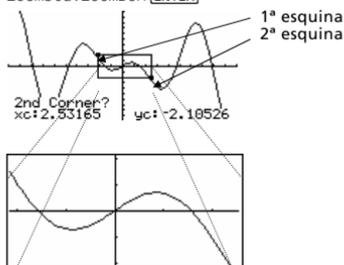
Presenta la pantalla Graph, permite dibujar el recuadro que define una nueva ventana de visualización, y actualiza la ventana.

En el modo de gráficas de función:

1.25x*cos(x)→y1(x) [ENTER]

Done

ZoomStd:ZoomBox [ENTER]



La pantalla después de definir **ZoomBox** pulsando [ENTER] la segunda vez.

ZoomData CATALOG

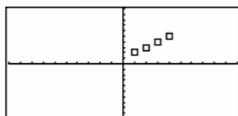
ZoomData

Ajusta los estados de la ventana de acuerdo con las gráficas (y datos) definidos, de forma que se incluyan todos los puntos correspondientes a datos estadísticos. También presenta la pantalla Graph.

Nota: No ajusta ymin e ymax para histogramas.

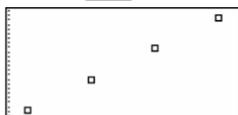
En el modo de gráficas de función:

```
{1.2.3.4} → L1 [ENTER]      {1 2 3 4}  
{2.3.4.5} → L2 [ENTER]      {2 3 4 5}  
newPlot 1.1.L1.L2 [ENTER]    Done  
ZoomStd [ENTER]
```



[HOME]
[CALC HOME]

ZoomData [ENTER]



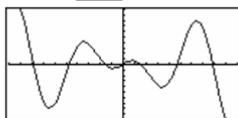
ZoomDec CATALOG

ZoomDec

Ajusta la ventana de visualización de manera que Δx y $\Delta y = 0.1$ muestren la pantalla Graph con el origen en el centro de la misma.

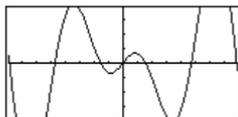
En el modo de gráficas de función:

```
1.25x*cos(x) → y1(x) [ENTER]    Done  
ZoomStd [ENTER]
```



[HOME]
[CALC HOME]

ZoomDec [ENTER]



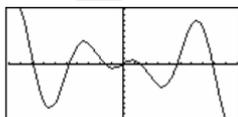
ZoomFit CATALOG

ZoomFit

Presenta la pantalla Graph y calcula el tamaño necesario de la ventana para las variables dependientes, con objeto de visualizar toda la imagen correspondiente a los valores actuales de la variable independiente.

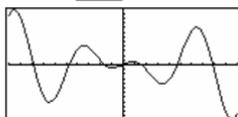
En el modo de gráficas de función:

```
1.25x*cos(x) → y1(x) [ENTER]    Done  
ZoomStd [ENTER]
```



[HOME]
[CALC HOME]

ZoomFit [ENTER]



ZoomIn CATALOG

ZoomIn

Presenta la pantalla Graph, permite establecer un punto central para efectuar un acercamiento y actualiza la ventana de visualización.

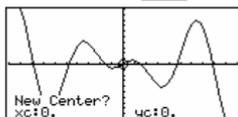
La magnitud del zoom depende de los factores Zoom, xFact e yFact. En el modo de representación gráfica en 3D, la magnitud dependerá de xFact, yFact y zFact.

En el modo de gráficas de función:

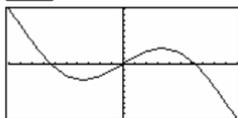
$1.25x * \cos(x) \rightarrow y1(x)$ **[ENTER]**

Done

ZoomStd:ZoomIn **[ENTER]**



[ENTER]



ZoomInt CATALOG

ZoomInt

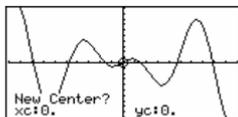
Presenta la pantalla Graph, permite establecer un punto central para el zoom y ajusta los estados de ventana para que cada pixel sea un número entero en todas las direcciones.

En el modo de gráficas de función:

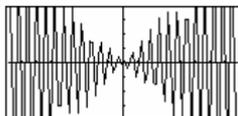
$1.25x * \cos(x) \rightarrow y1(x)$ **[ENTER]**

Done

ZoomStd:ZoomInt **[ENTER]**



[ENTER]



ZoomOut CATALOG

ZoomOut

Presenta la pantalla Graph, permite establecer un punto central para un alejamiento, y actualiza la ventana de visualización.

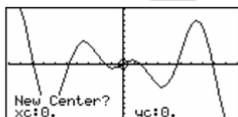
La magnitud del zoom depende de los factores Zoom, xFact e yFact. En el modo de representación gráfica en 3D, la magnitud dependerá de xFact, yFact y zFact.

En el modo de gráficas de función:

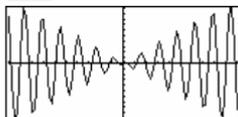
$1.25x * \cos(x) \rightarrow y1(x)$ **[ENTER]**

Done

ZoomStd:ZoomOut **[ENTER]**



[ENTER]



ZoomPrev CATALOG

ZoomPrev

Presenta la pantalla Graph y actualiza la ventana de visualización según la configuración existente antes del último zoom.

ZoomRcl CATALOG

ZoomRcl

Presenta la pantalla Graph y actualiza la ventana de visualización utilizando los estados que se hayan almacenado con la instrucción **ZoomSto**.

ZoomSqr CATALOG

ZoomSqr

Presenta la pantalla Graph, ajusta los estados de ventana x o y para que cada pixel tenga la misma anchura y altura en el sistema de coordenadas, y actualiza la ventana de visualización.

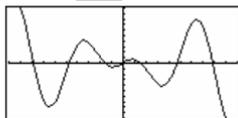
En el modo 3D Graph, **ZoomSqr** alarga los dos ejes más cortos para que tengan la misma longitud que el más largo.

En el modo de gráficas de función:

$1.25x * \cos(x) \rightarrow y1(x)$ **ENTER**

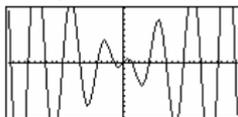
Done

ZoomStd **ENTER**



HOME

ZoomSqr **ENTER**



ZoomStd CATALOG

ZoomStd

Ajusta las variables de ventana en los siguientes valores estándar y después actualiza la ventana de visualización.

Gráficas de función:

x: [-10, 10, 1], y: [-10, 10, 1] y xres=2

Gráficas en paramétricas:

t: [0, 2 π , $\pi/24$], x: [-10, 10, 1], y: [-10, 10, 1]

Gráficas en polares:

θ : [0, 2 π , $\pi/24$], x: [-10, 10, 1], y: [-10, 10, 1]

Gráficas de sucesiones:

nmin=1, nmax=10, plotStrt=1, plotStep=1,
x: [-10, 10, 1], y: [-10, 10, 1]

Gráficas en 3D:

eye θ =20, eye ϕ =70, eye ψ =0

x: [-10, 10, 14], y: [-10, 10, 14],

z: [-10, 10], ncontour=5

Gráficas de ecuaciones diferenciales:

t: [0, 10, .1, 0], x: [-1, 10, 1], y: [-10, 10, 1],

ncurves=0, Estep=1, diftol=.001, fldres=20,

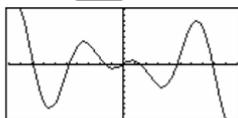
dtime=0

En el modo de gráficas de función:

$1.25x * \cos(x) \rightarrow y1(x)$ **ENTER**

Done

ZoomStd **ENTER**



ZoomSto CATALOG

ZoomSto

Almacena los estados de ventana actuales en la memoria de Zoom. Puede utilizarse **ZoomRcl** para restablecer dichos estados.

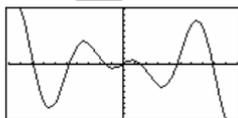
ZoomTrig CATALOG

ZoomTrig

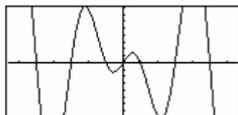
Presenta la pantalla Graph, ajusta Δx en $\pi/24$ y x_{scl} en $\pi/2$, centra el origen, ajusta los valores de y en $[-4, 4, .5]$ y actualiza la ventana de visualización.

En el modo de gráficas de función:

1.25x*cos(x) → y1(x) **[ENTER]** Done
ZoomStd **[ENTER]**



[HOME]
[CALC HOME]
ZoomTrig **[ENTER]**



+ (suma) Tecla **[+]**

$expresión1 + expresión2 \Rightarrow expresión$

Devuelve la suma de $expresión1$ y $expresión2$.

56 **[ENTER]** 56
ans(1)+4 **[ENTER]** 60
ans(1)+4 **[ENTER]** 64
ans(1)+4 **[ENTER]** 68
ans(1)+4 **[ENTER]** 72

$lista1 + lista2 \Rightarrow lista$

$matriz1 + matriz2 \Rightarrow matriz$

Devuelve una lista (o matriz) que contiene las sumas de los elementos correspondientes de $lista1$ y $lista2$ (o la $matriz1$ y la $matriz2$).

Los argumentos deben tener el mismo tamaño.

{22, π , $\pi/2$ } → L1 **[ENTER]** {22 π $\pi/2$ }
{10, 5, $\pi/2$ } → L2 **[ENTER]** {10 5 $\pi/2$ }
L1+L2 **[ENTER]** {32 $\pi+5 \pi$ }
ans(1)+{ π , -5, - π } **[ENTER]** { $\pi+32 \pi$ 0}

[a, b, c, d]+[1, 0, 0, 1] **[ENTER]**

$\begin{bmatrix} a+1 & b \\ c & d+1 \end{bmatrix}$

$expresión + lista1 \Rightarrow lista$

$lista1 + expresión \Rightarrow lista$

Devuelve una lista que contiene los resultados de las sumas de $expresión$ y cada elemento de la $lista1$.

15+{10, 15, 20} **[ENTER]** {25 30 35}
{10, 15, 20}+15 **[ENTER]** {25 30 35}

$expresión + matriz1 \Rightarrow matriz$

$matriz1 + expresión \Rightarrow matriz$

Devuelve una matriz con $expresión$ sumada a cada elemento de la diagonal de la $matriz1$. La $matriz1$ debe ser cuadrada.

Nota: Utilice **.+** (punto y signo de suma) para sumar una expresión a cada elemento.

20+[1, 2, 3, 4] **[ENTER]**

$\begin{bmatrix} 21 & 2 \\ 3 & 24 \end{bmatrix}$

– (resta) **Tecla** \square

$expresión1 - expresión2 \Rightarrow expresión$ $6 - 2$ [ENTER] 4
 Devuelve $expresión1$ menos $expresión2$. $\pi - \pi/6$ [ENTER] $\frac{5 \cdot \pi}{6}$

$lista1 - lista2 \Rightarrow lista$ $\{22, \pi, \pi/2\} - \{10.5, \pi/2\}$ [ENTER] $\{12 \pi - 5 0\}$
 $matriz1 - matriz2 \Rightarrow matriz$ $[3.4] - [1.2]$ [ENTER] $[2 2]$

Resta cada elemento de $lista2$ (o la $matriz2$) del correspondiente elemento de $lista1$ (o la $matriz1$) y devuelve los resultados.

El tamaño de los argumentos debe ser el mismo.

$expresión - lista1 \Rightarrow lista$ $15 - \{10.15, 20\}$ [ENTER] $\{5 0 -5\}$
 $lista1 - expresión \Rightarrow lista$ $\{10.15, 20\} - 15$ [ENTER] $\{-5 0 5\}$

Resta cada elemento de $lista1$ de la $expresión$ o resta la $expresión$ de cada elemento de $lista1$, después de lo cual devuelve una lista de los resultados.

$expresión - matriz1 \Rightarrow matriz$ $20 - [1.2; 3.4]$ [ENTER] $\begin{bmatrix} 19 & -2 \\ -3 & 16 \end{bmatrix}$
 $matriz1 - expresión \Rightarrow matriz$

En ambos casos devuelve la matriz cuya diagonal principal está constituida por $expresión$ - diagonal de la $matriz1$ o viceversa. $matriz1$ debe ser cuadrada.

Nota: Utilice $-$ (punto y signo de resta) para restar una expresión de cada elemento.

* (multiplic.) **Tecla** \times

$expresión1 * expresión2 \Rightarrow expresión$ $2 * 3.45$ [ENTER] 6.9
 Devuelve el producto de $expresión1$ por $expresión2$. $x * y * x$ [ENTER] $x^2 \cdot y$

$lista1 * lista2 \Rightarrow lista$ $\{1.0.2.3\} * \{4.5.6\}$ [ENTER] $\{4. 10 18\}$
 Devuelve una lista que contiene los productos de los elementos correspondientes de $lista1$ y $lista2$. $\{2/a. 3/2\} * \{a^2. b/3\}$ [ENTER] $\{2 \cdot a \frac{b}{2}\}$

El tamaño de las listas debe ser el mismo.

$matriz1 * matriz2 \Rightarrow matriz$ $[1.2.3; 4.5.6] * [a.d; b.e; c.f]$ [ENTER]
 Devuelve el producto matricial de $matriz1$ por $matriz2$.
 El número de filas de $matriz1$ debe ser igual al número de columnas de $matriz2$.

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \begin{bmatrix} a & d \\ b & e \\ c & f \end{bmatrix} = \begin{bmatrix} a+2 \cdot b+3 \cdot c & d+2 \cdot e+3 \cdot f \\ 4 \cdot a+5 \cdot b+6 \cdot c & 4 \cdot d+5 \cdot e+6 \cdot f \end{bmatrix}$$

$expresión * lista1 \Rightarrow lista$ $\pi * \{4.5.6\}$ [ENTER] $\{4 \cdot \pi 5 \cdot \pi 6 \cdot \pi\}$
 $lista1 * expresión \Rightarrow lista$

Devuelve una lista que contiene los productos de $expresión$ por cada elemento en la $lista1$.

