TI-73 Explorer™
Graphing Calculator Guidebook

In this guidebook, TI-73 refers to both the TI-73 and TI-73 Explorer. All functions, instructions, and examples in this guidebook work identically for both the TI-73 and the TI-73 Explorer.

Important Information

Texas Instruments makes no warranty, either express or implied, including but not limited to any implied warranties of merchantability and fitness for a particular purpose, regarding any programs or book materials and makes such materials available solely on an "as-is" basis.

In no event shall Texas Instruments be liable to anyone for special, collateral, incidental, or consequential damages in connection with or arising out of the purchase or use of these materials, and the sole and exclusive liability of Texas Instruments, regardless of the form of action, shall not exceed the purchase price of this calculator. Moreover, Texas Instruments shall not be liable for any claim of any kind whatsoever against the use of these materials by any other party.

US FCC Information Concerning Radio Frequency Interference

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference with radio communications. However, there is no guarantee that interference will not occur in a particular installation.

If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, you can try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.

Consult the dealer or an experienced radio/television technician for help.

Caution: Any changes or modifications to this equipment not expressly approved by Texas Instruments may void your authority to operate the equipment.

This digital apparatus does not exceed the Class B limits for radio noise emissions from digital apparatus set out in the Radio Interference Regulations of the Canadian Department of Communications.

Table of Contents

1 Operating the TI-73

Preparing to Use Your TI-73

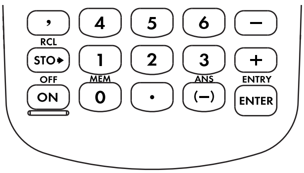
Before using your TI-73, you must install the batteries, turn on the calculator, and adjust the contrast. You may reset (clear) the calculator memory and defaults, if desired.

Installing the AAA Batteries

Install four AAA batteries in the battery compartment on the back of the calculator. Arrange the batteries according to the polarity (+ and -) diagram in the battery compartment. For more information about installing batteries, see Appendix C: Battery/Service and Warranty Information.

Turning the TI-73 On and Off

To turn on the TI-73, press [ON].



To turn off the TI-73 manually, press the yellow [2nd] key and then press [ON] (indicated in this book as [2nd] [OFF]).

When you press [2nd] [OFF], all settings and memory contents are retained by Constant Memory™. Any error condition is cleared. To prolong battery life, the APD™ (Automatic Power Down) feature turns the TI-73 off automatically after a few minutes without any activity.

When you turn on the calculator:

- The Home screen is displayed if you previously turned off the calculator by pressing [2nd] [OFF]. It appears as it did when you last used it; all errors are cleared.

  – or –

- The calculator displays the last screen (including the display, cursor, and any errors) that was displayed before Automatic Power Down turned off the calculator.

Adjusting the Display Contrast

The brightness and contrast of the display can depend on room lighting, battery freshness, and viewing angle.

To adjust the contrast:

1. Press and release the yellow [2nd] key.
2. Press and hold [▲] (to darken the screen) or [▼] (to lighten the screen).

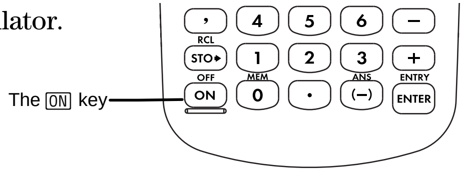
As you change the contrast setting, a number from 0 (lightest) to 9 (darkest) in the top-right corner indicates the current setting. You may not be able to see the number if the contrast is too light or too dark.

Resetting Memory and All Defaults

Follow these steps to reset the TI–73 to its factory settings and clear all memory:

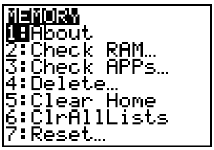
1. Turn on the calculator.

   [ON]



2. Display the **MEMORY** menu.

   [2nd] [MEM] (above [0])

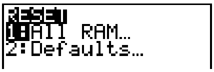


3. Display the **RESET** menu.

   7

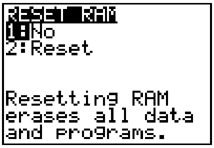
4. Display the **RESET RAM** menu.

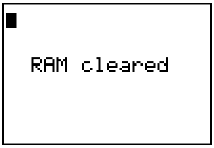
   1



5. Select **2:Reset**.

    2





All memory is cleared, and the calculator is reset to the factory settings.

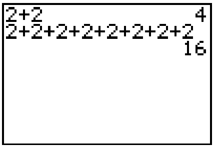
When you reset the TI-73, the display contrast is reset. To adjust the contrast, follow the directions in the previous section.

The Home Screen

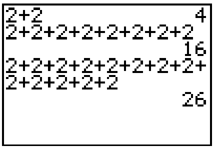
The Home screen is the primary screen of the TI-73. To go to the Home screen from any other screen or menu, press [2nd] [QUIT].

On the Home screen, you can enter instructions, functions, and expressions. The answers are displayed on the Home screen. The TI-73 screen can display a maximum of eight lines with a maximum of 16 characters per line.

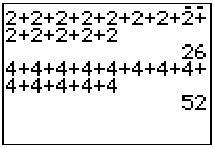
When you calculate an entry on the Home screen, depending upon space, the answer is displayed either directly to the right of the entry or on the right side of the next line.



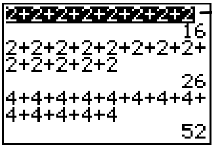
If an entry is longer than one line on the Home screen, it wraps to the beginning of the next line.



If all lines of the display are full, text scrolls off the top of the display. The TI–73 stores the previous entries as memory permits. See the section entitled, "Retrieving Previous Entries" on page 17.



You can scroll up with ▲ to see previously entered entries. If you press ENTER while a previous entry is highlighted (for example, 2+2+2+2+2+2+2+2), the calculator copies it to a new line below all entries (after 4+4+4… and its result, 52).



Scroll up with ▲.

To clear the Home screen, see the section entitled "Editing Numbers and Characters" on page 11.

Entering Numbers and Other Characters

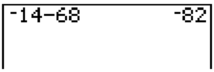
A symbol or abbreviation of each key's primary function is printed in white on the key. When you press that key, the function name is inserted at the cursor location.

Entering a Negative Number (–)

You enter a negative number with the negation key, (–). You can use negation to modify a number, expression, or each element in a list. Notice that this is different from the subtraction key, –, which CANNOT be used for negation.

Subtract ‾14 – 68.

2nd [QUIT] CLEAR
(–) **1 4** – **6 8** ENTER

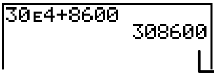


Entering a Number in Scientific Notation [2nd] [EE]

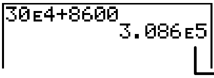
Using [2nd] [EE], you can enter a number in scientific notation. The notation used to display the result of a calculation depends upon the [MODE] setting (**Normal** or **Sci**). For more information on selecting modes, see the section in this chapter entitled “Mode Settings.”

Add 30 **E** 4 + 8600.

[2nd] [QUIT] [CLEAR] **3 0**
[2nd] [EE] [+] **8 6 0 0**
[ENTER]



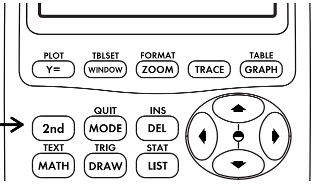
If in **Normal** mode



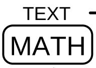
If in **Sci** mode

Entering Secondary Functions [2nd]

The [2nd] key



The secondary function of each key is printed in yellow above the key. When you press the yellow [2nd] key, the yellow character, abbreviation, or word above a key, becomes active for the next keystroke.



[2nd] [TEXT], which is the same as pressing [2nd] and then [MATH], displays the Text editor.

[MATH] displays four math menus.

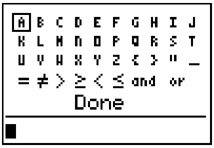
Entering Text [2nd] [TEXT]

Many examples in this manual require you to enter alphabetic characters, braces, a quotation mark, a space, or test operators. You can access all of these from the Text editor.

To exit the Text editor without saving the contents on the entry line, press [2nd] [QUIT], and the calculator returns you to the Home screen.

In all guidebook examples, when a character in the Text editor needs to be selected, the keystroke sequence shows the character followed by [ENTER]. Moving the Selection Cursor as necessary to highlight the character is implied. To exit the Text editor and display the contents on the entry line on the previous screen, select **Done**.

[2nd] [TEXT]



The entry line

Selection Cursor □	Highlights the character you want to select. Use the cursor keys ([▶], [◀], [▲], and [▼]) to move the cursor. **A** is highlighted with the selection cursor in the [2nd] [TEXT] example screen.
Letters **(A-Z)**	Lists letters A-Z in alphabetical order.
List Braces **{ }**	Surround a set of numbers separated by commas to create a list (outside of the List editor). For example, **{1,2,3}** on the Home screen is interpreted as a list.
Quotation Mark (**"**)	Surrounds the first text element in a categorical list or surrounds an attached list formula. (See Chapter 5: Lists for more information.)
Space (_)	Places a space between characters. It is frequently used in programs.
Test Operators **=, ≠, >, ≥, <, ≤**	Used to compare two values.
Logic (Boolean) Operators **and, or**	Used to interpret values as zero (false) or nonzero (true).

Done	Exits the Text editor when selected and pastes all contents on the entry line to the cursor location on the previous screen.
Entry Line	Displays all currently selected characters. All edit keys, except the cursor keys, edit characters on the entry line.

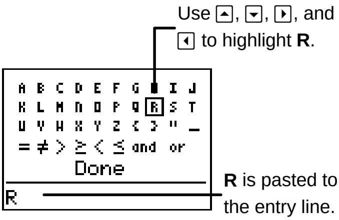
Insert R on the Home screen.

1. Go to the Home screen and clear it, if desired.

   [2nd] [QUIT] [CLEAR]

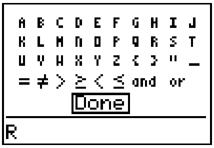
2. Use the Text editor to select **R**.

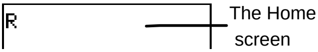
   [2nd] [TEXT] **R** [ENTER]



3. Exit the Text editor.

   Done [ENTER]





Test and Logic (Boolean) Operators

You select test and logic operators exactly as you would a letter. Both types of operators are explained in detail in Chapter 2: Math Operations.

Entry Line

The entry line displays all characters selected in the Text editor. The entry line also accepts all number keys ([1], [2], [3], …) and many keyboard operations ([x^2], [x], [+], [%], etc.). Enter these between Text editor characters, as necessary, without leaving the Text editor.

If you press a key that isn't accepted in the Text editor, the calculator does not return an error. You must select **Done** to exit and then continue your entry on the previous screen.

You can enter up to 16 characters on the entry line. If you need to enter more than 16 characters, select **Done** to save your entry. Then reenter the Text editor, and continue entering additional characters.

Common Display Cursors

In most cases, the appearance of the cursor indicates what happens when you press the next key or select the next menu item.

If you press [2nd] while the Insert Cursor (⇧) is displayed, the underline cursor becomes an underlined ↑.

Cursor	Appearance	Effect of Next Keystroke
Entry	Solid Rectangle ■	A character is entered at the cursor; any existing character is overwritten.
Insert	Underline _	A character is inserted in front of the cursor location.
Second	Reverse Arrow ⇧	A 2nd character (yellow on the keyboard) is entered, or a 2nd operation is executed.
Full	Checkerboard Rectangle ▦	No entry; the maximum characters are entered at a prompt, or memory is full.

Editing Numbers and Characters

Using the edit keys, you can edit an entry on the Home screen or Y= editor, programming commands in the Program editor, the entry lines of the Text editor and List editor, and constants in the Set Constant editor.