$expresión * matriz1 \Rightarrow matriz$ $[1.2; 3.4] * .01$ [ENTER] $\begin{bmatrix} .01 & .02 \\ .03 & .04 \end{bmatrix}$
 $matriz1 * expresión \Rightarrow matriz$

Devuelve una matriz que contiene los productos de la $expresión$ y cada elemento en la $matriz1$.

$\lambda * \text{identity}(3)$ [ENTER] $\begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}$

Nota: Utilice $*$ (punto y signo de multiplicación) para multiplicar una expresión por cada elemento.

/ (división) Tecla \div

$expresión1 / expresión2 \Rightarrow expresión$ 2/3.45 **ENTER** .57971
 Devuelve el resultado de dividir $expresión1$ dividida entre $expresión2$. x^3/x **ENTER** x^2

$lista1 / lista2 \Rightarrow lista$ {1.0.2.3}/ {4.5.6} **ENTER** { .25 2/5 1/2}

Devuelve una lista que contiene los cocientes de la $lista1$ dividida entre la $lista2$.

El tamaño de las listas debe ser el mismo.

$expresión / lista1 \Rightarrow lista$ $a/\{3.a.\sqrt{a}\}$ **ENTER** $\left\{\frac{a}{3} \ 1 \ \sqrt{a}\right\}$
 $lista1 / expresión \Rightarrow lista$

Devuelve una lista que contiene los cocientes de $expresión$ dividida entre $lista1$, o de $lista1$ dividida entre $expresión$.

$\{a.b.c\}/(a*b*c)$ **ENTER** $\left\{\frac{1}{b \cdot c} \ \frac{1}{a \cdot c} \ \frac{1}{a \cdot b}\right\}$

$matriz1 / expresión \Rightarrow matriz$ $[a.b.c]/(a*b*c)$ **ENTER** $\left[\frac{1}{b \cdot c} \ \frac{1}{a \cdot c} \ \frac{1}{a \cdot b}\right]$

Devuelve una matriz que contiene los cocientes de la división $matriz1 / expresión$.

Nota: Utilice \cdot (punto y signo de división) para dividir una expresión entre cada elemento.

^ (potencia) Tecla \wedge

$expresión1 ^ expresión2 \Rightarrow expresión$ 4^2 **ENTER** 16
 $lista1 ^ lista2 \Rightarrow lista$ $\{a.2.c\}^{\{1.b.3\}}$ **ENTER** $\{a \ 2b \ c^3\}$

Devuelve el primer argumento elevado al segundo.

En una lista, devuelve los elementos de la $lista1$ elevados a los elementos correspondientes de la $lista2$.

En el dominio real, las potencias fraccionarias que tienen exponentes simplificados con denominadores impares utilizan la solución real, frente a la solución principal en el modo Complex.

$expresión ^ lista1 \Rightarrow lista$ $p^{\{a.2.-3\}}$ **ENTER** $\{p^a \ p^2 \ \frac{1}{p^3}\}$

Devuelve $expresión$ elevada a los elementos de la $lista1$.

$lista1 ^ expresión \Rightarrow lista$ $\{1.2.3.4\}^{-2}$ **ENTER** $\{1 \ 1/4 \ 1/9 \ 1/16\}$

Devuelve los elementos de $lista1$ elevados a $expresión$.

$Matriz \ cuadrada1 ^ entero \Rightarrow matriz$ $[1.2;3.4]^2$ **ENTER** $\begin{bmatrix} 7 & 10 \\ 15 & 22 \end{bmatrix}$
 $[1.2;3.4]^{-1}$ **ENTER** $\begin{bmatrix} -2 & 1 \\ 3/2 & -1/2 \end{bmatrix}$
 $[1.2;3.4]^{-2}$ **ENTER** $\begin{bmatrix} 11/2 & -5/2 \\ -15/4 & 7/4 \end{bmatrix}$

Devuelve $Matriz \ cuadrada1$ elevada al número entero.

La $Matriz \ cuadrada1$ debe ser una matriz cuadrada.

Si el $entero = -1$, calcula la matriz inversa.

Si el $entero < -1$, calcula la matriz inversa de la correspondiente potencia positiva.

$$\begin{matrix} \blacksquare \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^2 & \begin{bmatrix} 7 & 10 \\ 15 & 22 \end{bmatrix} \\ \blacksquare \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1} & \begin{bmatrix} -2 & 1 \\ 3/2 & -1/2 \end{bmatrix} \\ \blacksquare \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-2} & \begin{bmatrix} 11/2 & -5/2 \\ -15/4 & 7/4 \end{bmatrix} \end{matrix}$$

.+ (pto., suma) Teclas $\boxed{+}$ $\boxed{+}$

$matriz1 + matriz2 \Rightarrow matriz$
 $expresión + matriz1 \Rightarrow matriz$

$matriz1 + matriz2$ devuelve una matriz que es la suma de cada par de elementos correspondientes de $matriz1$ y $matriz2$.

$expresión + matriz1$ devuelve una matriz que es la suma de la $expresión$ y cada elemento de $matriz1$.

$[a, 2; b, 3] + [c, 4; 5, d]$ \boxed{ENTER}
 $x + [c, 4; 5, d]$ \boxed{ENTER}

$\begin{bmatrix} b & 3 \\ c & 4 \\ 5 & d \end{bmatrix} + \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} a+c & 6 \\ b+5 & d+3 \\ x+c & x+4 \\ x+5 & x+d \end{bmatrix}$
--	--

.- (pto., resta) Teclas $\boxed{-}$ $\boxed{-}$

$matriz1 - matriz2 \Rightarrow matriz$
 $expresión - matriz1 \Rightarrow matriz$

$matriz1 - matriz2$ devuelve una matriz que es la diferencia entre cada par de elementos correspondientes $matriz1$ y $matriz2$.

$expresión - matriz1$ devuelve una matriz que es la diferencia entre $expresión$ y cada elemento de $matriz1$.

$[a, 2; b, 3] - [c, 4; 5, d]$ \boxed{ENTER}
 $x - [c, 4; 5, d]$ \boxed{ENTER}

$\begin{bmatrix} b & 3 \\ c & 4 \\ 5 & d \end{bmatrix} - \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} a-c & -2 \\ b-d & -2 \\ x-c & x-4 \\ x-d & x-5 \end{bmatrix}$
--	--

.* (pto., mult.) Teclas $\boxed{\cdot}$ $\boxed{\times}$

$matriz1 * matriz2 \Rightarrow matriz$
 $expresión * matriz1 \Rightarrow matriz$

$matriz1 * matriz2$ devuelve una matriz que es el producto de cada par de elementos correspondientes de $matriz1$ y $matriz2$.

$expresión * matriz1$ devuelve una matriz que contiene los productos de $expresión$ por cada elemento de $matriz1$.

$[a, 2; b, 3] * [c, 4; 5, d]$ \boxed{ENTER}
 $x * [a, b; c, d]$ \boxed{ENTER}

$\begin{bmatrix} b & 3 \\ c & 4 \\ 5 & d \end{bmatrix} * \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} a \cdot c & 8 \\ 5 \cdot b & 3 \cdot d \\ a \cdot x & b \cdot x \\ c \cdot x & d \cdot x \end{bmatrix}$
--	--

./ (pto., div.) Teclas $\boxed{\div}$ $\boxed{\div}$

$matriz1 ./ matriz2 \Rightarrow matriz$
 $expresión ./ matriz1 \Rightarrow matriz$

$matriz1 ./ matriz2$ devuelve una matriz que es el cociente de cada par de elementos correspondientes de $matriz1$ y $matriz2$.

$expresión ./ matriz1$ devuelve una matriz que es el cociente de $expresión$ y cada elemento de $matriz1$.

$[a, 2; b, 3] ./ [c, 4; 5, d]$ \boxed{ENTER}
 $x ./ [c, 4; 5, d]$ \boxed{ENTER}

$\begin{bmatrix} b & 3 \\ c & 4 \\ 5 & d \end{bmatrix} ./ \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} \frac{b}{c} & \frac{3}{4} \\ \frac{x}{c} & \frac{x}{4} \\ \frac{x}{5} & \frac{x}{d} \end{bmatrix}$
---	---

.^ (pto., pot.) Teclas $\boxed{\wedge}$ $\boxed{\wedge}$

$matriz1 ^ matriz2 \Rightarrow matriz$
 $expresión ^ matriz1 \Rightarrow matriz$

$matriz1 ^ matriz2$ devuelve una matriz en la que cada elemento de $matriz2$ es el exponente del correspondiente elemento de $matriz1$.

$expresión ^ matriz1$ devuelve una matriz en que cada elemento de $matriz1$ es el exponente de $expresión$.

$[a, 2; b, 3] ^ [c, 4; 5, d]$ \boxed{ENTER}
 $x ^ [c, 4; 5, d]$ \boxed{ENTER}

$\begin{bmatrix} a & 2 \\ b & 3 \\ c & 4 \\ 5 & d \end{bmatrix} ^ \begin{bmatrix} c & 4 \\ 5 & d \end{bmatrix}$	$\begin{bmatrix} a^c & 16 \\ b^5 & 3^d \\ x^c & x^4 \\ x^5 & x^d \end{bmatrix}$
---	---

- (negativo) Tecla $\boxed{-}$ y menú MATH/Base

- *expresión1* \Rightarrow *expresión*
 - *lista1* \Rightarrow *lista*
 - *matriz1* \Rightarrow *matriz*

Devuelve el opuesto del argumento.

En una lista o una matriz, devuelve el opuesto de cada elemento.

Si *expresión1* es un número entero binario o hexadecimal, el opuesto da el complemento a dos de ambos.

-2.43 $\boxed{\text{ENTER}}$ -2.43
 -{-1.0,4.1,2E19} $\boxed{\text{ENTER}}$
 {1 - .4 -1.2E19}
 -a * -b $\boxed{\text{ENTER}}$ a * b

En el modo de base Bin:

0b100101 \blacktriangleright dec $\boxed{\text{ENTER}}$ 37

Importante: Cero, no la letra O.

-0b100101 $\boxed{\text{ENTER}}$
 0b11111111111111111111111111111111011011

ans(1) \blacktriangleright dec $\boxed{\text{ENTER}}$ -37

Nota: Para escribir \blacktriangleright , pulse $\boxed{2\text{nd}} \boxed{\blacktriangleright}$.

% (porcent.) Menú CHAR/Punctuation

expresión1 % \Rightarrow *expresión*
lista1 % \Rightarrow *lista*
matriz1 % \Rightarrow *matriz*

Devuelve $\frac{\textit{argument}}{100}$.

En una lista o una matriz, devuelve una lista o matriz con cada elemento dividido entre 100.

13% $\boxed{\blacktriangledown} \boxed{\text{ENTER}}$.13
 {1, 10, 100}% $\blacktriangledown \boxed{\text{ENTER}}$ {.01 .1 1.}

= (igual) Tecla $\boxed{=}$

expresión1 = *expresión2* \Rightarrow *expresión booleana*
lista1 = *lista2* \Rightarrow *lista booleana*
matriz1 = *matriz2* \Rightarrow *matriz booleana*

Devuelve true si se determina que la *expresión1* es igual a la *expresión2*.

Devuelve false si se determina que la *expresión1* no es igual a la *expresión2*.

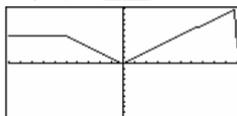
En todos los demás casos devuelve la ecuación simplificada.

En listas o matrices, devuelve comparaciones elemento por elemento.

Ejemplo de lista de función utilizando símbolos matemáticos: =, \neq , <, \leq , >, \geq

```
:g(x)
:Func
:If x $\leq$ -5 Then
: Return 5
: ElseIf x>-5 and x<0 Then
: Return -x
: ElseIf x $\geq$ 0 and x $\neq$ 10 Then
: Return x
: ElseIf x=10 Then
: Return 3
: EndIf
: EndFunc
```

Graph g(x) $\boxed{\text{ENTER}}$



≠ Tecla [≠]

expresión1 != *expresión2* ⇒ *expresión booleana*
lista1 != *lista2* ⇒ *lista booleana*
matriz1 != *matriz2* ⇒ *matriz booleana*

Consulte el ejemplo de "=" (igual)".

Devuelve true si se determina que *expresión1* es distinta a *expresión2*.

Devuelve false si se determina que *expresión1* es igual a *expresión2*.

En todos los demás casos devuelve la ecuación simplificada.

En listas o matrices, devuelve comparaciones elemento por elemento.

< Tecla [2nd] [<]

expresión1 < *expresión2* ⇒ *expresión booleana*
lista1 < *lista2* ⇒ *lista booleana*
matriz1 < *matriz2* ⇒ *matriz booleana*

Consulte el ejemplo de "=" (igual)".

Devuelve true si se determina que *expresión1* es menor que *expresión2*.

Devuelve false si se determina que la *expresión1* es mayor o igual que la *expresión2*.

En todos los demás casos devuelve la ecuación simplificada.

En el caso de listas y matrices, devuelve comparaciones elemento por elemento.

≤ Teclas [◀] [0]

expresión1 <= *expresión2* ⇒ *expresión booleana*
lista1 <= *lista2* ⇒ *lista booleana*
matriz1 <= *matriz2* ⇒ *matriz booleana*

Consulte el ejemplo de "=" (igual)".

Devuelve true si se determina que *expresión1* es menor o igual que *expresión2*.

Devuelve false si se determina que *expresión1* es mayor que *expresión2*.

En todos los demás casos devuelve la ecuación simplificada.

En listas y matrices, devuelve comparaciones elemento por elemento.

> Tecla [2nd] [>]

expresión1 > *expresión2* ⇒ *expresión booleana*
lista1 > *lista2* ⇒ *lista booleana*
matriz1 > *matriz2* ⇒ *matriz booleana*

Consulte el ejemplo de "=" (igual)".

Devuelve true si se determina que *expresión1* es mayor que *expresión2*.

Devuelve false si se determina que *expresión1* es menor o igual que *expresión2*.

En todos los demás casos devuelve la ecuación simplificada.

En listas y matrices, devuelve comparaciones elemento por elemento.

Teclas

$expresión1 \geq expresión2 \Rightarrow expresión\ booleana$
 $lista1 \geq lista2 \Rightarrow lista\ booleana$
 $matriz1 \geq matriz2 \Rightarrow matriz\ booleana$

Consulte el ejemplo de "=" (igual)".

Devuelve true si se determina que *expresión1* es mayor o igual que *expresión2*.

Devuelve false si se determina que *expresión1* es menor que *expresión2*.

En todos los demás casos devuelve la ecuación simplificada.

En listas y matrices devuelve comparaciones elemento por elemento.

! (factorial) **Tecla** **Tecla** 2^{nd} W

$expresión1! \Rightarrow expresión$
 $lista1! \Rightarrow lista$
 $matriz1! \Rightarrow matriz$

5! \boxed{ENTER} 120
 {5,4,3}! \boxed{ENTER} {120 24 6}
 [1,2,3,4]! \boxed{ENTER} $\begin{bmatrix} 1 & 2 \\ 6 & 24 \end{bmatrix}$

Devuelve el factorial del argumento.

En una lista o matriz, devuelve una lista o matriz de factoriales de los elementos.

La TI-89 calcula el factorial sólo para los números enteros no negativos.

& (anex.) **Tecla** **Tecla** 2^{nd} H

$cadena1 \& cadena2 \Rightarrow cadena$

"Hello " & "Nick" \boxed{ENTER} "Hello Nick"

Devuelve la cadena de texto formada por la *cadena2* anexada a la *cadena1*.

∫() (integrar) **Tecla** 2^{nd} \int

$\int (expresión1, var, inferior [, superior]) \Rightarrow expresión$
 $\int (lista1, var, orden) \Rightarrow lista$
 $\int (matriz1, var, orden) \Rightarrow matriz$

Devuelve la integral de la *expresión1* calculada respecto a la variable *var* desde el valor *inferior* hasta el *superior*.

$\int (x^2 \cdot x \cdot a \cdot b) \boxed{ENTER}$ $-\frac{a^3}{3} + \frac{b^3}{3}$

Devuelve la función primitiva si se omiten los valores *inferior* y *superior*. Omite las constantes simbólicas de integración como C.

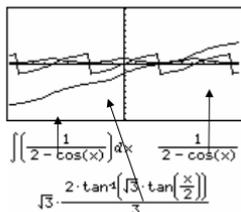
$\int (x^2 \cdot x) \boxed{ENTER}$ $\frac{x^3}{3}$

Sin embargo, añade el valor *inferior* como constante de integración si se omite únicamente el *superior*.

$\int (a \cdot x^2 \cdot x \cdot c) \boxed{ENTER}$ $\frac{a \cdot x^3}{3} + c$

Las funciones primitivas válidas pueden diferenciarse por una constante numérica. Dicha constante puede estar oculta, especialmente cuando una primitiva contiene logaritmos o funciones trigonométricas inversas. Además, a veces pueden añadirse expresiones constantes por intervalos para hacer que una primitiva sea válida en un intervalo más amplio de lo normal.

$f(1/(2-\cos(x)),x) \rightarrow \text{tmp}(x)$ [ENTER]
 ClrGraph:Graph tmp(x):Graph
 $1/(2-\cos(x))$:Graph $\sqrt{3}$
 $(2\tan^{-1}(\sqrt{3}(\tan(x/2))))/3$ [ENTER]



$\int()$ se calcula por partes permaneciendo el símbolo de integral para aquellas funciones que no sea capaz de resolver.

Cuando están presentes los valores *inferior* y *superior*, se intenta localizar cualquier discontinuidad o derivadas discontinuas en el intervalo *inferior* < *var* < *superior* y subdividir el intervalo en dichos lugares.

En el estado AUTO del modo Exact/Approx, se utiliza la integración numérica cuando no puede determinarse una primitiva o límite.

En el estado APPROX, se intenta utilizar primero la integración numérica, si da lugar. Las primitivas se intentan hallar sólo cuando no puede utilizarse o falla la integración numérica.

$\int()$ se puede anidar para calcular integrales múltiples. Los límites de integración pueden depender de las variables de integración fuera de ellos.

Nota: Consulte además **nint()**.

$f(b * e^{-(x^2)} + a / (x^2 + a^2), x)$ [ENTER]

$$\int \left[b \cdot e^{-x^2} + \frac{a}{x^2 + a^2} \right] dx$$

$$b \cdot \int \left[e^{-x^2} \right] dx + \tan^{-1} \left(\frac{x}{a} \right)$$

$f(e^{-(x^2)}, x, -1.1)$ [ENTER] 1.493...

$f(\int(\ln(x+y), y, 0, x), x, 0, a)$ [ENTER]

$$\int_a^x \int_0^x \ln(x+y) dy dx$$

$$\frac{a^2 \cdot \ln(a)}{2} + a^2 \cdot (\ln(2) - 3/4)$$

$\sqrt{()}$ (raíz cuad.) Tecla [2nd] [√]

$\sqrt{(\text{expresión})} \Rightarrow \text{expresión}$

$\sqrt{(\text{lista})} \Rightarrow \text{lista}$

Devuelve la raíz cuadrada del argumento.

En una lista, devuelve las raíces cuadradas de todos los elementos de *lista*.

$\sqrt{4}$ [ENTER] 2

$\sqrt{\{9, a, 4\}}$ [ENTER] {3 √a 2}

$\Pi()$ (prod.) Menú MATH/Calculus

$\Pi(\text{expresión1}, \text{var}, \text{inferior}, \text{superior}) \Rightarrow \text{expresión}$

Calcula *expresión1* para cada valor de *var* entre los valores *inferior* y *superior*, y devuelve el producto de los resultados.

$\Pi(\text{expresión1}, \text{var}, \text{inferior}, \text{inferior}-1) \Rightarrow 1$

$\Pi(\text{expresión1}, \text{var}, \text{inferior}, \text{superior}) \Rightarrow 1/\Pi(\text{expresión1}, \text{var}, \text{superior}-1, \text{inferior}-1)$
 si *superior* < *inferior* - 1

$\Pi(1/n, n, 1.5)$ [ENTER] $\frac{1}{120}$

$\Pi(k^2, k, 1, n)$ [ENTER] $(n!)^2$

$\Pi(\{1/n, n, 2\}, n, 1.5)$ [ENTER] $\left\{ \frac{1}{120}, 120, 32 \right\}$

$\Pi(k, k, 4, 3)$ [ENTER] 1

$\Pi(1/k, k, 4, 1)$ [ENTER] 6

$\Pi(1/k, k, 4, 1) * \Pi(1/k, k, 2, 4)$ [ENTER] 1/4

Σ (suma) Menú MATH/Calculus

$\Sigma(\text{expresión1}, \text{var}, \text{inferior}, \text{superior}) \Rightarrow \text{expresión}$

Calcula *expresión1* para cada valor de *var* entre los valores *inferior* y *superior*, y devuelve la suma de los resultados.

$$\Sigma(1/n.n, 1.5) \text{ [ENTER]} \quad \frac{137}{60}$$

$$\Sigma(k^2.k, 1.n) \text{ [ENTER]} \quad \frac{n \cdot (n+1) \cdot (2 \cdot n+1)}{6}$$

$$\Sigma(1/n^2.n, 1.\infty) \text{ [ENTER]} \quad \frac{\pi^2}{6}$$

$\Sigma(\text{expresión1}, \text{var}, \text{inferior}, \text{inferior}-1) \Rightarrow 0$

$$\Sigma(k.k, 4.3) \text{ [ENTER]} \quad 0$$

$\Sigma(\text{expresión1}, \text{var}, \text{inferior}, \text{superior}) \Rightarrow$
 $-\Sigma(\text{expresión1}, \text{var}, \text{superior}-1, \text{inferior}-1)$ si *superior* < *inferior* - 1

$$\Sigma(k.k, 4.1) \text{ [ENTER]} \quad -5$$

$$\Sigma(k.k, 4.1) + \Sigma(k.k, 2.4) \text{ [ENTER]} \quad 4$$

(dir. indirect.) CATALOG

varNombreCadena

Llama a la variable cuyo nombre es *varNombreCadena*. Permite crear y modificar variables desde un programa utilizando cadenas.

Parte de un programa:

```

:
:
:Request "Enter Your Name".str1
:NewFold #str1
:
:
:
:For i,1.5.1
:ClrGraph
:Graph i*x
:StoPic #("pic" & string(i))
:EndFor
:
:

```

° (gadian) Menú MATH/Angle

$\text{expresión1}^\circ \Rightarrow \text{expresión}$

$\text{lista1}^\circ \Rightarrow \text{lista}$

$\text{matriz1}^\circ \Rightarrow \text{matriz}$

Esta función es un medio para utilizar un ángulo en grados centesimales cuando el modo está definido en grados o radianes.

En el modo de ángulo en radianes, multiplica la *expresión1* por $\pi/200$. En el modo de ángulo en grados, multiplica la *expresión1* por $g/100$. En el modo de ángulo en grados centesimales, devuelve la *expresión1* sin cambios.

En el modo Angle en grados, grados centesimales o radianes:

$$\cos(50^\circ) \text{ [ENTER]} \quad \frac{\sqrt{2}}{2}$$

$$\cos(\{0, 100^\circ, 200^\circ\}) \text{ [ENTER]} \quad \{1.0, -1\}$$

ʳ (radianes) Menú MATH/Angle

$\text{expresión1}^r \Rightarrow \text{expresión}$

$\text{lista1}^r \Rightarrow \text{lista}$

$\text{matriz1}^r \Rightarrow \text{matriz}$

En el modo de ángulo en grados, multiplica la *expresión1* por $180/\pi$. En el modo de ángulo en radianes, devuelve la *expresión1* sin cambios. En el modo de ángulo en grados centesimales, multiplica la *expresión1* por $200/\pi$.

Esta función es un medio para utilizar un ángulo en radianes cuando el modo está definido en grados o grados centesimales.

Sugerencia: Utilice r si quiere forzar al uso de radianes en una definición de programa o función, con independencia del modo que prevalezca al utilizar el programa o función.

En el modo Angle en grados, grados centesimales o radianes:

$$\cos((\pi/4)^r) \text{ [ENTER]} \quad \frac{\sqrt{2}}{2}$$

$$\cos(\{0^r, (\pi/12)^r, -\pi^r\}) \text{ [ENTER]} \quad \left\{1, \frac{(\sqrt{3}+1) \cdot \sqrt{2}}{4}, -1\right\}$$

° (grados) Tecla [2nd] [°]

expresión[°] ⇒ *valor*
lista^{1°} ⇒ *lista*
matriz^{1°} ⇒ *matriz*

En el modo de ángulo en radianes, multiplica la expresión por $\pi/180$. En el modo de ángulo en grados, devuelve la expresión sin cambios. En el modo de ángulo en grados centesimales, multiplica la expresión *t* por 10/9.

Esta función es un medio para utilizar un ángulo en grados cuando el modo está definido en grados centesimales o radianes.

En el modo Angle en grados, grados centesimales o radianes:

$$\cos(45^\circ) \text{ [ENTER]} \quad \frac{\sqrt{2}}{2}$$

$$\cos(\{0.\pi/4.90^\circ.30.12^\circ\}) \text{ [ENTER]} \quad \{1.707\dots 0.864\dots\}$$

∠ (ángulo) Tecla [2nd] [∠]

[*radio*,∠*ángulo*] ⇒ *vector* (entrada de polar)
 [*radio*,∠*ángulo*,*Z* *coordinada*] ⇒ *vector*
 (entrada de cilíndrico)
 [*radio*,∠*ángulo*,∠*φ* *ángulo*] ⇒ *vector*
 (entrada de esférico)

Devuelve las coordenadas como un vector dependiendo del estado del modo Vector Format: rectangular, cilíndrico o esférico.

$$[5, \angle 60^\circ, \angle 45^\circ] \text{ [ENTER]}$$

En el modo en radianes y el formato de vector establecido en:

$\begin{bmatrix} 5 \angle 60^\circ \angle 45^\circ \\ \frac{5\sqrt{2}}{4} \frac{5\sqrt{6}}{4} \frac{5\sqrt{2}}{2} \end{bmatrix}$	rectangular
$\begin{bmatrix} 5 \angle 60^\circ \angle 45^\circ \\ \frac{5\sqrt{2}}{2} \angle \frac{\pi}{3} \frac{5\sqrt{2}}{2} \end{bmatrix}$	cilíndrico
$\begin{bmatrix} 5 \angle 60^\circ \angle 45^\circ \\ 5 \angle \frac{\pi}{3} \angle \frac{\pi}{4} \end{bmatrix}$	esférico

(*magnitud* ∠ *ángulo*) ⇒ *Valor complejo* (entrada en polar)

Introduce un valor complejo en forma polar ($r\angle\theta$). El *ángulo* se interpreta de acuerdo con estado actual del modo Angle.

En el modo Angle en radianes y en el modo de formato complejo rectangular:

$$5+3i - (10\angle\pi/4) \text{ [ENTER]} \quad 5 - 5\sqrt{2} + (3 - 5\sqrt{2}) \cdot i$$

$$\text{ [ENTER]} \quad -2.071\dots - 4.071\dots i$$

°, ', " Tecla [2nd] [°] (°), tecla [2nd] ['] ('), tecla [2nd] ["] (")

gg°mm'ss.ss" ⇒ *expresión*

gg Un número positivo o negativo
mm Un número no negativo
ss.ss Un número no negativo

Devuelve $gg+(mm/60)+(ss.ss/3600)$.

Este formato de entrada en base 60 permite lo siguiente:

- Introducir un ángulo en grados/minutos/segundos sin tomar en cuenta el estado actual del modo Angle.
- Introducir la hora como horas/minutos/segundos.