Keystrokes	Result
[◄] or [►]	Moves the cursor to the left or right. Moves the Selection cursor in the Text editor.
[▲] or [▼]	Moves/scrolls the cursor up or down.
[2nd] [◄]	Moves the cursor to the beginning of an entry.
[2nd] [►]	Moves the cursor to the end of an entry.
[CLEAR]	• Within a line on the Home screen, it clears all characters to the right of the cursor. • At the beginning or end of a line on the Home screen, it clears the current line. • On a blank line on the Home screen, it clears everything on the Home screen. • In an editor, it clears the expression or value where the cursor is located.
[DEL]	Deletes the character at the cursor.
[2nd] [INS]	Inserts characters in front of a character; to end insertion, press [2nd] [INS] or press [◄], [►], [▲] or [▼].
[UNIT]	Inserts a character before a fraction on the Home screen. ([2nd] [INS] inserts a character before a fraction on any other screen.)
[x]	Inserts the variable X at the cursor location.

Functions and Instructions

A *function* returns a value. Generally, the first letter of each function is *lowercase* on the TI-73. For example, **pxl-Test(** is a function because it returns a value, **0** or **1**.

An *instruction* initiates an action. Generally, the first letter of each instruction name is *uppercase*. For example, **Pxl-On(** is an instruction that draws a pixel on the graph screen.

Most functions and some instructions take at least one argument. An open parenthesis (**(**) at the end of the function or instruction name prompts you to enter an argument. Complete the function with an end parenthesis, [)].

Note: Do not use the Text editor to enter names of functions or instructions. For example, you cannot enter **L**, then **O**, then **G** to calculate the log of a value. If you did this, the calculator would interpret the entry as implied multiplication of the variables **L**, **O**, and **G**.

When this guidebook describes the syntax of a function or instruction, each argument is in italics. Optional arguments for a function are signified by brackets [ ]. Do not enter the brackets.

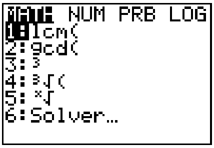
Accessing Functions and Instructions from Menus

You can find most functions and instructions on menus (in other words, not directly from the keyboard).

Displaying a Menu

To display a menu, press the key associated with the menu. Up to four separate menus are displayed from which you choose the menu item you want.

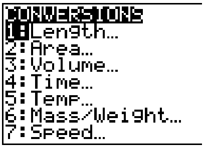
[MATH]



To move from menu to menu on a menu screen, press ▶ or ◀ until the menu name is highlighted.

When a menu item ends in an ellipsis (…), the item displays a secondary menu or editor when you select it.

2nd [CONVERT]



Accessing and Selecting Menu Items

To scroll up or down the menu items, press ▲ or ▼. To wrap to the last menu item directly from the first menu item, press ▲. To wrap to the first menu item directly from the last menu item, press ▼.

When the menu continues beyond the displayed items, a ↓ replaces the colon next to the last displayed item.

Select a menu item in one of two ways:

- Press ▼ or ▲ to move the cursor to the number or letter of the item, and then press ENTER.
- Press the number key for the number next to the item. If a letter is next to the item, access letters from the Text editor (2nd [TEXT]).

After you select an item from a menu, you usually are returned to the initial screen where you were working.

Exiting a Menu without Making a Selection

Exit a menu without making a selection in one of three ways:

- Press CLEAR to return to the screen where you were.
- Press 2nd [QUIT] to return to the Home screen.
- Press a key or key sequence for another menu or for another screen (except 2nd [TEXT], which is not accessible from all screens).

Accessing Functions/Instructions from the CATALOG

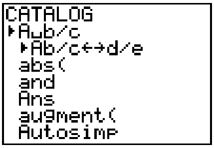
[2nd] [CATALOG] displays the **CATALOG**, which is an alphabetical list of all functions, instructions, programming commands, variables, and symbols on the TI-73. If, for example, you cannot remember where a particular menu item is located, you can find it in the **CATALOG**.

Items that begin with a number are in alphabetical order according to the first letter after the number. For example, **1-Var Stats** is among the items that begin with **V**.

Items that are symbols follow the last item that begins with **Z**. You can access the symbols quickly by pressing [▲] from the first catalog item, **A␣b/c**. The cursor moves to the bottom of the list.

[2nd] [CATALOG]

Selection Cursor



To select an item from the **CATALOG**:

1. Press [2nd] [CATALOG] to display the **CATALOG**. The Selection Cursor always points to the first item.

2. Press [▲] or [▼] to scroll the **CATALOG** until the Selection Cursor points to the item you want.

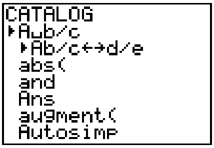
   To jump to the first item beginning with a particular letter, select that letter from the Text editor. Press [2nd] [TEXT] while in the **CATALOG**, use the cursor keys to highlight the letter you want, and then press [ENTER]. You are automatically returned to the **CATALOG**, and the Selection Cursor has now moved to the new section. Scroll to the item you want.

3. Press [ENTER] to paste the **CATALOG** item to the current screen.

Enter the **CATALOG** and go directly to the section starting with L.

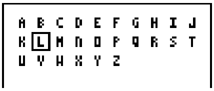
1. Go to the **CATALOG.**

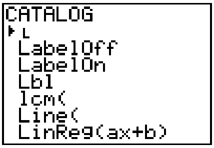
   [2nd] [CATALOG]



2. Select **L** from Text editor.

   [2nd] [TEXT] **L** [ENTER]





Selecting **L** by pressing [ENTER] pastes it to the previous screen, just as if you had selected it from a menu.

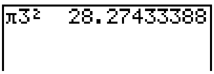
Entering Expressions

An *expression* is a group of numbers, variables, functions and their arguments, or a combination of these elements that evaluates to a single answer. Instructions cannot be used in expressions. An expression is completed when you press [ENTER], regardless of the cursor location.

On the TI-73, you enter an expression in the same order as you would write it on paper. The entire expression is evaluated according to the Equation Operating System (EOS™) rules (which is explained in detail in Appendix B: Reference Information), and the answer is displayed.

Calculate the area (A) of a circle whose radius (R)=3 using the formula $A=\pi R^2$. Then use the area to calculate the volume (V) of a cylinder whose height (H)=4. Use the formula $V=A \times H$.

[2nd] [QUIT] [CLEAR]
[2nd] [π] **3** [x^2] [ENTER]



[×] 4 [ENTER]

```
π3²   28.27433388
Ans*4
      113.0973355
```

Ans is the previous answer, 28.27433388.

Grouping Parts of Expressions with Parentheses

The calculator calculates an expression within parentheses first.

Calculate 4(1+2).

[2nd] [QUIT] [CLEAR]
4 [(] 1 [+] 2 [)] [ENTER]

```
4(1+2)           12
```

4(1+2)=4(3)=12

Using Implied Multiplication in Expressions

The calculator understands that two numbers separated by parentheses are multiplied together.

Calculate 4×3 using parentheses.

[2nd] [QUIT] [CLEAR]
4 [(] 3 [)] [ENTER]

```
4(3)             12
```

Entering Multiple Expressions on One Line

To store more than one expression on a line, separate two or more expressions or instructions on a line with a colon ([2nd] [CATALOG] [▲] [▲] [▲] [ENTER]).

Define the variable, R=5, and then calculate πR^2 on the same line.

1. Store 5 to **R**.

   [2nd] [QUIT] [CLEAR]
   5 [STO▸] [2nd] [TEXT]
   R [ENTER] **Done** [ENTER]
   [2nd] [CATALOG]
   [▲] [▲] [▲] [ENTER]

```
5→R:
```

2. Enter the second expression, π**R²**, and calculate the result.

   [2nd] [π] [2nd] [TEXT]
   R [ENTER] **Done** [ENTER]
   [x^2] [ENTER]

```
5→R:πR²
        78.53981634
```

Retrieving Previous Entries [2nd] [ENTRY]

When you press [ENTER] on the Home screen to evaluate an expression or execute an instruction, the expression or instruction is placed in a storage area called **Entry** (last entry). When you turn off the TI-73, **Entry** is retained in memory.

You can retrieve the last entry to the current cursor location, where you can edit it, if desired, and then execute it. On the Home screen or in an editor, press [2nd] [ENTRY]; the current line is cleared and the last entry is pasted to the line.

The TI-73 retains as many previous entries as memory permits. To cycle through these entries, press [2nd] [ENTRY] repeatedly. To view stored entries, use [▲] to scroll up the Home screen.

Store 1 to variable A, 1 to variable B, and then 3 to variable A using [2nd] [ENTRY].

1. Store 1 to **A**.

   [2nd] [QUIT] [CLEAR]
   1 [STO▸] [2nd] [TEXT]
   A [ENTER] **Done** [ENTER]
   [ENTER]

```
1→A                  1
```

2. Recall the last entry.

   [2nd] [ENTRY]

```
1→A                  1
1→A
```

3. Edit and enter the new expression.

   [◄] [2nd] [TEXT]
   B [ENTER] **Done** [ENTER]
   [ENTER]

4. Cycle back two entries.

   [2nd] [ENTRY] [2nd] [ENTRY]

5. Edit and enter the new expression.

   [◄] [◄] [◄] **3** [ENTER]

Recalling and Storing the Last Answer [2nd] [ANS]

When an expression is evaluated successfully from the Home screen or from a program, the TI–73 stores the answer to a system variable called **Ans** (last answer). Recall **Ans** by pressing [2nd] [ANS]. **Ans** can be a real number or a list. When you turn off the TI–73, the value in **Ans** is retained in memory.

You can use the variable **Ans** in any place that is appropriate for the type of answer **Ans** represents. For example, if **Ans** is a real number, you can use it anywhere where real numbers are accepted (Y= editor, **WINDOW**, List editor, etc.).

Continuing a Calculation with Ans

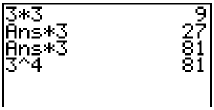
You can recall **Ans** as the first entry in the next expression without entering the value again or pressing [2nd] [ANS]. After completing a calculation, press an operation or function key (excluding [UNIT], [b/c], or [CONST]) and the calculator displays **Ans** and uses the value in the next calculation.

1. Calculate 3^4 using the **Ans** feature.

   [2nd] [QUIT] [CLEAR]
   3 [×] **3** [ENTER]
   [×] **3** [ENTER]
   [×] **3** [ENTER]

2. Check your answer, if desired.

   3 [^] **4** [ENTER]



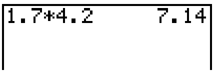
Using Ans as a Variable in an Expression

Since **Ans** is a variable, you can use it in expressions just as you would any other variable. When the expression is evaluated, the TI-73 uses the value of **Ans** in the calculation. For more information about variables, see the next two sections in this chapter entitled, "Storing Values to a Variable" and "Recalling Variable Values."

Calculate the area of a garden plot 1.7 meters by 4.2 meters. Then calculate the yield per square meter if the plot produces a total of 147 tomatoes.

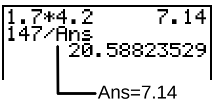
1. Calculate the area.

   [2nd] [QUIT] [CLEAR]
   1.7 [×] **4.2** [ENTER]



2. Divide 147 by **Ans**, which was calculated in the first step.

   147 [÷] [2nd] [ANS] [ENTER]



Storing Values to a Variable [STO▸]

You can store values or expressions that result in one value or lists to a one-letter variable or a system variable (types are listed below) to save for later use. Also, you can save a result for later use by storing **Ans** to a variable before you evaluate another expression.

When an expression containing the name of a variable is evaluated, the value of the variable at that time is used. You can enter and use several types of data for variables, including real numbers, lists, functions, statistical plots, and graph pictures.

Variable names cannot be the same as a name that is preassigned by the TI–73. These include built-in functions such as **abs(** , instructions such as **Line(** , and system variables such as **Xmin**.

Variable Type	Names
Real Numbers	**A**, **B**, …, **Z** ([2nd] [TEXT])
Lists—Numerical and Categorical	**L1**, **L2**, **L3**, **L4**, **L5**, **L6**, and any user-defined list names ([2nd] [STAT] **Ls**)
Functions	**Y_1**, **Y_2**, **Y_3**, **Y_4** ([2nd] [VARS] **2:Y-Vars**)
Stat Plots	**Plot1**, **Plot2**, **Plot3** ([2nd] [PLOT] from the Program editor)
Graph Pictures	**Pic1**, **Pic2**, **Pic3** ([2nd] [VARS] **4:Picture**)
System Variables	**Xmin**, **Xmax**, … ([2nd] [VARS] **1:Window**)

Store a value to either a system variable or a letter variable from the Home screen or a program using the [STO▸] key. Begin on a blank line and follow these steps.

1. Enter the numeric value. It can be an expression that results in a numeric value.
2. Press [STO▸]. → is copied to the cursor location.
3. Select the type of variable to which you want to store the value. Use the Text editor ([2nd] [TEXT]) to enter a letter variable, the **VARS** ([2nd] [VARS]) menu to enter a system variable, or the [2nd] [STAT] **Ls** menu to enter a list name.
4. Press [ENTER]. If you entered an expression, it is evaluated. The value is stored to the variable.

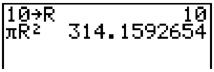
Store 10 to R, and then calculate πR^2.

1. On the Home screen, store 10 to **R**.

   [2nd] [QUIT] [CLEAR]
   1 0 [STO▸] [2nd] [TEXT]
   R [ENTER] **Done** [ENTER]
   [ENTER]

2. Calculate π**R²**.

   [2nd] [π] [2nd] [TEXT]
   R [ENTER] **Done** [ENTER]
   [x^2] [ENTER]



Recalling Variable Values [2nd] [RCL]

To recall a variable's value to the current cursor location, follow these steps. To leave **Rcl**, press [CLEAR].

1. Press [2nd] [RCL]. **Rcl** and the edit cursor are displayed on the bottom line of the screen.
2. Enter the name of the variable in any of four ways:
   - Enter letters using the Text editor ([2nd] [TEXT]).
   - Press [2nd] [STAT], and then select the name of the list from the **Ls** menu.

- Press [2nd] [VARS] to display the **VARS** menu; next select the type and then the name of the variable or function.
- Press [PRGM] (from the Program editor only), and then select the name of the program to call a program as a subroutine within another program.

3. The variable name you selected is displayed on the bottom line and the cursor disappears.
4. Press [ENTER]. The variable contents are inserted where the cursor was located before you began these steps.

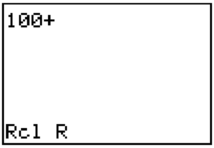
Calculate 100+R using the **Rcl** function. (R was defined in the previous section, “Storing Variable Values.”)

1. On the Home screen, enter the first part of the calculation.

   [2nd] [QUIT] [CLEAR] **1 0 0** [+]

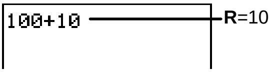
2. Recall **R**.

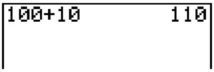
   [2nd] [RCL] [2nd] [TEXT]
   R [ENTER] **Done** [ENTER]



3. Finish the calculation.

   [ENTER]
   [ENTER]



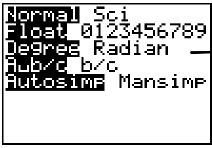


Mode Settings

Mode settings control how the TI-73 interprets and displays numbers. Mode settings are retained by the Constant Memory feature when the TI-73 is turned off. All numbers, including elements of lists, are displayed according to the current mode settings, as applicable. To display the mode settings, press [MODE]. The default settings are highlighted on the following screen.

To select a mode setting, highlight the one you want by using the cursor keys, and then press [ENTER].

[MODE]



Default Settings

Normal Sci	Numeric Notation mode
Float 0123456789	Decimal Notation mode
Degree Radian	Angle mode
A␣b/c b/c	Display Format mode (fractions only)
Autosimp Mansimp	Simplification mode (fractions only)

Numeric Notation Mode

The Numeric Notation mode settings affect the way an answer is displayed on the TI-73. Numeric answers can be displayed with up to 10 digits and a two-digit exponent. Answers (excluding fractional ones) on the Home screen, list elements in the List editor, and table elements on the Table screen are displayed according to the Numeric Notation mode selected.

The **Normal** setting displays results with digits to the left and right of the decimal, as in **123456.78**.

The **Sci** (scientific) setting expresses numbers with one digit to the left of the decimal and the appropriate power of 10 to the right of **E**, as in **1.2345678 E 5**, (which is the same as **123456.78**).

Note: Answers that have more than 10 digits and whose absolute value is greater than .001 are displayed in scientific notation.

Decimal Notation Mode

The Decimal Notation mode has two settings, **Float** and **0123456789**, which only affect the way an answer is displayed on the TI–73.

The **Float** (floating decimal point) setting displays up to 10 digits, plus the sign and decimal.

The **0123456789** (fixed decimal point) setting specifies the number of digits (**0** through **9**) to display to the right of the decimal. Place the cursor on the desired number of decimal digits, and then press [ENTER].

The decimal setting applies to answers (excluding fractional ones) on the Home screen, *X*- and *Y*-coordinates on a graph display, list elements in the List editor, table elements on the Table screen, and regression model results.

Angle Mode

The Angle mode has two settings, **Degree** and **Radian**, which control how the TI–73 interprets angle values in trigonometric functions. See Chapter 11: Trigonometry for a detailed explanation.

Display Format Mode

The Display Format mode has two settings, **A␣b/c** and **b/c**, which determine if a fraction is displayed as a mixed or simple fraction. See Chapter 3: Fractions for a detailed explanation.

Simplification Mode

The Simplification mode has two settings, **Autosimp** and **Mansimp**, which determine if the calculator automatically simplifies a fractional result completely or if you must simplify the results manually step-by-step. See Chapter 3: Fractions for detailed information.

2 Math Operations

Keyboard Math Operations

The following sections explain how to use the math functions, including [2nd] functions, found on the TI-73 keyboard. All of the examples in these sections assume that you are on the Home screen and that defaults are selected (unless specified otherwise).

Real numbers include fractions unless specified otherwise.

Basic Operations [+], [−], [×], [÷]

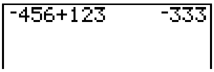
Returns the sum ([+]), difference ([−]), product ([×]), or quotient ([÷]) of *valueA* and *valueB*, which can be real numbers, expressions, or lists.

If both *values* are lists, they must have the same number of elements. If one *value* is a list and the other is a non-list, the non-list is paired with each element of the list, and a list is returned.

valueA + *valueB*
valueA − *valueB*
valueA ∗ *valueB*
valueA ÷ *valueB*

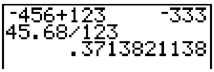
Add ⁻456 + 123.

[CLEAR] [(-)] **456** [+] **123**
[ENTER]



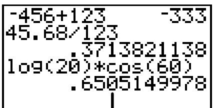
Divide 45.68 ÷ 123.

45.68 [÷] **123** [ENTER]



Multiply log(20) × cos(60).

[MATH] [▸] [▸] [▸] **1**
20 [)] [×] [2nd] [TRIG] **3**
60 [)] [ENTER]

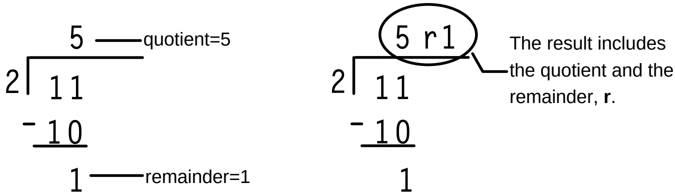


In **Degree** mode

Integer Division [2nd] [INT÷]

[2nd] [INT÷] divides two positive integers and displays the quotient and the remainder, **r**.

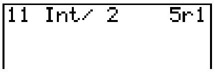
posintegerA **Int/** *posintegerB*



You can include integer division in an expression, but the remainder may not be displayed as part of the final answer.

After a calculation with [2nd] [INT÷] is completed, only the quotient from the result is stored in **Ans** (last answer). Therefore, if you use the result in another calculation, the remainder is ignored.

Calculate 11 ÷ 2 using integer division.

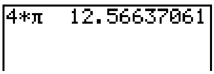


[CLEAR] **1 1** [2nd] [INT÷] **2** [ENTER]

π [2nd] [π]

Represents the value for the constant, π, in calculations. The calculator uses π=3.1415926535898, although it only displays 3.141592654 on the screen. π acts as a real number in any calculation.

Multiply 4 × π.



[CLEAR] **4** [×] [2nd] [π] [ENTER]

Calculate sin(π).

[CLEAR] [2nd] [TRIG] **1** [2nd] [π] [)] [ENTER]

If in **Radian** mode

If in **Degree** mode

Percent [%]

Changes a *real_number* to percent. Results display according to the Decimal Notation mode setting.

real_number%

Convert ⁻30.6% to a decimal.

1. Select Float Decimal setting.

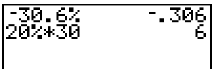
   [MODE] [▼] [ENTER] [2nd] [QUIT]

2. Convert ⁻30.6% to a decimal.

   [CLEAR] [(-)] **3 0 . 6** [%] [ENTER]

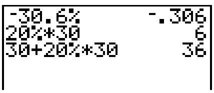
Calculate 20% of 30.

2 0 [%] [×] **3 0** [ENTER]



Calculate 30 + 20% of 30.

3 0 [+] **2 0** [%] [×] **3 0** [ENTER]



Inverse Function [2nd] [x^{-1}]

Returns the inverse, x^{-1}, of *value*, which is the equivalent of the reciprocal, $^1/_x$, of a real number, expression, or each element in a list.

value $^{-1}$

Important: To ensure that results are displayed as simple fractions instead of mixed numbers, select **b/c** Display Format mode.

Calculate $^5/_8{}^{-1}$.

[CLEAR] **5** [b/c] **8** [▶] [2nd] [x^{-1}] [ENTER]

Calculate ⁻2.5^{-1}.

[(–)] **2 . 5** [2nd] [x^{-1}] [ENTER]

Square [x^2]

Finds the square of a real number, an expression, or each element in a list. **Note**: Using parentheses with [x^2] ensures that you get the correct answer. Refer to Appendix B: Reference Information for Equation Operating System (EOS) calculation rules.

value^2

Calculate 5^2.

[CLEAR] **5** [x^2] [ENTER]

Compare the results of ⁻5^2 and (⁻5)^2.

1. Calculate ⁻5^2.

   [(–)] **5** [x^2] [ENTER]

2. Calculate $(^-5)^2$.

   [(] [(-)] **5** [)] [x^2] [ENTER]

Power [^]

Raises *value* to any *power*. *value* and *power* can be real numbers, expression, or lists. If both are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is paired with each element of the list, and a list is returned.

value^*power*

value is limited by mathematical rules. For example, ($^-$4)^.5 results in an error because this is the equivalent of ($^-$4)^$^1/_2$, which is $\sqrt{^-4}$, a complex number.

Calculate 2^5.

[CLEAR] **2** [^] **5** [ENTER]

Square Root [2nd] [√]

Calculates the square root of *value*, which can be a positive real number, an expression that results in a positive real number, or a list of positive numbers.

√(*value*)

Calculate $\sqrt{256}$.

[CLEAR] [2nd] [√] **2 5 6** [)]
[ENTER]

Test Operations 2nd [TEXT]

The two types of test operations included in the Text editor are relational operators (**=**, ≠, **>**, ≥, **<**, and ≤) and logic (Boolean) operators (**and** and **or**).

Both relational and logic operators often are used in programs to control program flow and in graphing to control the graph as a function over specific values.

Relational Operators

Relational operators compare *conditionA* and *conditionB* and return **1** if the conditional statement is true. They return **0** if the conditional statement is false. *conditionA* and *conditionB* can be real numbers, expressions, or lists.

If both *conditions* are lists, they must have the same number of elements. If one *condition* is a list and the other a non-list, the non-list is compared with each element of the list, and a list is returned.