En el modo Angle, en grados:

$$25^\circ 13' 17.5'' \text{ [ENTER]} \quad 25.221\dots$$

$$25^\circ 30' \text{ [ENTER]} \quad 51/2$$

' (primo) Tecla [2nd] [']

variable'
variable''

Introduce un símbolo de "prima" en una ecuación diferencial. Un solo símbolo de "prima" indica una ecuación diferencial de primer orden, dos indican una de segundo orden, etc.

deSolve($y''=y^{(-1/2)}$ and $y(0)=0$ and $y'(0)=0$,t,y) [ENTER]

$$\frac{2 \cdot y^{3/4}}{3} = t$$

(subrayado)

Tecla []

Tecla [2nd] []

expresión_unidad

Designa las unidades para una *expresión*. Todos los nombres de unidad comienzan por un guión de subrayado.

Puede utilizar unidades predefinidas o crear sus propias unidades. Para obtener una lista de unidades predefinidas, consulte el capítulo sobre constantes y unidades de medida. Puede pulsar:

 [2nd] [UNITS]

 [] [UNITS]

para seleccionar unidades en un menú, o bien puede escribir los nombres de unidad directamente.

3_m▶_ft [ENTER]

9.842..._ft

Nota: Para escribir ▶, pulse [2nd] [▶].

variable_

Cuando *variable* no tiene asignado un valor, considera que representa un número complejo. Por omisión, sin el , la variable se trata como real.

Si la *variable* tiene un valor asignado, el se ignora y la *variable* retiene su tipo de datos original.

Nota: Puede almacenar un número complejo en una variable utilizando . No obstante, para obtener los mejores resultados en cálculos tales como **cSolve()** y **cZeros()**, se recomienda utilizar el .

Partiendo del supuesto de que *z* no está definida:

real(<i>z</i>) [ENTER]	<i>z</i>
real(<i>z_</i>) [ENTER]	real(<i>z_</i>)
imag(<i>z</i>) [ENTER]	0
imag(<i>z_</i>) [ENTER]	imag(<i>z_</i>)

▶ (conversión)

Tecla [2nd] [▶]

expresión_unidad1 ▶ *unidad2* ⇒ *expresión_unidad2*

3_m▶_ft [ENTER]

9.842..._ft

Convierte una expresión de una unidad a otra. Las unidades deben pertenecer a la misma categoría.

El carácter de subrayado designa las unidades. Para obtener una lista de unidades predefinidas válidas, consulte el capítulo sobre constantes y unidades de medida. Puede pulsar:

 [2nd] [UNITS]

 [] [UNITS] para seleccionar unidades en un menú, o bien puede escribir los nombres de unidad directamente.

Para obtener el guión bajo al escribir unidades directamente, pulse:

 [] []

 [2nd] []

Nota: El operador de conversión ▶ no maneja unidades de temperatura. Como alternativa, utilice **tmpCnv()** y **ΔtmpCnv()**.

10^()

CATALOG

10^ (*expresión*) ⇒ *expresión*

10^(1.5) [ENTER]

31.622...

10^ (*lista*) ⇒ *lista*

10^{0, -2, 2, a} [ENTER]

Devuelve 10 elevado al argumento.

{1 $\frac{1}{100}$ 100 10^a}

En una lista, devuelve 10 elevado a los elementos de la *lista*1.

$10^{\wedge}(\text{Matriz cuadrada1}) \Rightarrow \text{Matriz cuadrada}$

Devuelve 10 elevado a la potencia de *Matriz cuadrada1*. Esto *no* es lo mismo que calcular 10 elevado a la potencia de cada elemento. Para más información sobre el método de cálculo, consulte **cos()**.

La *Matriz cuadrada1* debe ser diagonalizable. El resultado siempre contiene números en coma flotante.

$10^{\wedge}([1,5,3;4,2,1;6,-2,1])$ [ENTER]

$\begin{bmatrix} 1.143\dots\text{E}7 & 8.171\dots\text{E}6 & 6.675\dots\text{E}6 \\ 9.956\dots\text{E}6 & 7.115\dots\text{E}6 & 5.813\dots\text{E}6 \\ 7.652\dots\text{E}6 & 5.469\dots\text{E}6 & 4.468\dots\text{E}6 \end{bmatrix}$

x^{-1} CATALOG (\wedge^{-1})

expresión1 $x^{-1} \Rightarrow$ *expresión*

lista1 $x^{-1} \Rightarrow$ *lista*

Devuelve el inverso del argumento.

En una lista, devuelve el inverso de los elementos de la *lista1*.

$3.1^{\wedge-1}$ [ENTER] .322581

$\{a, 4, -1, x, -2\}^{\wedge-1}$ [ENTER]

$\left\{\frac{1}{a} \quad \frac{1}{4} \quad -10 \quad \frac{1}{x-2}\right\}$

Matriz cuadrada1 $x^{-1} \Rightarrow$ *Matriz cuadrada*

Devuelve la inversa de *Matriz cuadrada1*.

La *Matriz cuadrada1* debe ser una matriz cuadrada no singular.

$[1, 2; 3, 4]^{\wedge-1}$ [ENTER]

$[1, 2; a, 4]^{\wedge-1}$ [ENTER]

$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}^{-1} \quad \begin{bmatrix} -2 & 1 \\ 3/2 & -1/2 \end{bmatrix}$
 $\begin{bmatrix} 1 & 2 \\ a & 4 \end{bmatrix}^{-1} \quad \begin{bmatrix} -2 & 1 \\ \frac{a}{2(a-2)} & \frac{1}{2(a-2)} \end{bmatrix}$

("with") Tecla [1] Tecla [2nd] [1]

expresión | *expresión booleana1* | y | *expresión booleana2* | ... | y | *expresión booleanaM*

El símbolo (|) "with" sirve de operador binario. El operando a la izquierda de | es una expresión. El operando a la derecha de | especifica una o más relaciones que deben influir en la simplificación de la expresión. Si hay varias relaciones después del símbolo |, deben estar unidas por "and" lógico.

El operador "with" proporciona tres tipos básicos de funciones: sustituciones, restricciones de intervalos y exclusiones.

Las sustituciones son en la forma de una igualdad, como $x=3$ o $y=\sin(x)$. Para resultar más útiles, el lado izquierdo debe ser una variable única. *expresión* | *variable* = *valor* sustituye el *valor* en cada ocurrencia de la *variable* en la *expresión*.

Las condiciones del intervalo adoptan la forma de una o más desigualdades unidas por operadores "and" lógicos. Las condiciones de intervalos también permiten la simplificación, que de otra manera sería no válida o no calculable.

Las exclusiones utilizan los operadores relacionales "distinto de" (\neq o \neq), para excluir un valor específico. Se emplean principalmente para excluir una solución exacta cuando se utiliza **cSolve()**, **cZeros()**, **fMax()**, **fMin()**, **solve()**, **zeros()**, etc.

$x+1$ | $x=3$ [ENTER] 4

$x+y$ | $x=\sin(y)$ [ENTER] $\sin(y)+y$

$x+y$ | $\sin(y)=x$ [ENTER] $x+y$

$x^{\wedge}3-2x+7 \Rightarrow f(x)$ [ENTER] Done

$f(x)$ | $x=\sqrt{(3)}$ [ENTER] $\sqrt{3}+7$

$(\sin(x))^{\wedge}2+2\sin(x)-6$ | $\sin(x)=d$ [ENTER] d^2+2d-6

$\text{solve}(x^{\wedge}2-1=0, x) | x>0$ and $x<2$ [ENTER] $x=1$

$\sqrt{(x)} * \sqrt{(1/x)} | x>0$ [ENTER] 1

$\sqrt{(x)} * \sqrt{(1/x)}$ [ENTER] $\sqrt{\frac{1}{x}} * \sqrt{x}$

$\text{solve}(x^{\wedge}2-1=0, x) | x \neq 1$ [ENTER] $x=-1$

→ (almac.) **Tecla** 

expresión → var
lista → var
matriz → var
expresión → fun_nombre(parámetro1,...)
lista → fun_nombre(parámetro1,...)
matriz → fun_nombre(parámetro1,...)

$\pi/4$ → myvar  $\frac{\pi}{4}$
 2cos(x) → Y1(x)  Done
 {1,2,3,4} → Lst5  {1 2 3 4}
 [1,2,3;4,5,6] → MatG  $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$
 "Hello" → str1  "Hello"

Si no existe la variable *var*, crea *var* y la inicia con *expresión*, *lista* o *matriz*.

Si *var* ya existe y no está bloqueada o protegida, sustituye su contenido con *expresión*, *lista* o *matriz*.

Sugerencia: Si va a realizar cálculos simbólicos con variables no definidas, evite almacenar elementos en las variables de una letra utilizadas habitualmente, como a, b, c, x, y, z, etc.

● (coment.) **Menú Program Editor/Control** o

 **Tecla**  
 **Tecla**  X

● [*texto*]

● procesa el *texto* como una línea de comentario que puede utilizarse para anotar instrucciones de un programa.

● puede estar al principio o en cualquier parte de la línea. Todo lo que esté entre la derecha de ● y el final de la línea es el comentario.

Parte de un programa:

```

:
:
: ● Get 10 points from the Graph
: screen
: For i,1,10 ● This loops 10 times
:
:

```

Ob, Oh

 **Teclas**  alpha   **Teclas**  B
 **Teclas**  alpha   **Teclas**  H

Ob Número binario

Oh Número hexadecimal

Indica un número binario o hexadecimal, respectivamente. Para introducir un número binario o hexadecimal, debe introducirse el prefijo Ob ó Oh independientemente del estado del modo Base. Sin un prefijo, un número se trata como decimal (base 10).

Los resultados se presentan de acuerdo con el estado del modo Base.

En el modo de base Dec:

Ob10+0hF+10  27

En el modo de base Bin:

Ob10+0hF+10  Ob11011

En el modo de base Hex:

Ob10+0hF+10  Oh1B

Apéndice B: Información general

Información sobre productos, servicios y garantías de TI

Información sobre productos y servicios de TI Para obtener más detalles acerca de los productos y servicios de TI, póngase en contacto mediante correo electrónico o acceda a la página inicial de calculadoras en la world wide web.

dirección de correo electrónico: **ti-cares@ti.com**

dirección de internet: **education.ti.com**

Información sobre servicios y garantías Para obtener más detalles acerca de la duración y las condiciones de la garantía o sobre el servicio de asistencia a productos, consulte la declaración de garantía que se adjunta a este producto o póngase en contacto con su distribuidor o minorista de Texas Instruments.

Precauciones con las pilas

Tome estas precauciones al sustituir las pilas.

- No deje las pilas al alcance de los niños.
- No mezcle pilas nuevas y usadas. No mezcle marcas de pilas (ni tipos de una misma marca).
- No mezcle pilas recargables y no recargables.
- Instale las pilas siguiendo los diagramas de polaridad (+ y -).
- No coloque pilas no recargables en un cargador de pilas.
- Deseche las pilas usadas inmediatamente en la forma adecuada.
- No queme ni desmonte las pilas.

Índice alfabético

Símbolos

- ➔, almacenar 310
- !, factorial 57, 304
- " , notación de segundo 307
- ≠, ≠, distinto de 303
- #, direccionamiento indirecto 306
- √(), raíz cuadrada 305
- %, porcentaje 302
- &, anexas 304
- ' , notación de minuto 307
- ' , primo 307
- Σ(), suma 306
- ∫(), integrar 304
- *, multiplicar 299
- +, sumar 298
- [F1]-[F8] (teclas de función)
 - desplazamiento entre menús de barras de herramientas 42
 - posición física 7
 - selección de categorías 22
 - selección de menús 38
 - usos 11
- °, notación de grados 125, 307
- ∠, ángulo 307
- .*, multiplicación de puntos 301
- +. , adición de puntos 301
- ./, división de puntos 301
- .^, potencia de puntos 301
- .-, resta de puntos 301
- /, dividir 300
- <, menor que 303
- =, igual 302
- >, mayor que 303
- [C] (tecla negativa) 11
- Δlist(), lista de diferencias 234
- ΔtmpCnv(), conversión del rango de temperatura 288
- ^, potencia 306
- _ , guión bajo 308
- ≤, <=, menor o igual que 303
- [] (tecla de sustracción) 11
- ≥, >=, mayor o igual que 304
- , comentario 310
- |, with 63, 309
- [←] / [→] [DEL] (borrar carácter) 14
- [M] (tecla de modificador de mano)
 - descripción 10
 - estado 30
 - posición física 7
- [2nd] (tecla de modificador secundaria)
 - descripción 10
 - estado 30
 - posición física 7
- [2nd] [MEM] (MEMORIA) 14
- [2nd] [EE] (tecla de exponente) 12
- [2nd] [▶] (conversiones de medidas) 14
- [2nd] [-] (unidades de medida) 14
- [2nd] [RCL] (recuperar) 14
- [2nd] [CATALOG] (Catálogo)
 - descripción 17
 - orden del teclado 14
 - órdenes 18
 - salida 19
- [2nd] [CUSTOM] (Personal)
 - descripción 43
 - ejemplo 43
 - orden del teclado 13
- [2nd] [CHAR] (Caracteres)
 - introducción de caracteres especiales 8
 - orden del teclado 14
 - selección de caracteres 8
- [2nd] [QUIT]
 - apagado de la calculadora 4
 - introducción de órdenes 18
 - orden del teclado 14
 - pantalla principal de la calculadora 20
 - salida del modo de pantalla dividida 49
- ▶, convertir 308
- ▶Bin, mostrar como binario 185
- ▶Cylind, mostrar como vector cilíndrico 200
- ▶DD, mostrar como ángulo decimal 203
- ▶Dec, mostrar como entero decimal 203