Test operations are frequently used in programs.

conditionA **relational_operator** *conditionB*

Relational operators are evaluated after mathematical functions according to EOS rules (Appendix B: Reference Information). Therefore, for 2+2=2+3, the TI-73 returns **0**. It compares 4 with 5 and returns 0, because the operation is false. For 2+(2=2)+3, the TI-73 returns **6**. The relational test in parentheses returns 1, because the operation is true. Then it adds 2+(1)+3.

Operator:	Returns true (1) if:
= (equal)	Two conditions are equal.
≠ (not equal to)	Two conditions are not equal.
> (greater than)	*conditionA* is greater than *conditionB*.
≥ (greater than or equal to)	*conditionA* is greater than or equal to *conditionB*.
< (less than)	*conditionA* is less than *conditionB*.
≤ (less than or equal to)	*conditionA* is less than or equal to *conditionB*.

Logic (Boolean) Operators

Logic (Boolean) operators compare *conditionA* and *conditionB* and return **1** if the conditional statement is true. They return **0** if the conditional statement is false. *conditionA* and *conditionB* can be real numbers, expressions, or lists.

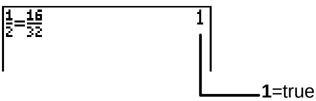
If both *conditions* are lists, they must have the same number of elements. If one *condition* is a list and the other a non-list, the non-list is compared with each element of the list, and a list is returned.

> *conditionA* **and** *conditionB*
> *conditionA* **or** *conditionB*

Operator:	Returns true (1) if:
and	Both conditions are nonzero.
or	At least one condition is nonzero.

Test $1/2 = 16/32$.

[2nd] [QUIT]
[CLEAR] **1** [b/c] **2** [▸]
[2nd] [TEXT] **=** [ENTER] **Done**
[ENTER] **1 6** [b/c] **3 2** [ENTER]



For **L1={1,2,3}**, test **L1>log(30)**.

1. Define **L1**.

   [CLEAR] [2nd] [TEXT]
   { [ENTER] **1** [,] **2** [,] **3 }**
   [ENTER] **Done** [ENTER]
   [STO▸] [2nd] [STAT] **1** [ENTER]

```
{1,2,3}→L1
           {1 2 3}
```

2. Test **L1 > log(30)**.

   [2nd] [STAT] **1** [2nd] [TEXT]
   > [ENTER] **Done** [ENTER]
   [MATH] [◄] **1** **3 0** [)] [ENTER]

```
{1,2,3}→L1
           {1 2 3}
L1>log(30)
           {0 1 1}
```

1>log(30) is false;
2>log(30) is true;
3>log(30) is true.

Test **cos(90) and sin(0)**.

[CLEAR] [2nd] [TRIG] **3**
9 0 [)] [2nd] [TEXT] **and**
[ENTER] **Done** [ENTER]
[2nd] [TRIG] **1** **0** [)] [ENTER]

```
cos(90) and sin(
0)
                0
```

The [MATH] MATH Menu

The [MATH] **MATH** menu includes various math functions.

[MATH]	`MATH NUM PRB LOG` `1:lcm(` `2:gcd(` `3:³` `4:³√(` `5:ˣ√` `6:Solver…`

1:lcm(	Finds the least common multiple, which is the smallest number that two integers can divide into evenly.
2:gcd(	Finds the greatest common divisor, which is the largest number that divides into two integers evenly.
3:^{3}	Calculates the cube.
4:$^3\sqrt{}$(	Calculates the cube root.
5:$^x\sqrt{}$	Calculates the x^{th} root.
6:Solver…	Displays the Equation Solver.

lcm([MATH] 1

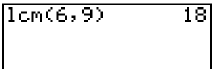
The least common multiple (LCM) function returns the smallest number that two positive whole numbers can divide into evenly, of two positive whole numbers or lists of positive whole numbers. If both arguments are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is paired with each element of the list, and a list is returned.

lcm(is frequently used with fractions to find a common denominator. See Chapter 3: Fractions for more information on entering fractions.

lcm(_valueA_,_valueB_**)**

Find the LCM of 6 and 9.

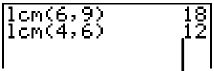
[CLEAR] [MATH] **1**
6 [,] **9** [)] [ENTER]



Add $^1/_4 + {}^5/_6$ (using LCM).

1. Find the LCM of the denominators.

   [MATH] **1**
   4 [,] **6** [)] [ENTER]



Therefore, 12 is the common denominator.

2. Use the LCM to convert $^1/_4$ and $^5/_6$ to fractions where 12 is the common denominator (without using the calculator).

$$\frac{1}{4} \times \frac{3}{3} = \frac{3}{12}$$

$$\frac{5}{6} \times \frac{2}{2} = \frac{10}{12}$$

LCM=12

3. Add the newly converted fractions (without using the calculator).

$$\frac{3}{12} + \frac{10}{12} = \frac{13}{12}$$

4. Verify your answer by adding the original fractions on the calculator. Select the **b/c** Display Format mode setting and clear the Home screen, if desired.

   [MODE] [▼] [▼] [▼] [▶] [ENTER]
   [2nd] [QUIT] [CLEAR]
   1 [b/c] **4** [▶] [+] **5** [b/c] **6**
   [ENTER]

gcd([MATH] 2

The greatest common divisor (GCD) function returns the largest number that divides into two positive whole numbers or lists of positive whole numbers evenly. If both arguments are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is paired with each element of the list, and a list is returned.

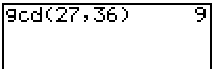
This is frequently used with fractions to reduce them to lowest terms. See Chapter 2: Fractions for more information on entering fractions.

gcd(*valueA*,*valueB***)**

Find the greatest common divisor for the fraction, $^{27}/_{36}$.

1. Find the GCD of $^{27}/_{36}$.

   $\boxed{\text{MATH}}$ **2**
   2 7 $\boxed{,}$ **3 6** $\boxed{)}$ $\boxed{\text{ENTER}}$



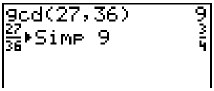
2. Simplify the fraction completely using the GCD (without using the calculator).

GCD=9

$$\frac{27}{36} \div \frac{9}{9} = \frac{3}{4}$$

3. Verify your answer by simplifying $^{27}/_{36}$ by 9 on the calculator. You must be in **Mansimp** mode setting.

   $\boxed{\text{MODE}}$ $\boxed{\blacktriangledown}$ $\boxed{\blacktriangledown}$ $\boxed{\blacktriangledown}$ $\boxed{\blacktriangledown}$ $\boxed{\blacktriangleright}$
   $\boxed{\text{ENTER}}$ $\boxed{\text{2nd}}$ [QUIT]
   2 7 $\boxed{b/c}$ **3 6** $\boxed{\blacktriangleright}$ $\boxed{\text{SIMP}}$ **9**
   $\boxed{\text{ENTER}}$



3 $\boxed{\text{MATH}}$ 3

Calculates the cube of n, which is equivalent to $n \times n \times n$ of any real number, expression, or each element in a list.

n^3

Calculate 5^3.

5 $\boxed{\text{MATH}}$ **3**
$\boxed{\text{ENTER}}$

$5^3 = 5 \times 5 \times 5 = 125$

$^3\sqrt{}$([MATH] 4

Calculates the cube root of *value*, which is equivalent to *n* where n^3=*value*. *value* can be a real number, expression, or list.

```
For n³=value, ³√value=n
```

3√(*value***)**

Calculate $^3\sqrt{}(125)$.

[MATH] **4** **1 2 5** [)] [ENTER]

$^3\sqrt{}$(125) = 5
because 5^3=125

$^x\sqrt{}$ [MATH] 5

Calculates the x^{th} root of *value*, which is equivalent to *n* where n^x=*value*. *value* can be a real number, expression, or list. *x* can be any real number.

```
For nˣ=value, ˣ√value=n
```

x **ˣ√(***value***)**

Calculate $^6\sqrt{}64$.

[CLEAR] **6** [MATH] **5**
6 4 [ENTER]

$6^x\sqrt{}64$ = 2
because 2^6=64

Solver [MATH] 6

The Equation Solver allows you to solve for one unknown one-letter variable in an equation containing up to 5 one-letter variables. By default, the equation is assumed to be equal to 0; however, you can set the equation equal to any real number (or an expression that results in a real number).

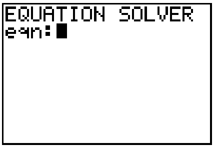
The screen you see when you select **Solver** depends on whether an equation has been defined previously.

To exit Solver and return to the Home screen, press [2nd] [QUIT].

The EQUATION SOLVER Screen

If no equation is currently defined, pressing [MATH] **6** takes you to the **EQUATION SOLVER** screen. Enter the equation at the cursor, using the Text editor ([2nd] [TEXT]) to enter the variable names.

[MATH] **6**



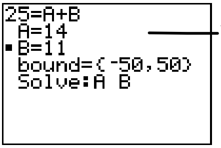
You can have more than one variable on each side of the equation. For example, **A + B = B + D + E**.

If you do not set the equation equal to a value, the calculator automatically sets it equal to 0. For example, to enter A+B=0, just enter **A+B** and press [ENTER]. You are limited to 5 variables per equation.

The Equation Variables Screen

If an equation has been defined previously, pressing [MATH] **6** takes you to the Equation Variables screen.

[MATH] **6**



Your screen may vary.

Equation	Displays the currently defined equation.
Equation Variables	Displays all equation variables and their values.
bound Default={⁻1E99,1E99}	Displays the **bound** limits that apply to the unknown variable value for which you are solving.
Solve	You select one variable, the one you want to solve for, from this list.

Equation

The first line of the Equation Variables screen displays the equation you defined on the **EQUATION SOLVER** screen.

If you would like to edit a defined equation, press [▲] until the **EQUATION SOLVER** screen is displayed. Edit the equation with [CLEAR], [DEL], or [2nd] [INS], as necessary. Then press [ENTER] to return to the Equation Variables screen.

Equation Variables

All variables included in the defined equation are displayed. If those variables have never been assigned a value, they are set equal to 0. If a variable has been defined previously (for example, from the Home screen), that value appears.

If a value extends beyond the screen, press [▶] to scroll to the end of the number. This is especially important if a number is in scientific notation and you need to see whether it has a negative or positive exponent.

For an equation with more than one variable, you must define all variables except the unknown variable for which you want to solve.

bound

bound limits apply to the unknown variable value for which you are solving. Default bounds are {⁻1E99,1E99}. Use these limits to narrow the unknown value solution to a specific range of numbers, especially if more than one answer exists.

Hint: For answers with many solutions (for example, trig functions), consider graphing the function first to get an idea of the most ideal (or specific) **bound** limits.

Solve

Specify the unknown variable from the **Solve** line. This prompts the calculator to solve for it.

To select a variable on the **Solve** line, highlight the unknown variable with the cursor, and then press [ENTER]. After you press [ENTER], a solid black square appears next to the solved (previously unknown) variable displayed in the Equation Variables section.

Hint: The Solver allows for a small tolerance when solving a result, which is noticeable especially when solving complex equations or those with multiple solutions. For example, a result of 3.99999999999999 (instead of 4) for the equation **16=x²** is considered a correct answer.

Solving Equations with Only One Possible Answer

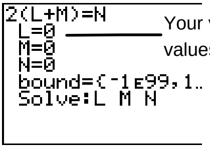
For $2(L+M)=N$, solve for L when N=268, and M=40, ⁻14, and 307.

1. Define the equation on the **EQUATION SOLVER** screen.

   [MATH] **6**
   [▲] [CLEAR] (if necessary)

2. Enter the equation.

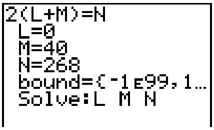
   [2nd] [TEXT]
   2 [(] **L** [ENTER] [+] **M** [ENTER]
   [)] **=** [ENTER] **N** [ENTER]
   Done [ENTER] [ENTER]



Your variable values may vary.