- ▶DMS, mostrar como grado/minuto/segundo 208
- ▶Hex, mostrar como hexadecimal 227
- ▶Polar, mostrar como vector polar 251
- ▶Rect, mostrar como vector rectangular 260
- ▶Sphere, mostrar como vector esférico 279
- Ⓢ (tecla de modificador Mayús)
 - descripción 10
 - estado 30
 - posición física 7
- Ⓢ (tecla de modificador de diamante)
 - descripción 10
 - estado 30
 - posición física 7
- Ⓢ F (FORMATOS/FORMATOS DE GRÁFICOS) 13
- Ⓢ N (archivo nuevo) 13
- Ⓢ O (abrir archivo) 13
- Ⓢ S (GUARDAR COMO)
 - cuadro de diálogo 13
 - descripción 13
 - ejemplo 41
- Ⓢ (teclas del cursor)
 - abrir Apps 22
 - función 11
 - funciones adicionales 11
 - introducción de órdenes 18
 - posición física 7
 - selección de pares entrada/respuesta 21
 - uso con la tecla de mano 10
 - uso del menú CARACTERES 8
 - utilizadas con la tecla de mano 10
- E, exponente 211
- , negar 302
- , restar 299
- Π(), producto 305
- r, radianes 306
- T, transponer 283

Números

- 0b, indicador binario 310
- 0h, indicador hexadecimal 310
- 10^(), potencia de diez 308
- 3D (tridimensional)
 - modo 31
- ▶Grad() 226
- ▶ln() 234
- ▶logbase() 236
- ▶Rad() 258

A

- Abrir archivo (Ⓢ O) 13
- abs(), valor absoluto 127, 182
- abscisa, P▶Rx() 248
- activar reloj, ClockOn 188
- Actividad
 - con funciones racionales 118
 - con la función Time-Value-of-Money 130
 - con paralelepípedos 115
 - con renta anual estándar 129
 - de extracción sin
 - reemplazamiento 133
- Actividades. Véase ejemplos, previsualizaciones, actividades
- Actual, modo 15
- actualización del sistema operativo (SO) 149, 150
- Adaptador de vídeo TI-Presenter
 - compatibilidad v
 - conexión 52
- agrupación trigonométrica, tCollect() 286
- Aleatoria
 - inicio del generador de números, RandSeed 113
 - matriz, randMat() 113
- aleatoria
 - matriz, randMat() 259
 - norma, randNorm() 259
- aleatorio
 - número, rand() 258
 - polinomio, randPoly() 259
- aleatorios
 - inicio del generador de números, RandSeed 259

- All (Todo), categoría 25
 - almacenar
 - base de datos gráfica, StoGDB 281
 - imagen, StoPic 281
 - símbolo, ➤ 310
 - Ampliar/concatenar, augment() 113
 - ampliar/concatenar, augment() 185
 - and (booleano), and 182
 - and, and booleano 182
 - AndPic, imagen con and 183
 - anexar, & 304
 - Angle (Ángulo)
 - modo 31
 - angle(), ángulo 183
 - ángulo, ∠ 307
 - ángulo, angle() 183
 - Ángulo, modo 15
 - ans(), última respuesta 184
 - Apagado 4
 - tras APD 5
 - tras inactividad 5
 - APD (Automatic Power Down)
 - durante cálculo o programa 5
 - en modo de descarga de SO 54
 - encendido posterior 5
 - Aplicaciones de software para dispositivos de mano (Apps) 3
 - iconos 2
 - preinstalado v
 - Aplicaciones Flash 157, 158, 160
 - aplicaciones Flash
 - borrado 144
 - APLICACIONES FLASH ( [APPS])
 - descripción 39
 - orden del teclado 13
 - sin instalar 17
 - APLICACIONES, menú ( [APPS]) 44
 - approx(), aproximación 184
 - Apps (aplicaciones de software para dispositivos de mano)
 - abrir 22, 44
 - alternancia 49
 - eliminación 51
 - icono resaltado, última abierta 3
 - iconos 2
 - métodos abreviados 26
 - preinstalado v
 - aproximación, approx() 184
 - Archivar variables, Archive 170
 - archivar variables, Archive 184
 - Archive, archivar variables 170, 184
 - Archivo, abrir ( O) 13
 - Archivo, nuevo ( N) 13
 - arcLen(), longitud del arco 184
 - Área de historia
 - estado 31
 - asíntotas 70
 - augment(), ampliar/concatenar 113, 185
 - Automatic Power Down (APD)
 - durante cálculo o programa 5
 - en modo de descarga de SO 54
 - encendido posterior 5
 - avgRC(), índice de cambio promedio 185
- ## B
- barra de herramientas
 - activar, CustmOn 199
 - definir, Custom 199
 - desactivar, CustmOff 199
 - Base, modo 15
 - binario
 - indicador, 0b 310
 - mostrar, ▶Bin 185
 - BldData, crear datos 186
 - bloquear variable, Lock 236
 - booleano
 - and, and 182
 - not, not 245
 - or exclusivo, xor 292
 - or, or 246
 - BorPrinc 22
 - Borrado de variables 107
 - borrar
 - dibujo, ClrDraw 188
 - error, ClrErr 189
 - Borrar carácter ( /  [DEL]) 14
 - BorrTipo(), DelType() 204
 - bucle, Loop 237
- ## C
- cable de conectividad TI 137, 150, 153

- Cable de unidad-a-unidad 54
 - conexión 51
- Cable TI-GRAPH LINK
 - instalación de Apps 49
- Cables v, 51
- cables 137, 150, 153
- cadena de entrada, InputSt 147
- cadena
 - anexar, & 304
 - cadena en expresión, expr() 215
 - código de carácter, ord() 247
 - de caracteres, char() 187
 - de entrada, InputSt 229
 - dentro, InString 229
 - derecha, right() 262
 - desplazar, shift() 272
 - direccionamiento indirecto, # 306
 - expresión en cadena, string() 281
 - formato, format() 220
 - interior cadena, mid() 240
 - introducción, InputSt 147
 - izquierda, left() 232
 - número de caracteres, dim() 207
 - operaciones 181
 - rotar, rotate() 262
- cambiar
 - switch() 283
- Caracteres
 - eliminación 14
 - especiales 8, 10, 14
 - griegos 8, 14
 - internacionales/accentuados 8, 14
 - matemáticos 8, 14
 - mayúsculas 7, 10
- caracteres
 - cadena, char() 187
 - código numérico, ord() 247
- Carpetas 162
 - definir, setFold() 164
 - eliminación 168
 - nuevas, NewFold 164
 - pegar nombre 168
 - VAR LINK 160, 161, 166
- carpetas
 - definir, setFold() 267
 - eliminar, DelFold 204
 - nuevas, NewFold 242
 - obtener/devolver, getFold() 223
 - transmisión 139, 140, 142, 143
- Catálogo (2nd) [CATALOG]
 - descripción 17
 - orden del teclado 14
 - órdenes 18
 - salida 19
- Categorías
 - All (Todo) 25
 - ejemplo de edición 27
 - English (Inglés) 25
 - Escritorio de Apps 26
 - Graphing (Representación gráfica) 25
 - Math (Matemáticas) 25
 - personalización 26
 - Science (Ciencia) 25
 - selección de vacías 26
 - SocSt (EstSoc (Estudios sociales)) 25
 - Utils (Utilids (Utilidades)) 25
- CBL
 - actividad 123
 - enviar variable de lista, Send 265
 - obtener/devolver, Get 221
 - programas 123
- CBR
 - enviar variable de lista, Send 265
 - obtener/devolver, Get 221
 - programas 123
- ceiling(), entero superior 114, 186
- Ceros
 - actividad 126
- Ceros, zeros() 110
- ceros, zeros() 292
- certificado 144, 149, 150, 151, 152, 153
- cFactor(), factor complejo 132, 187
- char(), cadena de caracteres 187
- checkTmr(), comprobar temporizador 188
- ciclo, Cycle 199
- Circle, dibujar círculo 188
- Círculo
 - representación gráfica 66
- círculo, Circle 188
- Circunferencia

- representación gráfica 69, 70
- clasificar
 - en orden ascendente, SortA 278
 - en orden descendente, SortD 279
- ClockOff, desactivar reloj 188
- ClockOn, activar reloj 188
- ClrDraw, borrar dibujo 188
- ClrErr, borrar error 189
- ClrGraph, vaciar gráfica 189
- ClrHome, vaciar pantalla Home 189
- ClrIO, vaciar E/S 189
- colDim(), número de columnas de la matriz 190
- colNorm(), máximo de las columnas de la matriz 190
- combinaciones, nCr() 241
- comDenom(), denominador común 190
- comentario, ● 310
- complejo
 - factor, cFactor() 187
 - número conjugado, conj() 191
 - resolver, cSolve() 196
- Complejos
 - factor, cFactor() 132
 - números 57
- complejos
 - ceros, cZeros() 200
- comprobar temporizador,
 - checkTmr() 188
- Condiciones de error tras APD 5
- conectar y transmitir 265, 266
 - enviar variable de lista, Send 265
 - obtener/devolver valor CBL/CBR, Get 221
- Conexión
 - adaptador de vídeo TI-Presenter 51
 - dispositivos 51
 - ordenador 51
 - panel de proyección TI ViewScreen 51
 - Voyage 200 51
- conj(), número complejo conjugado 191
- Contraste
 - ajuste 2, 53
 - primeros pasos 2
- conversión
 - de la temperatura, tmpCnv() 288
 - del rango de temperatura, ΔtmpCnv() 288
- convertir hora, timeCnv() 287
- Convertir medidas 14
- convertir, ► 308
- Copiar variable, CopyVar 166
- copiar variable, CopyVar 191
- CopyVar, copiar variable 166, 191
- cos(), coseno 191
- cos⁻¹(), arco coseno 192
- cosecante hiperbólica, csch() 195
- cosecante, csc(), 195
- coseno, cos() 191
- cosh(), coseno hiperbólico 193
- cosh⁻¹(), arco coseno hiperbólico 193
- cot(), cotangente 193
- cot⁻¹(), cotangente inversa 194
- cotangente hiperbólica, coth() 194
- cotangente, cot(), 193
- coth(), cotangente hiperbólica 194
- coth⁻¹(), cotangente hiperbólica inversa 194
- crear
 - datos, BldData 186
 - tabla, Table 284
- crossP(), producto vectorial 194
- csc(), cosecante 195
- csc⁻¹(), cosecante inversa 195
- csch(), cosecante hiperbólica 195
- csch⁻¹(), cosecante hiperbólica inversa 195
- cSolve(), resolver complejo 196
- Cuadro de diálogo
 - ☐ S (GUARDAR COMO) 13
 - CLOCK (RELOJ) 33
 - Editar categorías 26
 - FORMATOS/FORMATOS DE GRÁFICO ☐ F) 13
 - indicador de menú 41
 - MODO 15
 - para abrir Apps 22
- Cuadro de diálogo CLOCK (RELOJ) 33
- Cuando, when() 84
- cuando, when() 291
- cuarto grado, regresión, QuartReg 257

cúbica, regresión, CubicReg 198
CubicReg, regresión cúbica 198
cumSum(), suma acumulada 198
Cursos
 desplazamiento 11
 eliminación de caracteres 14
 eliminación de un par entrada/
 respuesta 22
 en el área de historia 31
 funcionalidad 11
 posición tras APD 5
 selección de órdenes 18
 visualización de entradas 21
CustmOff, desactivar barra de
 herramientas personalizada 199
CustmOn, activar barra de
 herramientas personalizada 199
Custom, definir barra de
 herramientas 199
Cycle, ciclo 199
CyclePic, serie de imágenes 200
cZeros(), ceros complejos 200

D

d(), primera derivada 202
Data/Matrix Editor
 desplazar, shift() 272
data▶mat() 202
datos
 (nuevos), NewData 242
Datos, gráficas 89
dayOfWk(), día de la semana 202
de lo contrario si, Elseif 212
de lo contrario, Else 228
decimal
 mostrar como ángulo, ▶DD 203
 mostrar como entero, ▶Dec 203
Define, definir 109, 203
definidas por el usuario
 funciones 203
Definir
 carpeta, setFold() 164
 Define 109
definir
 barra de herramientas, Toolbar
 289
 carpeta, setFold() 267

 gráfica, setGraph() 267
 modo, setMode() 268
 tabla, setTable() 269
 unidades, setUnits() 270
definir fecha, setDate() 266
definir formato de fecha,
 setDtFmt() 267
definir formato de hora, setTmFmt()
 269
definir hora, setTime() 269
definir zona horaria, setTmZn() 270
definir, Define 203
deleting
 variables of type 168
DelFold, eliminar carpeta 204
DelType command 168
DelType(), BorrTipo() 204
DelVar, eliminar variable 204
denominador 190
 común, comDenom() 190
dentro de cadena, inString() 229
derecha, right() 262
derivadas
 derivada numérica, nDeriv() 242
 primera derivada, d() 202
desactivar reloj, ClockOff 188
Desarchivar variables, Unarchiv 170
desarchivar variables, Unarchiv 290
Desarrollar, expand() 111, 127
desarrollar, expand() 214
desbloquear, Unlock 290
deSolve(), solución 205
Desplazamiento 21
desplazar, shift() 272
desviación estándar, stdDev() 280
desvPbst(), stdDevPop() 280
det(), determinante de matriz 206
devolver cadena de fecha,
 getDtStr() 223
devolver cadena de hora,
 getTmStr() 224
devolver fecha actual, getDate() 222
devolver formato de fecha,
 getDtFmt() 223
devolver formato de hora,
 getTmFmt() 224
devolver hora actual, getTime() 224
devolver Véase obtener/devolver

devolver zona horaria, getTmZn() 225
devolver, Return 261
día de la semana, dayOfWk() 202
diag(), diagonal de la matriz 206
Dialog, definir recuadro de diálogo 207
dibujos y dibujar
 borrar, ClrDraw 188
 círculo, Circle 188
 contorno, DrwCtour 211
 función, DrawFunc 209
 inversa, DrawInv 209
 paramétrica, DrawParm 209
 pendiente, DrawSlp 210
 polar, DrawPol 210
dibujos y dibujar, recta
 horizontal, LineHorz 233
 Line 233
 tangente, LineTan 233
 vertical, LineVert 233
diferencias, lista, Δlist() 234
difImp(), impDif() 229
dim(), número de caracteres 207
direccionamiento indirecto, # 306
Disp, mostrar pantalla de E/S 98, 207
DispG, mostrar gráfica 208
DispHome, mostrar pantalla Home 208
DispTbl, mostrar tabla 208
distinto de, ≠, /= 303
dividir, / 300
Documentos
 actividad 116
 de órdenes, actividad 116
 tutorial 116
dotP(), producto escalar 209
DrawFunc, dibujar función 209
DrawInv, dibujar inversa 209
DrawParm, dibujar paramétrica 209
DrawPol, dibujar polar 210
DrawSlp, dibujar pendiente 210
DropDown, menú desplegable 210
DrwCtour, dibujar contorno 211

E

E (símbolo de exponente) 12

e elevado a la potencia, $e^{\wedge}()$ 211
E, exponente 211
 $e^{\wedge}()$, e elevado a la potencia 211
ecuaciones
 simultáneas, simulti() 274
ED (ecuación diferencial)
 modo 31
Editor de programas 23
Editor de ventanas 45
eigVc(), vector propio 212
eigVl(), valor propio 212
ejecutar
 lenguaje ensamblador, Exec 214
 programa, Prgm 252
Ejecutar programa, Prgm 96
Ejemplo
 activación y desactivación del
 menú personalizado 43
 cambio de los valores de
 configuración de modo 16
 creación de un programa nuevo
 23
 desactivación del reloj 37
 edición de categorías 27
 restauración del menú
 personalizado
 predeterminado 43
 selección de opciones de menú
 39
 uso de cuadros de diálogo 41
 uso del Catalog (Catálogo) 18
 uso del mapa del teclado 9
 uso del menú CARACTERES 8
Ejemplos, previsualizaciones,
 actividades
 actividad con $\cos(x)=\sin(x)$ 113
 árboles y bosque 76
 bases numéricas 102
 béisbol 125
 cálculo simbólico 63
 ceros complejos 126
 constantes 65
 conversión de medidas de
 ángulos 62
 Data/Matrix Editor 88
 descomposición de una función
 racional 118
 detección de discontinuidades 70

- documento de tutorial con Text Editor 116
- ecuaciones diferenciales 80
- estadísticas 89
- expansión de expresiones 59
- extracción sin reemplazamiento 133
- factores complejos 132
- factores primos 57
- factores racionales 132
- factores reales 132
- factorial 57
- filtrado de datos 120
- fórmula de segundo grado 110
- función Time-Value-of-Money 130
- funciones definidas a trozos 83
- gestión de la memoria 104
- gestión de variables 104
- números complejos 57
- obtención de logaritmos de cualquier base 62
- obtención de raíces 58
- operaciones con texto 98
- pantalla dividida 86, 125
- población 89
- polinomio de tercer grado 126
- problema poste-esquina 109
- programa del CBL 123
- programación 96, 98
- recorrido de un proyectil 72
- renta anual estándar 129
- representación gráfica de funciones 66, 69
- representación gráfica de sucesiones 76
- representación gráfica en 3D 77, 115
- representación gráfica en paramétricas 72, 125
- resolución de desigualdades 60
- rosa polar 74
- tablas 85
- Teorema de Pitágoras 109
- unidades de medida 65
- elemento de menú, Item 231
- eliminar
 - carpeta, DelFold 204
 - variable, DelVar 204
- Else, de lo contrario 228
- Elseif, de lo contrario si 212
- Encendido
 - primeros pasos 1
- EndCstm, terminar personalizar 199
- EndDlog, terminar recuadro de diálogo 207
- EndFor, terminar para 219
- EndFunc, terminar función 220
- EndIf, terminar si 228
- EndLoop, terminar bucle 237
- EndPrgm, terminar programa 96, 252
- EndTBar, terminar barra de herramientas 289
- EndTry, terminar intentar 289
- EndWhile, terminar mientras 291
- English (Inglés), categoría 25
- enlace y transmisión
 - aplicaciones Flash 139, 140, 142, 145, 146
 - cancelación 144
 - carpetas 139, 140, 143, 144
 - de calculadora a calculadora 137, 139, 142, 146, 147, 148, 149
 - enviar a calculadora, SendCalc 146, 147
 - enviar chat, SendChat 146, 147
 - errores 144, 152
 - programa 146, 147
 - variables 139, 140, 142, 143
- Entero
 - inferior, floor() 114
 - superior, ceiling() 114
- entero
 - de división, intDiv() 230
 - inferior, floor() 217
 - int() 230
 - superior, ceiling() 186
- entonces, Then 228
- entrada
 - entry() 213
 - Input 229
- entry(), entrada 213
- enviar
 - a calculadora, SendCalc 265

- charla, SendChat 266
- variable de lista, Send 265
- enviar a calculadora, SendCalc 146, 147
- enviar chat, SendChat 146, 147
- errores y resolución de problemas
 - borrar error, ClrErr 189
 - transferir error, PassErr 250
- errores y solución de problemas
 - transmisión 144, 152
- esArchiv(), isArchiv() 230
- esBloq(), isLocked() 231
- escalar
 - producto, dotP() 209
- Escribir
 - caracteres en mayúsculas 7
 - nombre de archivo 23
 - para desplazarse por el Catalog (Catálogo) 18
- Escritorio de Apps
 - apagado de la calculadora 4
 - categorías 22, 26
 - desactivación 31
 - estado de pantalla dividida 29
 - fecha y hora 34
 - modo 15, 31
 - partes 3
 - primeros pasos 2
 - reloj 33
 - y pantalla principal de la calculadora 20
- está el reloj activado, isClkOn() 230
- Estadísticas
 - inicio del generador de números aleatorios, RandSeed 113
- estadísticas
 - activar gráficas, PlotsOn 251
 - combinaciones, nCr() 241
 - desactivar gráficas, PlotsOff 251
 - desviación estándar, stdDev() 280
 - estadísticas de una variable, OneVar 246
 - factoriales, ! 304
 - gráfica nueva, NewPlot 243
 - inicio del generador de números aleatorios, RandSeed 259
 - media, mean() 239
 - mediana, median() 239
 - mostrar resultados, ShowStat 273
 - norma aleatoria, randNorm() 259
 - número aleatorio, rand() 258
 - operaciones 181
 - permutaciones, nPr() 245
 - resultados de dos variables, TwoVar 289
 - varianza, variance() 290
- Estado
 - carpeta actual 30
 - en escritorio de Apps 2, 14
 - modo APROX 31
 - modo AUTO 31
 - modo EXACT 31
 - modo Gráfico 31
 - modo Número de gráfico 31
 - Ocupado/Pausa 31
 - pantalla dividida 28
 - pilas gastadas 53
 - variable bloqueada/archivada 31
 - visualización 3
- estilo, Style 282
- esVar(), isVar() 231
- etiqueta, Lbl 231
- evaluar polinomio, polyEval() 252
- exact(), modo Exact 213
- Exacto/aprox, modo 15
- examples, previews, activities
 - angle modes 62
- Exec, ejecutar lenguaje ensamblador 214
- Exit, salir 214
- explist(), expresión a lista 214
- expand(), desarrollar 111, 127, 214
- expand(), expand 58
- expand, expand() 58
- expansión trigonométrica, tExpand() 287
- exponencial, regresión, ExpReg 216
- exponente, E 211
- expr(), cadena en expresión 215
- ExpReg, regresión exponencial 216
- Expresiones 20
- expresiones
 - cadena en expresión, expr() 215

expresión a lista, `explist()` 214
expressions
expanding 58

F

`factor()`, factor 216
Factor, `factor()` 112, 132
`factor, factor()` 216
Factorial, ! 57
factorial, ! 304
factorización QR, QR 256
Factorizar
 actividad 132
Fecha
 ajuste 32
 reajuste 38
 visualización 3
fila o columna, vector, `unitV()` 290
Fill, llenar matriz 217
Filtrado de datos 120
Flash, actualización del sistema
 operativo 149, 150
`floor()`, entero inferior 114, 217
`fMax()`, función máxima 218
`fMin()`, función mínima 218
`FnOff`, desactivar función 218
`FnOn`, activar función 219
For, para 219
Forma
 reducida escalonada, `rref()` 113
forma
 escalonada, `ref()` 260
 reducida escalonada, `rref()` 264
`format()`, formato de cadena 220
Formato complejo, modo 15
formato de cadena, `format()` 220
Formato exponencial, modo 15
Formato vectorial, modo 15
FORMATOS (☐ F)
 cuadro de diálogo 13
 orden del teclado 13
FORMATOS DE GRÁFICO (☐ F) 13
 cuadro de diálogo 13
FORMATS, cuadro de diálogo 80, 81
`fPart()`, parte de función 220
Fracción propia, `propFrac` 118
fracción propia, `propFrac` 253

Fracciones 118
fracciones 253
FUNC (función)
 modo 31
Func, función de programa 220
Funciones 17
funciones
 activar, `FnOn` 219
 definidas por el usuario 203
 desactivar, `FnOff` 218
 función de programa, `Func` 220
 máxima, `fMax()` 218
 mínima, `fMin()` 218
 parte, `fPart()` 220
Funciones definidas a trozos 83

G

Garbage collection, mensaje 171,
 173, 174
`gcd()`, máximo común divisor 221
`Get`, obtener/devolver valor CBL/CBR
 221
`GetCalc`, obtener/devolver
 calculadora 146, 147, 221
`getConfig()`, obtener/devolver
 configuración 222
`getDate()`, devolver fecha actual 222
`getDenom()`, obtener/devolver
 denominador 222
`getDtFmt()`, devolver formato de
 fecha 223
`getDtStr()`, devolver cadena de
 fecha 223
`getKey()`, obtener/devolver tecla
 223
`getMode()`, obtener/devolver modo
 224
`getNum()`, obtener/devolver
 número 224
`getTime()`, devolver hora actual 224
`getTmFmt()`, devolver formato de
 hora 224
`getTmStr()`, devolver cadena de
 hora 224
`getTmZn()`, devolver zona horaria
 225

getType(), obtener/devolver tipo 225
getUnits(), obtener/devolver unidades 226
Goto, ir a 226
GRA (grados)
 modo 31
GRAD(gradian) mode 62
Gradian angle mode 62
gradian,^G 306
grado/minuto/segundo, mostrar como, ►DMS 208
Gráfica, Graph 84
gráfica, Graph 227
Gráficas
 datos 89
gráficas
 activar, PlotsOn 251
 desactivar, PlotsOff 251
 nuevas, NewPlot 243
Gráficas y representación gráfica
 trazar, Trace 115, 122, 124, 127
 Y= editor 66, 69, 70
gráficas y representación gráfica
 activar funciones, FnOn 219
 almacenar base de datos gráfica, StoGDB 281
 definir, setGraph() 267
 desactivar funciones, FnOff 218
 estilo, Style 282
 gráfica, Graph 227
 operaciones 178
 restablecer base de datos gráfica, RclGDB 259
 sombreado, Shade 271
 trazar, Trace 289
 vaciar, ClrGraph 189
Gráfico
 modo 31
 modo de número 31
Gráfico, modo 15
Gráficos
 dibujo sobre 10
 número 47, 49
Graph, gráfica 84, 227
Graphing (Representación gráfica), categoría 25
Griegos, caracteres 8

GUARDAR COMO () S
 cuadro de diálogo 13
 descripción 13
 ejemplo 41
guión bajo, _ 308

H

hexadecimal
 indicador, 0h 310
 mostrar, ►Hex 227
hiperbólica
 tangente, tanh() 285
hiperbólico
 coseno, cosh() 193
 seno, sinh() 275
Hora
 ajuste 32
 reajuste 38
 visualización 3

I

Icono principal 20
identidad, matriz, identity() 228
identity(), matriz de identidad 228
Idioma, modo
 cambio de los valores de configuración 16
 visualización 15
If, si 228
igual, = 302
imag(), parte imaginaria 228
imagen
 con and, AndPic 183
 con or exclusivo, XorPic 292
imágenes
 almacenar, StoPic 281
 and, AndPic 183
 nuevas, NewPic 243
 or exclusivo, XorPic 292
 recuperar, RclPic 259
 reemplazar, RplcPic 264
 serie, CyclePic 200
ImpDif(), difImp() 229
Impresión nítida, modo 15
Indicador de historia 21
índice de cambio promedio, avgRC() 185

- iniciar temporizador, startTmr() 279
 - Input, entrada 229
 - InputSt, cadena de entrada 147, 229
 - Inserción, modo (2nd [INS]) 14
 - inString(), dentro de cadena 229
 - Instrucciones
 - Catálogo 17
 - pantalla principal de la calculadora 20
 - int(), entero 230
 - intDiv(), entero de división 230
 - integrar, ∫() 304
 - intentar, Try 289
 - interior cadena, mid() 240
 - Internacionales/accentuados, caracteres 8
 - inverso, x(309
 - inverso, x⁻¹ 309
 - iPart(), parte entera 76, 230
 - ir a, Goto 226
 - isArchiv(), esArchiv() 230
 - isArchiv(), is archived 160
 - isClkOn(), está el reloj activado 230
 - isLocked(), esBloq() 231
 - isLocked(), is locked 161
 - isPrime(), prueba de número primo 231
 - isVar(), esVar() 231
 - isVAR(), is variable 160
 - Item, elemento de menú 231
 - izquierda, left() 232
- L**
- Lbl, etiqueta 231
 - lcm, mínimo común múltiplo 232
 - left(), izquierda 232
 - lenguaje ensamblador 214
 - limit(), limitar 232
 - limitar, limit() 232
 - Line, dibujar recta 233
 - Línea de entrada
 - borrado del área de historia 22
 - introducción de órdenes 18
 - permanencia del cursor 21
 - recuperación 21
 - Línea de estado
 - información de historia 21
 - parámetros de orden 19
 - lineal
 - de mediana a mediana, regresión, MedMed 239
 - regresión, LinReg 234
 - LineHorz, dibujar recta horizontal 233
 - LineTan, dibujar recta tangente 233
 - LineVert, dibujar recta vertical 233
 - LinReg, regresión lineal 234
 - list▶mat(), lista a matriz 234
 - lista de ID 153, 154
 - listas
 - a matriz, list▶mat() 234
 - ampliar/concatenar, augment() 185
 - clasificar en orden ascendente, SortA 278
 - clasificar en orden descendente, SortD 279
 - datos nuevos, NewData 242
 - diferencias, Δlist() 234
 - expresión a lista, exp▶list() 214
 - interior cadena, mid() 240
 - lista a matriz, list▶mat() 234
 - matriz a lista, mat▶list() 238
 - máximo, max() 238
 - mínimo, min() 240
 - nuevas, newList() 242
 - número de caracteres, dim() 207
 - operaciones 178
 - producto escalar, dotP() 209
 - producto vectorial, crossP() 194
 - producto, product() 253
 - suma acumulada, cumSum() 198
 - sumar, sum() 262, 282
 - ln(), logaritmo natural 235
 - LnReg, regresión logarítmica 235
 - Local, variable local 235
 - local, variable, Local 235
 - Lock, bloquear variable 236
 - log(), logaritmo 236
 - logarítmica, regresión, LnReg 235
 - logaritmo natural, ln() 235
 - logaritmo, log() 236
 - logaritmos 235, 236
 - Logistic, regresión logística 237
 - logística, regresión, Logistic 237

longitud del arco, `arcLen()` 184
Loop, bucle 237
LU, descomposición inferior-superior de la matriz 237