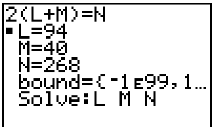
3. Enter the first value for **M**, 40, and **N**, 268.

   [▼] **4 0** [▼] **2 6 8**



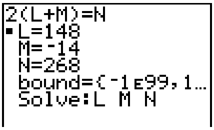
4. Solve for **L**.

   [▼] [▼] [ENTER]



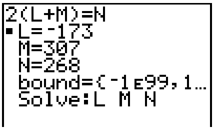
5. Solve for **L** when **M**=⁻14.

   [▼] [(-)] **1 4** [▼] [▼] [▼] [ENTER]



6. Solve for **L** when **M**=307.

   [▼] **3 0 7** [▼] [▼] [▼] [ENTER]



Solving Equations with More Than One Answer

The calculator only returns one solution even if more than one possible solution exists. When this is the case, you can first enter a guess by assigning a value to that variable and then asking the calculator to solve your equation. The TI-73 always chooses the solution closest to that guess. However, the guess must be within the bound limits; otherwise, you get an error.

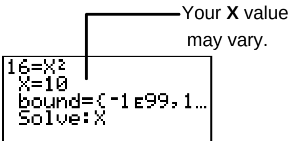
Find the negative solution to the equation, $16 = X^2$.

1. Define the equation on the **EQUATION SOLVER** screen.

   [MATH] **6**
   [▲] [CLEAR] (if necessary)

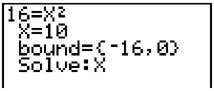
2. Enter the equation.

   [2nd] [TEXT]
   1 6 = [ENTER] [x] [x²] **Done**
   [ENTER] [ENTER]



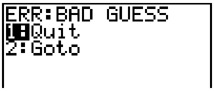
3. Use **bound** to limit your answer to a negative one (between ⁻16 and 0).

   [▼] [▶] [(–)] **16** [DEL] [DEL] [▶] **0**
   [DEL] [DEL] [DEL]



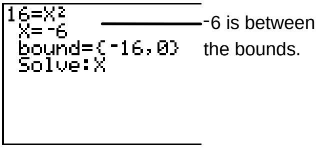
4. Solve for **X**.

   [▼] [ENTER]



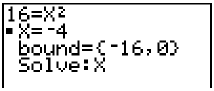
5. The guess, **X=10**, is not between the limit bounds. You must clear or change it. (This step uses a different guess, ⁻6.)

   2 [CLEAR] [(–)] **6**



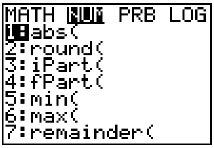
6. Solve for **X**.

   [▼] [▼] [ENTER]



The [MATH] NUM Menu

The [MATH] **NUM** (number) menu includes seven different math functions.

[MATH] [▶]	

1:abs(	Calculates the absolute value of a real number, list, or expression.
2:round(	Rounds a real number, list, or expression.
3:iPart(	Returns only the integer part of a result.
4:fPart(	Returns only the fractional part of a result.
5:min(	Returns the minimum of two real numbers, lists, or expressions.
6:max(	Returns the maximum of two real numbers, lists, or expressions.
7:remainder(	Returns the remainder resulting from the division of two real numbers or lists.

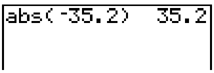
abs([MATH] [▶] 1

Returns the absolute value of a real number, expression, or each element in a list. For an expression, the expression is calculated and the absolute value of that result is returned.

abs(*value***)**

Find the absolute value of ⁻35.2.

[MATH] [▶] **1** [(-)] **3 5 . 2** [)]
[ENTER]



round([MATH] [▸] 2

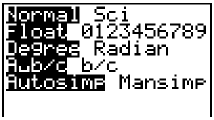
Returns a number, expression, or each element in a list rounded to 10 digits or *#decimal_places* (≤9), if specified. The final result is always displayed according to the Decimal Notation mode ([MODE]) unless *#decimal_places* is specified, which overrides the current setting. Notice that the Decimal Notation mode settings *do* change the display but not the value of the result. Therefore, the entire result is stored in the calculator ready to use for future calculations, as applicable.

round(*value*[,*#decimal_places*]**)**

Round π to different numbers of decimal places using different Decimal Notation mode settings.

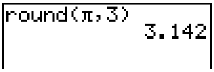
1. Set Decimal Notation mode to **Float**, if necessary.

   [MODE] [▾] [ENTER]
   [2nd] [QUIT] [CLEAR]



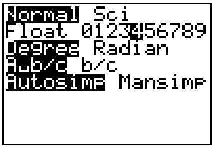
2. Round π to 3 decimal places.

   [MATH] [▸] **2** [2nd] [π]
   [,] **3** [)] [ENTER]



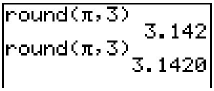
3. Set Decimal Notation mode to **4**.

   [MODE] [▾] [▸] [▸] [▸] [▸] [▸]
   [ENTER] [2nd] [QUIT]



4. Round π to 3 decimal places.

   [2nd] [ENTRY] [ENTER]



5. Leave the Decimal Notation mode at **4** and round π to 5 digits.

   [2nd] [ENTRY] [◄] [◄] **5**
   [ENTER]

```
round(π,3)
           3.142
round(π,3)
          3.1420
round(π,5)
          3.1416
```

iPart(and fPart([MATH] [►] 3 and 4

iPart(returns the integer part of a real number, expression, or each element in a list. For an expression, the expression is calculated and the integer part of the result is displayed.

iPart(*value***)**

fPart(returns the fractional part of a real number, expression, or each element in a list. For an expression, the expression is calculated and the fractional part of the result is displayed.

If *value* is a mixed number, the fractional part is returned and displayed according to the current Simplification mode setting.

fPart(*value***)**

Find the integer and fractional part of 23.45.

1. Set Decimal Notation mode to **Float**.

   [MODE] [▼] [ENTER]
   [2nd] [QUIT]

2. Find the integer part.

   [CLEAR] [MATH] [►] **3**
   2 3 . 4 5 [)] [ENTER]

```
iPart(23.45)   23
```

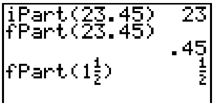
3. Find the fractional part.

   [MATH] [►] **4**
   2 3 . 4 5 [)] [ENTER]

```
iPart(23.45)   23
fPart(23.45)
              .45
```

Find the fractional part of 1 $^1/_2$.

[MATH] [▸] **4** **1** [UNIT]
1 [b/c] **2** [▸] [)] [ENTER]



min(and max([MATH] [▸] 5 and 6

These are identical to the **min(** and **max(** commands found on the [2nd] [STAT] **MATH** menu.

min((minimum) returns the smaller of two *values* or the smallest element in one *list*. *value* can be a real number, expression, or a list.

If both arguments are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is compared with each element of the list, and a list is returned.

min(*valueA*,*valueB***)**
min(*list***)**

max((maximum) functions exactly like **min(**, but it always returns the *larger* of two *values* or the largest element in a list.

max(*valueA*,*valueB***)**
max(*list***)**

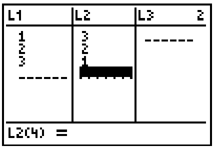
For this example, the Decimal mode settings are set to **Float**.

Compare **L1** and **L2** to find the **min(** and **max(**. **L1={1,2,3}**, and **L2={3,2,1}**.

1. Define **L1** and **L2** in the List editor.

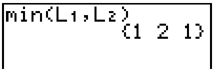
   [LIST]

   For more information on entering lists, see Chapter 5: Lists.



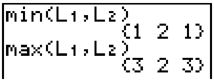
2. Find the list minimums.

   [2nd] [QUIT] [CLEAR]
   [MATH] [▶] **5**
   [2nd] [STAT] **1** [,]
   [2nd] [STAT] **2** [)] [ENTER]



3. Find the list maximums.

   [MATH] [▶] **6**
   [2nd] [STAT] **1** [,]
   [2nd] [STAT] **2** [)] [ENTER]



remainder([MATH] [▶] 7

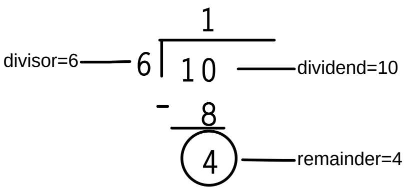
Returns the remainder resulting from the division of two positive whole numbers, *dividend* and *divisor*, each of which can be a list of positive whole numbers. They also are subject to mathematical rules. For example, *divisor*≠0.

remainder(*dividend*,*divisor***)**

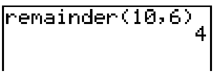
If both arguments are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is paired with each element of the list, and a list of remainders is returned.

remainder(*list*,*divisor***)**
remainder(*dividend*,*list***)**
remainder(*list*,*list***)**

Calculate 10÷6 and return the remainder only.



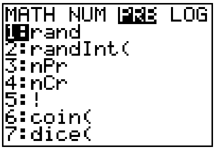
[CLEAR] [MATH] [▶] **7**
1 0 [,] **6** [)] [ENTER]



The [MATH] PRB Menu

The [MATH] **PRB** (probability) menu lets you select functions that are often used to calculate probabilities.

[MATH] [▸] [▸]



1:rand	Generates a random number between 0 and 1.
2:randInt(	Generates a random integer between two values.
3:nPr	Calculates the number of permutations for a group of items.
4:nCr	Calculates the number of combinations for a group of items.
5:!	Calculates the factorial of a positive integer.
6:coin(	Simulates one or more coin tosses.
7:dice(	Simulates one or more dice rolls.

rand [MATH] [▸] [▸] 1

Generates a random real number between 0 and 1 (0<*number*<1). **rand** takes no arguments.

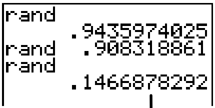
rand

If you want to control a sequence of random numbers, first store an integer "seed value" to **rand**. The calculator generates a specific sequence of random numbers from each seed value. To get a different sequence, use a different seed value. The default seed value is 0.

seed [STO▸] **rand**

Generate a sequence of random numbers using whatever value happens to be the current seed.

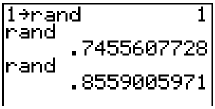
[CLEAR] [MATH] [▶] [▶] **1**
[ENTER] [ENTER] [ENTER]



Your results may vary.

Generate a sequence of random numbers using *seed*=1.

[CLEAR] **1** [STO▶] [MATH] [▶]
[▶] **1** [ENTER] [MATH] [▶] [▶] **1**
[ENTER] [ENTER]



randInt([MATH] [▶] [▶] 2

Generates a random integer between *lower* and *upper* (both integers) boundaries.

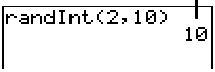
The random integer returned may be one of the boundaries. For example, **randInt(1,5)** may return 1, 2, 3, 4, or 5.

To generate more than one random integer, specify *#ofIntegers*, a positive whole number >0.

randInt(*lower*,*upper*[,*#ofIntegers*]**)**

Find a random integer from 2 through 10.

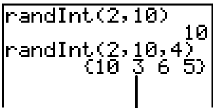
[CLEAR] [MATH] [▶] [▶] **2**
2 [,] **1 0** [)] [ENTER]



Your result may vary.

Find 4 random integers from 2 through 10. (Recall and edit the last entry.)

[2nd] [ENTRY] [◀] [,] **4** [)]
[ENTER]



Your result may vary.

nPr [MATH] [▶] [▶] 3

Returns the number of permutations of n *items* taken r *number* at a time. The order in which you select the items DOES matter. *items* and *number* can be nonnegative integers or lists of nonnegative integers.

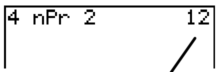
If both arguments are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is paired with each element in the list, and a list of permutations is returned.

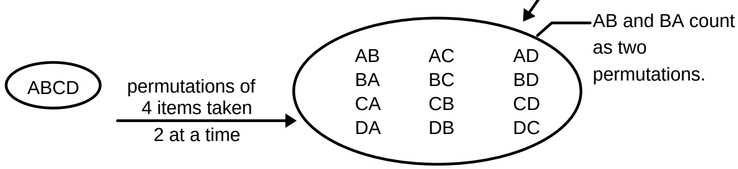
items **nPr** *number*

From a group of 4 items (ABCD), how many ways can you select 2 of the items if the order does matter?

Find 4 **nPr** 2.

[CLEAR] **4** [MATH] [▶] [▶] **3**
2 [ENTER]





nCr [MATH] [▶] [▶] 4

Returns the number of combinations of n *items* taken r *number* at a time. In combinations, the order in which you select the items DOES NOT matter. *items* and *number* can be nonnegative integers or lists of nonnegative integers.

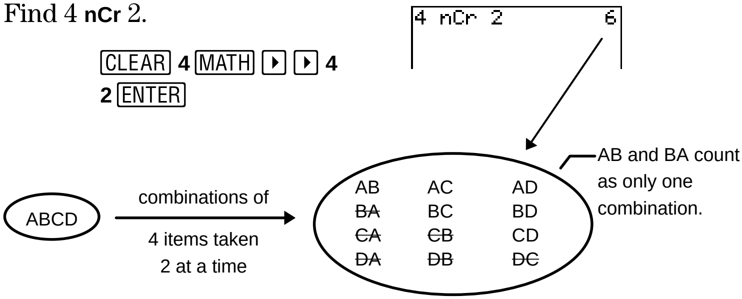
If both arguments are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is paired with each element in the list, and a list of combinations is returned.

items **nCr** *number*

From a group of 4 items (ABCD), how many ways can you select 2 of the items if the order does not matter?

Find 4 **nCr** 2.

[CLEAR] **4** [MATH] [▶] [▶] **4**
2 [ENTER]



! [MATH] [▶] [▶] 5

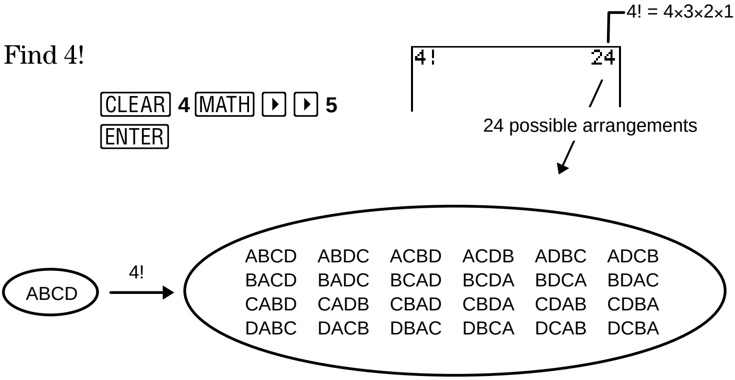
Returns the factorial of *value*. *value* can be an integer or list of integers between 0 and 69. By definition, 0! = 1.

Factorials are similar to permutations because the order DOES matter. You can think of 4! as the total number of ways that 4 items can be arranged.

value **!**

Find 4!

[CLEAR] **4** [MATH] [▶] [▶] **5**
[ENTER]



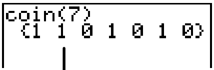
coin([MATH] [▶] [▶] 6

Returns a random list of 0s and 1s that represents heads and tails for one or more coin *tosses*. *tosses* is a positive whole number.

coin(*tosses***)**

Simulate tossing a coin 7 times.

[CLEAR] [MATH] [▶] [▶] **6**
7 [)] [ENTER]



4 heads and 3 tails (or 3 heads and 4 tails). Your result may vary.

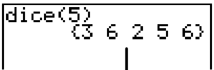
dice([MATH] [▶] [▶] 7

Returns a random list of numbers (between 1 and 6) that represents dice rolls. **dice(** takes one optional argument, *#ofdice*, a positive whole number>1. If *#ofdice* is specified, each list element is the total sum of one roll's results.

dice(*rolls*[,*#ofDice*]**)**

Simulate 5 dice rolls for one die.

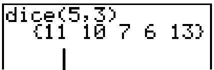
[CLEAR] [MATH] [▶] [▶] **7**
5 [)] [ENTER]



Your result may vary.

Simulate 5 rolls of 3 dice.

[CLEAR] [2nd] [ENTRY]
[◀] [,] **3** [)] [ENTER]

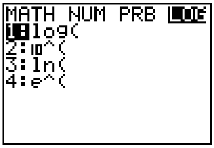


The three dice totaled 11 on the first roll, 10 on the second roll, etc. Your result may vary.

The [MATH] LOG Menu

The [MATH] **LOG** (logarithm) menu lets you select functions that are used to calculate base-10 and base-*e* logarithms and powers.

[MATH] [▶] [▶] [▶]
-or-
[MATH] [◀]



1:log(	Returns the base-10 logarithm of a value.
2:10^(	Raises 10 to a power.
3:ln(	Calculates the natural logarithm of a value.
4:e^(	Raises e to a power (e = 2.71828182846).

log([MATH] [▶] [▶] [▶] 1

The logarithm is the exponent, x, indicating the power which a fixed number (using base 10) must be raised to in order to produce a given number, a.

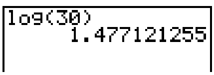
For $10^x = a$, $\log_{10} a = x$

log(returns the logarithm of a positive real number, an expression that results in a positive real number, or a list of positive real numbers.

log(*value***)**
log(*list***)**

Calculate log(30).

[CLEAR] [MATH] [▶] [▶] [▶] **1**
3 0 [)] [ENTER]



10^([MATH] [▶] [▶] [▶] 2

Raises 10 to a power of x, where x is an integer, an expression that results in an integer, or a list of integers. If $x \le 10^{-4}$ or $\ge 10^{10}$, the result is displayed in scientific notation.

10^(*integer***)**
10^(*x***)**

Calculate 10^(6), which is often written as 10^6.

[CLEAR] [MATH] [▶] [▶] [▶] **2**
6 [)] [ENTER]

```
10^(6)        1000000
```

Calculate 10^(⁻4).

[MATH] [▶] [▶] [▶] **2**
[(–)] **4** [)] [ENTER]

```
10^(6)        1000000
10^(-4)          1E-4
```

ln([MATH] [▶] [▶] [▶] 3

The natural logarithm is the exponent, x, indicating the power which the base, e, must be raised to in order to produce a given number, a.

```
For e^x=a, ln(a)=x
```

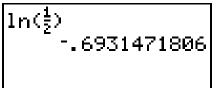
The calculator uses e=2.718281828459, although it only displays 2.718281828 on the screen.

ln(returns the natural logarithm of a positive real number, an expression that results in a positive real number, or a list of positive real numbers.

ln(*value***)**
ln(*list***)**

Calculate ln(1/2).

CLEAR MATH ▶ ▶ ▶ 3
1 b/c 2 ▶) ENTER



e^(MATH ▶ ▶ ▶ 4

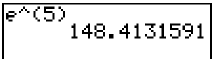
Raises e to a power of x, where x is a real number, an expression that results in an real number, or a list of real numbers.

The calculator uses e=2.718281828459, although it only displays 2.718281828 on the screen.

e^(x**)**
e^(*list***)**

Calculate e^5, which is often written as e^5.

CLEAR MATH ▶ ▶ ▶ 4
5) ENTER



3 Fractions

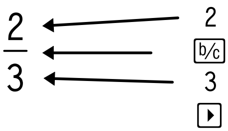
Entering Fractions

Simple fractions consist of a numerator and denominator. Mixed numbers combine a whole number with a fraction. **Note:** The numerator and denominator *cannot* be a fraction.

Simple Fractions

numerator [b/c] *denominator* [▶]

Enter 2/3.



1. Enter the numerator, 2.

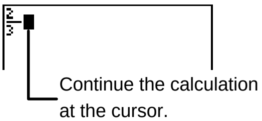
   2 [b/c]

2. Enter the denominator, 3.

   3

3. End the fraction.

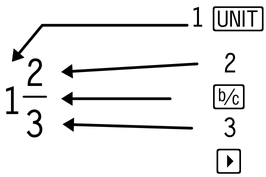
   [▶]



Mixed Numbers

whole_number [UNIT] *numerator* [b/c] *denominator* [▶]

Enter 1 2/3.



1. Enter the whole number, 1.

   1 [UNIT]

2. Enter the numerator, 2.

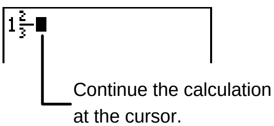
   2 [b/c]

3. Enter the denominator, 3.

   3

4. End the mixed number.

   [▶]



Using Fractions in Calculations

The type of calculation and the input values determine whether the results of a calculation are shown as a fraction or a decimal. You can enter fractions with all operation keys ([+], [×], [÷], etc.), most function keys ([x^2], [%], [x^{-1}], etc.), and many menu items (**abs(**, **fPart(**, **sin(**, etc.).

Fractional calculations return fractional results, if possible, except for those that:

- Use [2nd] [π], [%], **log(**, **ln(**, **e^(**

  – or –

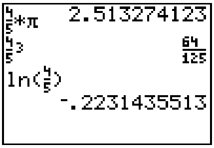
- Calculate to a result $\geq \frac{1000}{1}$ or $< \frac{1}{1000}$

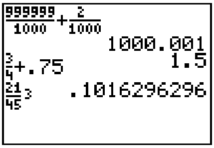
  – or –

- Include both a fraction and a decimal

  – or –

- Use items from the following menus: [2nd] [CONVERT]; [2nd] [STAT] **MATH** and **CALC**; [2nd] [TRIG] **TRIG** and **ANGLE**





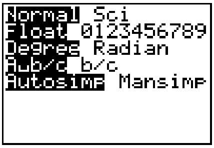
Fraction Modes

Two fraction modes exist on the calculator: Display Format mode and Simplification mode.

Display Format Mode Settings

The Display Format mode settings, **A⌴b/c** and **b/c**, determine whether or not a fractional result is displayed as a mixed number or a simple fraction. To select a mode setting, press [MODE], highlight the setting with the cursor keys, and then press [ENTER].

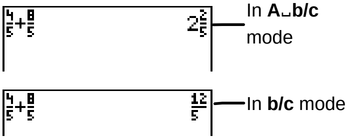
[MODE]



A⌴b/c	Displays result as a mixed number, if applicable.
b/c	Displays result as a simple fraction.

Add $4/5 + 8/5$.

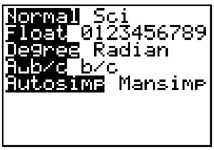
4 [b/c] **5** [▶] [+] **8** [b/c] **5** [ENTER]



Simplification Mode Settings

The Simplification mode settings, **Autosimp** and **Mansimp**, determine whether or not a fractional result is simplified automatically.

[MODE]



Autosimp	The calculator automatically simplifies fractional results.
Mansimp	The user simplifies fractions manually step-by-step. ↓ next to the result signifies that it can be simplified at least one more time.

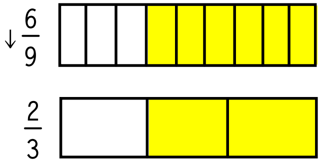
Add $^1/_9 + {}^5/_9$.

1 [b/c] **9** [▶] [+] **5** [b/c] **9** [ENTER]

In **Autosimp** mode

In **Mansimp** mode

The total shaded area in the first diagram is equal to the total shaded area in the second one.



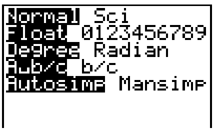
Autosimp Setting

In this example, Display Format mode settings do not affect the display of the result because the result is a simple fraction.

Add $^1/_4 + {}^1/_4$.