M

`mat▶data()` 238
`mat▶list()`, matriz a lista 238
Matemáticos, caracteres 8
Math (Matemáticas), categoría 25
Matrices
 aleatorias, `randMat()` 113
 ampliar/concatenar, `augment()` 113
matrices
 adición con filas, `rowAdd()` 263
 adición de puntos, `.+` 301
 aleatorias, `randMat()` 259
 ampliar/concatenar, `augment()` 185
 datos nuevos, `NewData` 242
 descomposición inferior-superior, LU 237
 determinante, `det()` 206
 diagonal, `diag()` 206
 división de puntos, `./` 301
 factorización QR, QR 256
 forma escalonada, `ref()` 260
 forma reducida escalonada, `rref()` 264
 identidad, `identity()` 228
 intercambio de las filas, `rowSwap()` 264
 lista a matriz, `list▶mat()` 234
 llenar, `Fill` 217
 matriz a lista, `mat▶list()` 238
 máximo de las columnas, `colNorm()` 190
 máximo de las filas, `rowNorm()` 263
 máximo, `max()` 238
 mínimo, `min()` 240
 multiplicación de puntos, `.*` 301
 multiplicación y adición con filas, `mRowAdd()` 241
 nuevas, `newMat()` 243
 número de caracteres, `dim()` 207

número de columnas, `colDim()` 190
número de filas, `rowDim()` 263
operaciones 179
operaciones con filas, `mRow()` 241
potencia de puntos, `.^` 301
producto, `product()` 253
resta de puntos, `.-` 301
submatriz, `subMat()` 282
suma acumulada, `cumSum()` 198
sumar, `sum()` 262, 282
transponer, `↑` 283
valor propio, `eigVl()` 212
vector propio, `eigVc()` 212
matriz a lista, `mat▶list()` 238
`max()`, máximo 238
máximo
 común divisor, `gcd()` 221
 `max()` 238
mayor
 o igual que, `≥`, `>=` 304
 que, `>` 303
Mayúsculas, caracteres 7
`mean()`, media 239
media, `mean()` 239
`median()`, mediana 239
mediana, `median()` 239
Medida
 conversiones (`[2nd] [▶]`) 14
 unidades (`[2nd] [-]`) 14
MedMed, regresión lineal de mediana a mediana 239
Memoria 157
 archivar, `Archive` 170
 comprobar 157, 158
 desarchivar, `Unarchive` 170
 pantalla VAR LINK 158, 159, 160, 161, 166, 170
 reiniciar 157, 158
memoria
 archivar, `Archive` 184
 desarchivar, `Unarchive` 290
MEMORIA (`[2nd] [MEM]`) 14
Memoria Flash de sólo lectura v menor
 o igual que, `≤`, `<=` 303
 que, 303

- mensaje, Prompt() 253
- Mensajes
 - Garbage collection 171, 173, 174
- Menú CARACTERES (2nd [CHAR])
 - introducción de caracteres especiales 8
- menú desplegable
 - DropDown 210
 - PopUp 252
- Menús
 - APLICACIONES (APPS) 44
 - APLICACIONES FLASH (♦ [APPS]) 13
 - APLICACIONES FLASH (APPS) 45
 - cancelación 42
 - CARACTERES 8, 14
 - opciones 10
 - opciones de submenú 40
 - PERSONAL (2nd [CUSTOM]) 13, 43
 - selección de opciones 39
- Menús de barras de herramientas
 - desplazamiento entre 42
 - pantalla principal de la calculadora 38
 - selección de operaciones matemáticas 11, 20
 - sustitución por menús personalizados 43
- mid(), interior cadena 240
- mientras, While 291
- min(), mínimo 240
- mínimo
 - común múltiplo, lcm 232
 - min() 240
- mod(), módulo 241
- Modo de pantalla dividida
 - estado y Apps abiertas 2
 - gráfico activo 31
- modo Exact, exact() 213
- Modos
 - 3D (tridimensional) 31
 - Actual 15
 - Ángulo 15, 31
 - APROX 31
 - atenuado 15
 - AUTO 31
 - Base 15
 - definiciones 15
 - ED (ecuación diferencial) 31
 - Escritorio de Apps 15, 31
 - EXACT 31
 - Exacto/Aprox 15
 - Formato complejo 15
 - Formato exponencial 15
 - Formato vectorial 15
 - FUNC (función) 31
 - GRA (grados) 31
 - Gráfico 15
 - Idioma 15, 16
 - Impresión nítida 15
 - Inserción (2nd [INS]) 14
 - Mostrar dígitos 15
 - Número de gráfico 31
 - Pantalla completa 14, 24, 29, 47, 49
 - Pantalla dividida 2, 15, 24, 28, 31, 45, 47, 49
 - PAR (paramétrico) 31
 - POL (polar) 31
 - RAD (radianes) 31
 - Sistema de unidades 15
 - Sobrescritura (2nd [INS]) 14
 - SUC (sucesión) 31
 - Tipo Gráfico 31
 - Unidades personalizadas 15
 - Unit System 65
- modos
 - definir, setMode() 268
 - obtener/devolver, getMode() 224
- módulo, mod() 241
- Mostrar
 - pantalla de E/S, Disp 98
- mostrar
 - gráfica, DispG 208
 - pantalla de E/S, Disp 207
 - pantalla Home, DispHome 208
 - tabla, DispTbl 208
- mostrar como
 - ángulo decimal, ►DD 203
 - binario, ►Bin 185
 - entero decimal, ►Dec 203
 - grado/minuto/segundo, ►DMS 208
 - hexadecimal, ►Hex 227
 - vector cilíndrico, ►Cylind 200

- vector esférico, ▶Sphere 279
- vector polar, ▶Polar 251
- vector rectangular, ▶Rect 260

Mostrar dígitos, modo 15

mostrar resultados estadísticos, ShowStat 273

mover variable, MoveVar 241

MoveVar, mover variable 241

mRow(), operaciones con filas de matriz 241

mRowAdd(), multiplicación y adición con filas de matriz 241

multiplicar, * 299

N

nCr(), combinaciones 241

nDeriv(), derivada numérica 242

negar, - 302

NewData, datos nuevos 242

NewFold, carpeta nueva 164, 242

newList(), lista nueva 242

newMat(), matriz nueva 243

NewPic, imagen nueva 243

NewPlot, gráfica nueva 243

NewProb, problema nuevo 244

nInt(), integral numérica 244

norm(), norma de vector o matriz 244

norma de vector o matriz, norm() 244

not (booleano), not 245

not, not booleano 245

Notación

- de grados, ° 125

notación

- de grados, ° 307
- de minuto, ' 307
- de segundo, 307

Notación científica 11

nPr(), permutaciones 245

nSolve(), solución numérica 246

Nueva

- carpeta, NewFold 164

nueva

- carpeta, NewFold 242
- gráfica, NewPlot 243
- imagen, NewPic 243

- lista, newList() 242
- matriz, newMat() 243

nuevo

- problema, NewProb 244

Nuevo archivo (📄 N) 13

nuevos

- datos, NewData 242

numérica

- derivada, nDeriv() 242
- integral, nInt() 244
- solución, nSolve() 246

número

- de caracteres, dim() 207

número de identificación (ID) 149, 150, 153, 154

Números negativos 11

O

obtener/devolver

- calculadora, GetCalc 146, 147, 221
- carpeta, getFold() 223
- configuración, getConfig() 222
- denominador, getDenom() 222
- modo, getMode() 224
- número, getNum() 224
- tecla, getKey() 223
- tipo, getType() 225
- unidades, getUnits() 226
- valor CBL/CBR, Get 221

OCUPADO 31

OneVar, estadísticas de una variable 246

operaciones

- algebraicas 178
- de cálculo 178
- matemáticas 179

or

- (booleano), or 246
- exclusivo (booleano), xor 292

ord(), código de carácter numérico 247

ordenada, ▶Ry() 248

Órdenes

- Apps Flash 17
- Tecla v
- Teclado 9

- Órdenes del teclado
 - caracteres especiales 8
 - mapa del teclado 9
- Output, salida 247
- P**
 - P►Rx(), abscisa 248
 - P►Ry(), ordenada 248
 - Panel de proyección TI ViewScreen
 - compatibilidad v
 - conexión 51
 - Pantalla ACERCA DE 50
 - Pantalla completa, modo
 - [2nd] [QUIT] 14
 - cambio desde pantalla dividida 49
 - escritorio de Apps 29
 - presentación de Apps 49
 - pantalla dividida
 - cambiar, switch() 283
 - Pantalla dividida arriba-abajo
 - definición 45
 - definición de Apps iniciales 47
 - estado 28
 - Pantalla dividida izquierda-derecha
 - definición 45
 - definición de Apps iniciales 47
 - estado 28
 - Pantalla dividida, modo
 - definición 45
 - definición de Apps iniciales 47
 - división 1 App 47
 - división 2 App 47
 - especificación de Apps mostradas 47
 - estado 28
 - número de gráficos 47
 - proporción 47
 - regreso desde una App 24
 - salida 49
 - selección de la App activa 49
 - visualización 15
 - Pantalla principal de la calculadora
 - [2nd] [QUIT] 14
 - apagado de la calculadora 4
 - introducción de órdenes 18
 - menú personalizado 43
 - menús de barras de herramientas 38
 - orden del teclado 14
 - regreso al escritorio de Apps 32
 - teclas de función 11
 - Pantalla principal. Véase pantalla principal de la calculadora
 - PAR (paramétrico)
 - modo 31
 - para, For 219
 - parar
 - Stop 281
 - Pares entrada/respuesta
 - estado 31
 - part(), parte 248
 - parte
 - entera, iPart() 230
 - imaginaria, imag() 228
 - part() 248
 - Parte entera, iPart() 76
 - PassErr, transferir error 250
 - PAUSA 31
 - pausa, Pause 250
 - Pause, pausa 250
 - Pegar automáticamente 21
 - permutaciones, nPr() 245
 - PERSONAL, menú ([2nd] [CUSTOM]) 43
 - descripción 43
 - orden del teclado 13
 - personalizada, barra de herramientas Véase barra de herramientas
 - Pilas
 - advertencias 54
 - estado agotado 31
 - indicador de sustitución 31
 - primeros pasos 1
 - prolongación de la duración 5
 - sustitución 1, 53
 - pixel
 - activar, PxlOn 255
 - cambiar, PxlChg 254
 - círculo, PxlCrcl 254
 - desactivar, PxlOff 255
 - prueba, pxlTest() 255
 - recta horizontal, PxlHorz 255
 - recta vertical, PxlVert 256
 - recta, PxlLine 255

texto, PxlText 256
 PlotsOff, desactivar gráficas 251
 PlotsOn, activar gráficas 251
 POL (polar)
 modo 31
 polar
 coordenada, R►Pr() 258
 mostrar como vector, ►Polar 251
 polinomio de Taylor, taylor() 286
 Polinomios
 actividad 126
 polinomios
 aleatorios, randPoly() 259
 evaluar, polyEval() 252
 polyEval(), evaluar polinomio 252
 PopUp, menú desplegable 252
 porcentaje, % 302
 potencia de diez, 10^() 308
 potencia, ^ 306
 potencial, regresión, PowerReg 252
 PowerReg, regresión potencial 252
 Previsualizaciones. Véase ejemplos,
 previsualizaciones, actividades
 Prgm, ejecutar programa 96, 252
 Primeros pasos 1
 primo, ' 307
 problemas (nuevos), NewProb 244
 product(), producto 253
 producto
 II() 305
 product() 253
 vectorial, crossP() 194
 Programas 17
 Programas y programación
 CBL 123
 CBR 123
 ejecutar programa, Prgm 96
 mostrar pantalla de E/S, Disp 98
 terminar programa, EndPrgm 96
 programas y programación
 borrar error, ClrErr 189
 bucle, Loop 237
 comentario, ● 310
 ctivar barra de herramientas
 personalizada, CustmOn 199
 de lo contrario si, ElseIf 212
 de lo contrario, Else 228
 desactivar barra de
 herramientas, CustmOff 199
 devolver, Return 261
 ejecutar lenguaje ensamblador,
 Exec 214
 ejecutar programa, Prgm 252
 elemento de menú, Item 231
 eliminar tabla, ClrTable 190
 entonces, Then 228
 entrada, Input 229
 etiqueta, Lbl 231
 formato de cadena, format()
 220
 función, Func 220
 intentar, Try 289
 ir a, Goto 226
 local, Local 235
 mensaje, Prompt() 253
 menú desplegable, DropDown
 210
 menú desplegable, PopUp 252
 mientras, While 291
 mostrar gráfica, DispG 208
 mostrar pantalla de E/S, Disp 207
 mostrar pantalla Home,
 DispHome 208
 mostrar tabla, DispTbl 208
 obtener/devolver calculadora,
 GetCalc 146, 147
 operaciones 180
 para, For 219
 parar, Stop 281
 pausa, Pause 250
 salida, Output 247
 salir, Exit 214
 si, If 228
 solicitar, Request 261
 terminar si, EndIf 228
 texto, Text 287
 título, Title 287
 transferir error, PassErr 250
 vaciar E/S, ClrIO 189
 vaciar gráfica, ClrGraph 189
 vaciar Home, ClrHome 189
 Programas y programación, definir
 Define 109
 programas y programación, definir