1. Select **Autosimp** mode, if necessary, and return to the Home screen.

   [MODE] [▼] [▼] [▼] [▼] [ENTER] [2nd] [QUIT]



2. Add $^1/_4 + {}^1/_4$.

   1 [b/c] **4** [▶] [+] **1** [b/c] **4** [ENTER]

Mansimp Setting

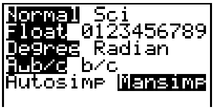
When the **Mansimp** setting is selected, the result of a calculation is not simplified automatically. ↓ next to a result means that it is unsimplified and can be simplified at least one more time. You then can decide if you want the calculator to simplify the result step-by-step using simplification factors it chooses or if you want the calculator to simplify the result using the simplification factors that you choose.

Letting the Calculator Choose the Simplification Factor

After getting an unsimplified result (one with ↓ next to it) from any fractional calculation, press [SIMP] [ENTER]. The simplified result and simplification factor which the calculator chose are displayed. For example, **Fac=3** means simplification factor=3. The Display Format mode settings affect whether a result is displayed as a mixed number or a simple fraction.

1. Select **Mansimp** mode setting, if necessary, and return to the Home screen.

   [MODE] [▼] [▼] [▼] [▼] [▶]
   [ENTER] [2nd] [QUIT]



2. Add $1\ ^3/_4 + 2\ ^4/_8$.

   [CLEAR] **1** [UNIT] **3** [b/c] **4** [▶] [+]
   2 [UNIT] **4** [b/c] **8** [ENTER]

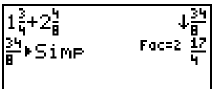
In **A⌴b/c** mode

In **b/c** mode

3. Let the calculator simplify the result.

   [SIMP] [ENTER]

In **A⌴b/c** mode



In **b/c** mode

Choosing the Simplification Factor

After getting an unsimplified result from any fractional calculation, press [SIMP] *simplification_factor* [ENTER], where *simplification_factor* is a positive integer that you choose. The Display Format mode settings affect whether a result is displayed as a mixed number or as a simple fraction.

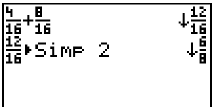
Add $4/16 + 8/16$ and choose the simplification factor to reduce the sum to lowest terms.

1. Enter $4/16 + 8/16$.

   [2nd] [QUIT] [CLEAR]
   4 [b/c] **1 6** [▸] [+] **8** [b/c]
   1 6 [ENTER]

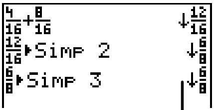
2. Simplify by 2.

   [SIMP] **2** [ENTER]



3. Simplify by 3.

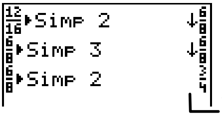
   [SIMP] **3** [ENTER]



The result did not change. Try again with another factor.

4. Simplify by 2.

   [SIMP] **2** [ENTER]



Simplification is complete.

Recalling the Factor [2nd] [VARS] 6:Factor

If you execute a fractional calculation in **Mansimp** mode and then the user or the calculator simplifies the result, you can recall the simplification factor at a later time by selecting [2nd] [VARS] **6:Factor.**

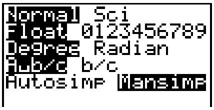
Since **Factor** is a variable, you can use **Factor** in expressions or on any screen that accepts whole numbers (Y= editor, List editor, Home screen, etc.).

Only one simplification factor (the last one calculated) is stored in memory. Also, you can store a positive whole number to **Factor** using the [STO▸] key, just as you would store a number to any variable. For more information about storing values to variables, see Chapter 1: Operating the TI-73.

From the Home screen, simplify $6/8$ by a factor of 2, and then recall the factor.

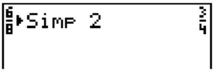
1. Select **Mansimp** mode, if necessary.

   [MODE] [▼] [▼] [▼] [▼] [▶]
   [ENTER] [2nd] [QUIT]



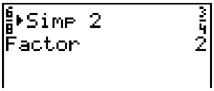
2. Enter the fraction and simplify.

   [CLEAR] **6** [b/c] **8** [SIMP] **2**
   [ENTER]



3. Recall the simplification factor, **2**.

   [2nd] [VARS] **6** [ENTER]



Converting Between Fractions and Decimals

To convert a fraction to a decimal or a decimal to a fraction, use [F◂▸D]. If a fractional equivalent of a decimal does not exist, the calculator returns the same decimal number. Also, the calculator only recognizes and converts (if possible) the first ten digits of any decimal number.

You must follow [F◂▸D] with [ENTER]; otherwise, you get an error.

The current Decimal Notation mode determines the display of the result. In the following example, the calculator is set to **Float** Decimal Notation mode.

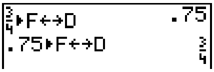
Convert 3/4 to a decimal and back to a fraction.

1. Convert 3/4 to a decimal.

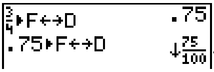
   [2nd] [QUIT] [CLEAR]
   3 [b/c] **4** [F◂▸D] [ENTER]

2. Convert .75 back to a fraction.

   [F◂▸D] [ENTER]



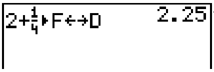
In **Autosimp** mode



In **Mansimp** mode

Add 2 plus the decimal equivalent of 1/4.

2 [+] **1** [b/c] **4** [F◂▸D] [ENTER]



1/4=.25

Converting Between Mixed Numbers and Simple Fractions

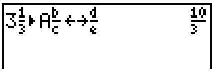
To convert a mixed number to a simple fraction or a simple fraction to a mixed number use [A b/c◂▸d/e]. The Display Format mode settings do not affect the results when using [A b/c◂▸d/e].

You must follow [A b/c◂▸d/e] with [ENTER]; otherwise, you get an error.

Convert 3 1/3 to a simple fraction and back to a mixed number.

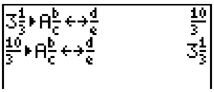
1. Convert 3 1/3 to a simple fraction.

   [2nd] [QUIT] [CLEAR]
   3 [UNIT] **1** [b/c] **3** [A b/c◂▸d/e]
   [ENTER]



2. Convert 10/3 back to a mixed number.

   [A b/c◂▸d/e] [ENTER]

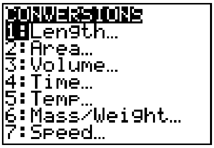


4 Measurement Conversions and Constant Calculations

The [2nd] [CONVERT] CONVERSIONS Menu

Use this menu to access all conversion categories.

[2nd] [CONVERT]



1:Length	Displays the **LENGTH** menu.
2:Area	Displays the **AREA** menu.
3:Volume	Displays the **VOLUME** menu.
4:Time	Displays the **TIME** menu.
5:Temp	Displays the **TEMPERATURE** menu.
6:Mass/Weight	Displays the **MASS/WT.** menu.
7:Speed	Displays the **SPEED** menu.

Length [2nd] [CONVERT] 1

mm .... millimeters
cm .... centimeters
m .... meters
inch .... inches
ft .... feet
yard .... yards
km .... kilometers
mile .... miles

Area [2nd] [CONVERT] 2

ft² .... square feet
m² .... square meters
mi² .... square miles
km² .... square kilometers
acre .... acres
in² .... square inches
cm² .... square centimeters
yd² .... square yards
ha .... hectares

Volume [2nd] [CONVERT] 3

liter .... liters
gal .... gallons
qt .... quarts
pt .... pints
oz .... ounces
cm³ .... cubic centimeters
in³ .... cubic inches
ft³ .... cubic feet
m³ .... cubic meters
galUK .... UK gallons
ozUK .... UK ounces

Time [2nd] [CONVERT] 4

sec .... seconds
min .... minutes
hr .... hours
day .... days
week .... weeks
year .... years

Temp (Temperature) [2nd] [CONVERT] 5

degC .... degrees Celsius
degF .... degrees Fahrenheit
degK .... degrees Kelvin

Mass/Weight [2nd] [CONVERT] 6

g .... grams
kg .... kilograms
lb .... pounds
ton (US) .... tons
mton (US) .... metric tons

Speed [2nd] [CONVERT] 7

ft/s .... feet per second
m/s .... meters per second
mi/hr .... miles per hour
km/hr .... kilometers per hour
knot .... knots

Converting a Unit of Measure

To convert a measurement value, enter the measurement value, select the category from the **CONVERSIONS** menu, select the unit you are converting *from*, and then the unit you are converting *to*. To know which category to select, look at the units of the *original* value. You can only convert within one category.

measurement_value current_unit▶*new_unit*

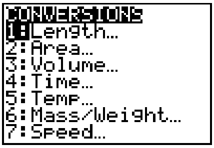
Convert 50 meters to inches.

1. Clear the Home screen, if desired. Enter the value, 50.

   [2nd] [QUIT] [CLEAR] **5 0**

2. Display the **CONVERSIONS** menu.

   [2nd] [CONVERT]

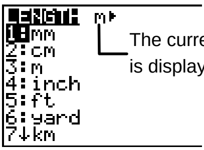


3. Select the applicable category, **1:LENGTH**.

   1

4. Select the current unit, meters.

   3



The current unit is displayed.

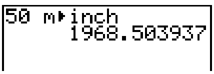
5. Select the unit which you want to convert *to*, inches.

   4

m▶ inch is pasted to the Home screen.

6. Calculate the result.

   [ENTER]



EOS operating rules (Appendix B: Reference Information) apply when converting negative measurements as shown in the next example.

Compare the results of -5°F▸°C and (-5)°F▸°C.

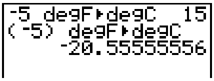
1. From the Home screen, calculate -5°F▸°C.

   [2nd] [QUIT] [CLEAR]
   [(-)] **5** [2nd] [CONVERT] **5**
   2 1 [ENTER]

The calculator converts 5°F to °C and then returns the negative of the result.

2. Calculate (-5)°F▸°C.

   [2nd] [ENTRY] [2nd] [◄]
   [2nd] [INS] [(] [►] [►]
   [2nd] [INS] [)] [ENTER]



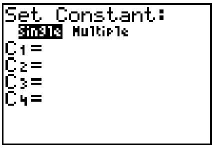
The calculator converts (-5)°F to °C.

Constants

To save time re-entering long or complicated expressions and to help prevent entry errors, you can enter numbers, expressions, lists, commands, or functions into the calculator's memory by defining them as constants in the Set Constant editor. As constants, they then can be recalled at any time.

You can define up to four constants in the Set Constant editor and choose from one of two different modes: **Single** or **Multiple**. The mode you select determines how many of the constants you can recall at a time. To enter a constant in the editor, select the mode from the Set Constant editor ([2nd] [SET]), move the cursor to one of the four constants, and define it.

[2nd] [SET]



To use a constant:

1. Define the constant in the Set Constant editor ([2nd] [SET]).
2. Recall the constant with the [CONST] key.

Single Mode

By selecting **Single** mode, you tell the calculator that you only want to access one constant from the list, even if more than one is defined.

To select the one constant (C_1, C_2, C_3, or C_4) you want to use, highlight the **=** next to it, and then press [ENTER]. This automatically deselects any other defined constants.

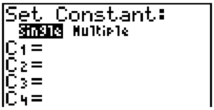
Defining Constants in Single Mode

Enter the constants in the Set Constant editor as shown in the following example. At any time you can enter this editor and edit, delete, or add constants.

Define $C_1 = +1/2$ and $C_3 = *1/2$.

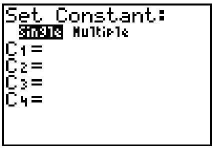
1. Enter the Set Constant editor.

   [2nd] [SET]



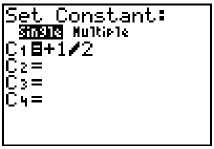
2. Highlight **Single** with the cursor, if necessary.

   [▲] [ENTER]



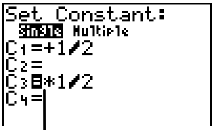
3. Define C_1 as $+1/2$.