barra de herramientas, Custom
 199
 barra de herramientas, Toolbar
 289
 Define 203
 recuadro de diálogo, Dialog 207
 programas y programación,
 obtener/devolver
 carpeta, getFold() 223
 configuración, getConfig() 222
 desde calculadora, GetCalc 221
 modo, getMode() 224
 tecla, getKey() 223
 unidades, getUnits() 226
 programas y programación,
 terminar
 barra de herramientas, EndTBar
 289
 bucle, EndLoop 237
 función, EndFunc 220
 intentar, EndTry 289
 mientras, EndWhile 291
 para, EndFor 219
 personalizada, EndCustm 199
 programa, EndPrgm 252
 recuadro de diálogo, EndDlog
 207
 Prompt(), mensaje 253
 propFrac, fracción propia 118, 253
 prueba de número primo, isPrime()
 231
 PtChg, cambiar punto 253
 PtOff, desactivar punto 254
 PtOn, activar punto 254
 ptTest(), prueba de punto 254
 PtText, texto de punto 254
 Puerto
 accesorio 51
 E/S 51
 punto
 activar, PtOn 254
 adición, + 301
 cambiar, PtChg 253
 desactivar, PtOff 254
 división, / 301
 multiplicación, * 301
 potencia, ^ 301
 prueba, ptTest() 254

 resta, - 301
 texto, PtText 254
 PxlChg, cambiar pixel 254
 PxlCrcl, círculo de pixel 254
 PxlHorz, recta horizontal de pixel
 255
 PxlLine, recta de pixel 255
 PxlOff, desactivar pixel 255
 PxlOn, activar pixel 255
 pxlTest(), prueba de pixel 255
 PxlText, texto de pixel 256
 PxlVert, recta vertical de pixel 256

Q

QR, factorización QR 256
 QuadReg, regresión de segundo
 grado 257
 QuartReg, regresión de cuarto
 grado 257

R

r, radianes 306
 R►Pθ(), coordenada polar 258
 R►Pr(), coordenada polar 258
 RAD (radianes)
 modo 31
 radianes, r 306
 raíz cuadrada, √() 305
 raíz(), root() 262
 rand(), número aleatorio 258
 randMat(), matriz aleatoria 113, 259
 randNorm(), norma aleatoria 259
 randPoly(), polinomio aleatorio 259
 RandSeed, inicio del generador de
 números aleatorios 113, 259
 RclGDB, restablecer base de datos
 gráfica 259
 RclPic, recuperar imagen 259
 real(), real 260
 real, real() 260
 rectangular
 mostrar como vector, ►Rect 260
 recuadro de diálogo, definir, Dialog
 207
 recuperar
 imagen, RclPic 259
 Recuperar (2nd [RCL]) 14

redondear, round() 263
 reemplazar imagen, RplcPic 264
 ref(), forma escalonada 260
 Regresiones
 actividad con fórmula de
 segundo grado 110
 regresiones 234
 cúbicas, CubicReg 198
 de cuarto grado, QuartReg 257
 de segundo grado, QuadReg 257
 exponenciales, ExpReg 216
 lineales de mediana a mediana,
 MedMed 239
 lineales, LinReg 234
 logarítmicas, LnReg 235
 logísticas, Logistic 237
 potenciales, PowerReg 252
 sinusoidales, SinReg 276
 Reloj
 activación 38
 desactivación 37
 funcionamiento 32
 remain(), resto 261
 Rename, renombrar 261
 renombrar, Rename 261
 Representación gráfica con niveles
 de contorno 80
 Representación gráfica en 3D
 animación 77
 CONTOUR LEVELS 80
 HIDDEN SURFACE 80
 WIRE AND CONTOUR 80
 WIRE FRAME 80
 Representación gráfica en mallas
 transparentes 80
 Reproducción visual 67
 Request, solicitar 261
 Resaltar
 para ver nombre completo de
 App 2
 Resalte
 caracteres al editar 10
 resolver
 solve() 276
 Resolver, solve() 63
 respuesta (última), ans() 184
 restablecer
 base de datos gráfica, RclGDB
 259
 restar, - 299
 resto, remain() 261
 Resultados 20
 resultados de dos variables, TwoVar
 289
 Retroceso (\leftarrow) 14
 Return, devolver 261
 right(), derecha 262
 root(), raíz() 262
 rotar, rotate() 262
 rotate(), rotar 262
 round(), redondear 263
 rowAdd(), adición con filas de
 matriz 263
 rowDim(), número de filas de matriz
 263
 rowNorm(), máximo de las filas de
 matriz 263
 rowSwap(), intercambio de las filas
 de matriz 264
 RplcPic, reemplazar imagen 264
 rref(), forma reducida escalonada
 113, 264

S

salida, Output 247
 Salir ($\left[2nd\right]$ [QUIT]) 14
 salir, Exit 214
 Science (Ciencia), categoría 25
 sec(), secante 264
 sec⁻¹(), secante inversa 265
 secante hiperbólica, sech() 265
 secante, sec(), 264
 sech(), secante hiperbólica 265
 sech⁻¹(), secante hiperbólica inversa
 265
 segundo grado, regresión, QuadReg
 257
 Send, enviar variable de lista 265
 SendCalc, enviar a calculadora 146,
 147, 265
 SendChat, enviar charla 266
 SendChat, enviar chat 146, 147
 seno, sin() 274
 serie de imágenes, CyclePic 200

setDate(), definir fecha 266
 setDtFmt(), definir formato de fecha 267
 setFold(), definir carpeta 164, 267
 setGraph(), definir gráfica 267
 setMode(), definir modo 268
 setTable(), definir tabla 269
 setTime(), definir hora 269
 setTmFmt(), definir formato de hora 269
 setTmZn(), definir zona horaria 270
 setUnits(), definir unidades 270
 Shade, sombra 271
 shift(), desplazar 272
 ShowStat, mostrar resultados estadísticos 273
 si, If 228
 sign(), signo 273
 signo, sign() 273
 Símbolo de exponente (E) 12
 simul(), ecuaciones simultáneas 274
 sin(), seno 274
 sin⁻¹(), arco seno 275
 sinh(), seno hiperbólico 275
 sinh⁻¹(), arco seno hiperbólico 275
 SinReg, regresión sinusoidal 276
 sinusoidal, regresión, SinReg 276
 Sistema Calculator-Based Laboratory
 compatibilidad v
 conexión 51
 Sistema Calculator-Based Ranger
 compatibilidad v
 conexión 51
 Sistema CBL 2/CBL
 compatibilidad v
 Sistema CBL 2
 conexión 51
 Sistema CBR
 compatibilidad v
 conexión 51
 Sistema de unidades, modo 15
 sistema operativo 151, 152
 Sistema operativo (SO)
 descarga 54
 sistema operativo, actualización 149, 150
 SO 149, 150
 Sobrescritura, modo (2nd) [INS] 14
 SocialSt (EstSoc (Estudios sociales)), categoría 25
 Software TI Connect 49
 solicitar, Request 261
 solución, deSolve() 205
 solve(), resolver 63, 276
 sombra, Shade 271
 SortA, clasificar en orden ascendente 278
 SortD, clasificar en orden descendente 279
 startTmr(), iniciar temporizador 279
 stdDev(), desviación estándar 280
 stdDevPop(), desvPbst() 280
 StoGDB, almacenar base de datos gráfica 281
 Stop, parar 281
 StoPic, almacenar imagen 281
 string(), expresión en cadena 281
 Style, estilo 282
 subMat(), submatriz 282
 submatriz, subMat() 282
 SUC (sucesión)
 modo 31
 sum(), sumar 262, 282
 suma
 Σ() 306
 acumulada, cumSum() 198
 sumar
 + 298
 sum() 262, 282
 Superficie oculta 80
 switch(), cambiar 283

T

T, transponer 283
 tabla de enlaces para transmisión 155
 tablas
 crear, Table 284
 definir, setTable() 269
 eliminar, ClrTable 190
 mostrar, DispTbl 208
 Table, crear tabla 284
 tan(), tangente 284
 tan⁻¹(), arco tangente 285
 tangente, tan() 284

- tanh(), tangente hiperbólica 285
 tanh⁻¹(), arco tangente hiperbólico 286
- Tapa
 plegar 4
 poner 4
 quitar 4
- taylor(), polinomio de Taylor 286
- tCollect(), agrupación trigonométrica 286
- Tecla (Almacenar) 14
- Tecla Almacenar (STO▶) 14
- Tecla de exponente (2nd [EE]) 12
- Tecla de modificador de diamante (◆)
 descripción 10
 estado 30
- Tecla de modificador de mano (☞)
 descripción 10
 estado 30
- Tecla de modificador Mayús (⇧)
 descripción 10
 estado 30
- Tecla de modificador secundaria (2nd)
 descripción 10
 estado 30
- Tecla de sustracción (⊖) 11
- Tecla negativa (⊖) 11
- Teclado
 mapa 8
 QWERTY 7
- Teclado numérico 11
 posición física 7
- Teclado QWERTY 7
- Teclas
 de función 7, 11
 de modificador 7, 10
 del cursor 7, 11
 otras 12
- Teclas de función (F1)–(F8)
 desplazamiento entre menús de barras de herramientas 42
 operaciones 11
 posición física 7
 selección de categorías 22
 selección de menús 38
- Teclas de modificador (2nd) (◆) (☞) 10
 estado 30
 posición física 7
- Teclas del cursor (⬅) (➡) (⬆) (⬇)
 abrir Apps 22
 función 11
 funciones adicionales 11
 introducción de órdenes 18
 posición física 7
 selección de pares entrada/respuesta 21
 uso con la tecla de mano 10
 uso del menú CARACTERES 8
 utilizadas con la tecla de mano 10
- Terminar
 programa, EndPrgm 96
- terminar
 barra de herramientas, EndTBar 289
 bucle, EndLoop 237
 función, EndFunc 220
 intentar, EndTry 289
 mientras, EndWhile 291
 para, EndFor 219
 personalizar, EndCustm 199
 programa, EndPrgm 252
 recuadro de diálogo, EndDlog 207
 si, EndIf 228
- tExpand(), expansión trigonométrica 287
- Text, texto 287
- texto, Text 287
- Then, entonces 228
- TI Connectivity Cable
 instalación de Apps v
- TI Connect, software v, 150
- Tildes
 menú CARACTERES 14
 órdenes del teclado 10
- timeCnv(), convertir hora 287
- Title, título 287
- título, Title 287
- tmpCnv(), conversión de la temperatura 288
- Toolbar, barra de herramientas 289

Trace, trazar 289
transferir error, PassErr 250
transponer, T 283
Trazar, Trace 115, 122, 124, 127
trazar, Trace 289
Try, intentar 289

U

una variable, estadísticas, OneVar 246
Unarchiv, desarchivar variables 170, 290
unidades
 definir, setUnits() 270
 obtener/devolver, getUnits() 226
Unidades personalizadas, modo 15
Unidades, modos 65
Unit System, modo 65
unitV(), vector fila o columna 290
Unlock, desbloquear 290

V

vaciar
 E/S, ClrIO 189
 gráfica, ClrGraph 189
 Home, ClrHome 189
Valor absoluto, abs() 127
valor absoluto, abs() 182
valor propio, eigVl() 212
Variables 31
 almacenamiento 162
 almacenar 14
 archivadas 31
 archivar y desarchivar 169
 archivar, Archive 170
 bloqueadas 31
 como archivos de App 23
 copiar 166
 copiar, CopyVar 166
 desarchivar, Unarchiv 170
 en aplicaciones 168
 pegar nombre 168
 recuperación 14
 VAR LINK 158, 159, 160, 161, 166, 170
variables
 archivar, Archive 184

borrado 144
copiar, CopyVar 191
deleting
 DelType 168
desarchivar, Unarchiv 290
eliminar, DelVar 204
locales, Local 235
transmisión 137, 139, 143
variance(), varianza 290
varianza, variance() 290
vector
 cilíndrico, mostrar como, ►Cylind 200
 esférico, mostrar como, ►Sphere 279
 propio, eigVc() 212
vectores
 fila o columna, unitV() 290
 mostrar vector cilíndrico, ►Cylind 200
 producto escalar, dotP() 209
 producto vectorial, crossP() 194

W

when(), cuando 84, 291
While, mientras 291
with, | 63, 309

X

x^{-1} , inverso 309
xor, or exclusivo booleano 292
XorPic, imagen con or exclusivo 292

Y

Y= editor 66, 69, 70

Z

zeros(), ceros 110, 292
zoom
 ajuste, ZoomFit 295
 almacenar, ZoomSto 297
 ampliar, ZoomIn 296
 cuadrado, ZoomSqr 297
 datos, ZoomData 295
 decimal, ZoomDec 295
 enteros, ZoomInt 296

recuadro, ZoomBox 294
recuperar, ZoomRcl 297
reducir, ZoomOut 296
trigonométrico, ZoomTrig 298
último, ZoomPrev 296
valores estándar, ZoomStd 297
ZoomBox, recuadro de zoom 294
ZoomData, datos de zoom 295
ZoomDec, decimal de zoom 295
ZoomFit, ajuste de zoom 295
ZoomIn, ampliar 296
ZoomInt, enteros con zoom 296
ZoomOut, reducir 296
ZoomPrev, último zoom 296
ZoomRcl, recuperar zoom 297
ZoomSqr, cuadrado de zoom 297
ZoomStd, valores estándar de zoom
297
ZoomSto, almacenar zoom 297
ZoomTrig, zoom trigonométrico 298