   [▼] [+] **1** [b/c] **2**



4. Define C_3 as $*{}^1/_2$.

   [▼] [▼]
   [×] 1 [b/c] 2



The most recently entered constant (C_3) remains selected.

5. Exit the Set Constant editor.

   [2nd] [QUIT]

Single Mode Constant Calculations

After a constant is defined and selected, return to the screen where you want to use it in a calculation. Pressing [CONST] pastes it to the cursor location. In **Single** mode, only one defined constant is available for use in calculations, and an expression using a constant is automatically solved after pressing [CONST] (without pressing [ENTER]).

You recognize that C_1=+ $^1/_2$.

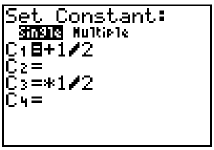
Calculate $40+{}^1/_2$.

1. Select **Single** mode, if necessary.

   [2nd] [SET] [▲] [ENTER]

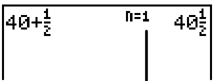
2. Select C_1 (which deselects C_3), and exit the Set Constant editor.

   [▼] [◄] [ENTER] [2nd] [QUIT]



3. Clear the Home screen, if desired. Solve the problem using C_1.

   [CLEAR] 4 0 [CONST]



Count=1.

Recalling a Constant in a Series of Calculations

When pressing [CONST] more than once in a series of calculations, the calculator automatically keeps count for you (shown in the following example) unless the defined constant includes a list. The counter starts over any time a new entry precedes [CONST], including **Ans**.

Find the multiple of 2 so that $5 * 2^n = 40$.

The calculator's constant counter automatically computes **n**.

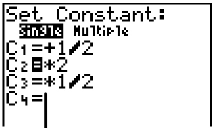
5∗2ⁿ=40	Constant mode = Single Set C_n = ∗ 2

1. Select **Single** mode, if necessary.

   [2nd] [SET] [▲] [ENTER]

2. Enter C_2 = ∗ 2.

   [▼] [▼] [×] **2**



Other defined constants are deselected automatically.

3. Return to the Home screen and clear, if desired.

   [2nd] [QUIT] [CLEAR]

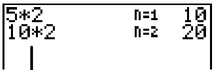
4. Count the number of times you have to multiply 5 by 2 to get 40 (so that $5 \times 2^n = 40$).

   5 [CONST]

5 × 2 is calculated automatically.

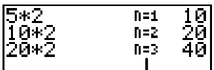
Count=1

   [CONST]



Pressing [CONST] a second time acts like [2nd] [ANS] [CONST].

   [CONST]



n=3 because you multiplied by 2 three times in a row.

Solution	5∗2∗2∗2=40, or $5*2^3$ =40

n=3

Multiple Mode

In **Multiple** mode, all defined constants are available to use at any time. To define **Multiple** mode, highlight **Multiple** using the cursor keys, and then press [ENTER].

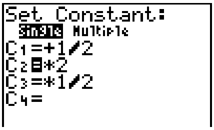
Defining Constants in Multiple Mode

You define constants in **Multiple** mode exactly the same way you define them in **Single** mode. All constants are always selected, even if they are not defined.

Select **Multiple** mode and use the constants defined in the previous examples.

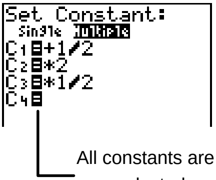
1. Enter the Set Constant editor.

   [2nd] [SET]



2. Select **Multiple** mode.

   [▲] [▶] [ENTER]

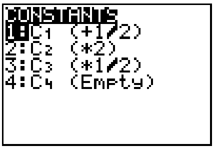


All constants are now selected.

Recalling Constants in Multiple Mode

When you press [CONST] from the Home screen and the Set Constant editor is in **Multiple** mode, the first six characters of every defined constant is displayed. Undefined constants are marked as **Empty**.

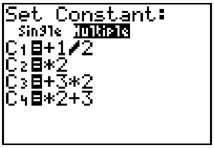
[CONST]



To select a constant, press the number associated with the constant (**1**, **2**, **3**, or **4**). You may choose another constant (or the same one) by pressing [CONST] again. In **Multiple** mode (unlike in **Single** mode), your constant expression is not evaluated until you press [ENTER].

Define C_3**=+3∗2** and C_4**=∗2+3** in **Multiple** mode.

[2nd] [SET] [▲] [▶] [ENTER]
[▼] [▼] [▼] [CLEAR]
[+] **3** [×] **2** [▼] [CLEAR]
[×] **2** [+] **3**



You recognize that C_3**=+3∗2**.

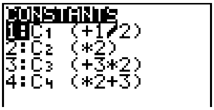
Calculate 4+3∗2.

1. Go to the Home screen and clear it, if desired.

   [2nd] [QUIT] [CLEAR]

2. Find the result.

   4 [CONST] **3** [ENTER]

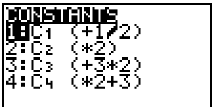


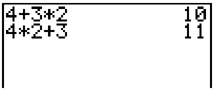
Press [ENTER] to evaluate the expression.

You recognize that C_4**=∗2+3**.

Calculate 4∗2+3.

4 [CONST] **4** [ENTER]





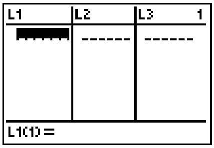
5 Lists

Steps for Creating a List

On the TI-73, a set of numerical or text information is called a list. Follow these basic steps when defining a list.

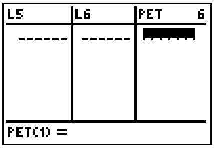
Display the List editor.

[LIST]



Name the list you want to define, or use a prenamed list (**L1**-**L6**).

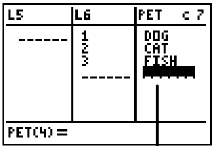
[2nd] [TEXT] (to name a list)



Prenamed lists, **L5** and **L6**, and one user-named list, **LPET**.

Enter list elements (numerical or text).

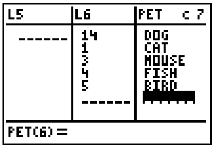
number keys or [2nd] [TEXT]



Note: Surround text elements with quotation marks.

Edit list as necessary.

[2nd] [INS], [CLEAR], or [DEL]

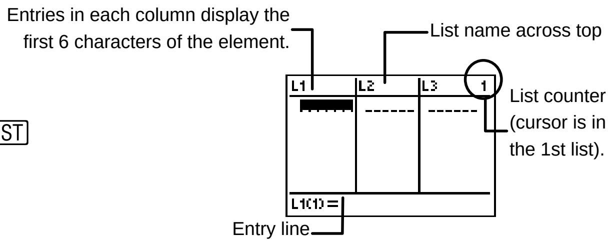


The List Editor LIST

You can enter up to 20 lists in the List editor. Each list can have up to 999 elements. You can only display three lists at the same time; use ▶ or ◀ to scroll to see all other defined lists.

List notation looks like this: **L5={1,2,3,4,5,6}.** Read it as "elements 1, 2, 3, 4, 5, and 6 are stored in the list named **L5**."

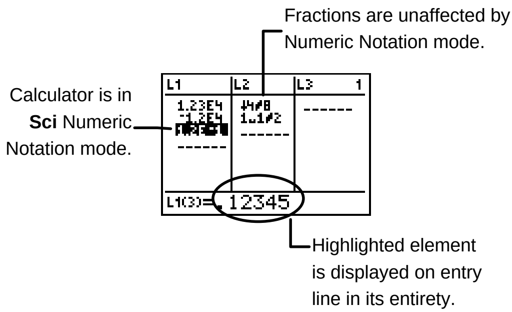
LIST



L1, **L2**, **L3**, **L4**, **L5**, **L6**, and one empty, unnamed list initially are included in the List editor.

Numeric Notation, Decimal Notation, and Angle modes affect the display of an element (except fractional elements).

LIST



Naming a List

When you are ready to define your list, you can move to one of the columns labeled **L1- L6** and begin entering your list elements.

If you do not want to use **L1-L6** (you cannot rename them), you can create a new list and name it anything you want. A list name can be one to five characters long. The first character must be a letter from A to Z. The second through fifth characters can be any combination of letters and numbers. Access letters from the Text editor ([2nd] [TEXT]). A list accepts elements only after it is named.

Note: You cannot rename a user-named list, but you can copy its elements to a list with a different name. See the section entitled, "Copying One List to Another" on page 103.

In this guidebook, when a list name is referred to, its name is always preceded by the **ʟ** symbol; however, you don't type the **ʟ** when naming a list in the List editor.

If a defined list name is highlighted, the list elements or the attached formula are displayed on the entry line.

Create a list named **NUM**.

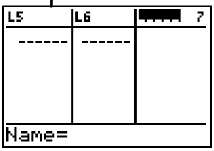
1. Display the List editor.

   [LIST]

Or, insert a blank list. Highlight a list name and press [2nd] [INS].

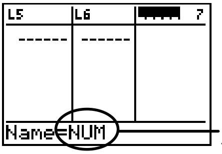
2. Scroll to the blank, unnamed list to the far right of the List editor.

   [▶] or [◀]



3. Use the Text editor to name the list **NUM.**

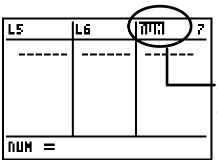
   [2nd] [TEXT]
   N [ENTER] **U** [ENTER]
   M [ENTER] **Done** [ENTER]



NUM appears on the entry line.

4. Move "**NUM**" from the entry line to the list name line.

   [ENTER]



ʟNUM now accepts elements.

Entering List Elements

A named list accepts two types of elements: *numerical* and *text*.

- Lists that contain numerical elements not enclosed in quotation marks are called *numerical* lists.
- Lists that contain text elements or numerical elements whose numerical values are ignored (because they are enclosed in quotation marks) are called *categorical* lists.

To enter an element, highlight the space in the column under the list name where you want the element to be entered (you can't skip any spaces) and type in the element (it is displayed on the entry line). Press [ENTER] or [▼] to move the element into the list. Pressing [▼] or [ENTER] also moves the cursor to the next element space.

Access the quotation marks (for categorical lists) from the Text editor ([2nd] [TEXT]).

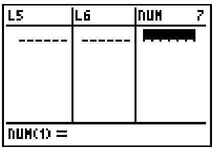
Numerical Lists

Numerical lists contain real numbers, fractions, or expressions that evaluate to real numbers or fractions. If you enter an expression like **sin(30)**, the calculator displays the decimal equivalent in the list element space. The Numeric Notation, Decimal Notation, and Angle modes determine how the calculator displays all elements, except fractions.

Define **ʟNUM={18,25,45}**.

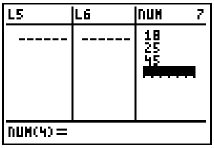
1. Go to the first element space of the numerical list **ʟNUM**.

   [LIST]
   [▶] (as necessary)
   [▼] (if necessary)



2. Enter the list elements.

   18 [▼] **25** [▼] **45** [▼]

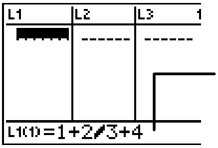


Entering Fractional Elements

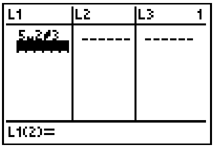
When entering fractions from the Home screen, parentheses are optional around the numerator and denominator.

When entering fractions in the List editor (and any other editor), parentheses are mandatory around the numerator and denominator ONLY when operators are included:

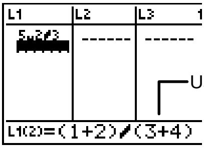
[LIST] **1** [+] **2** [b/c] **3** [+] **4** [ENTER]



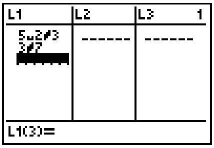
Interpreted as 1+(2/3)+4



[(] **1** [+] **2** [)] [b/c] [(] **3** [+] **4** [)] [ENTER]



Using parentheses



Dependent Numerical Lists

The numerical list described in the previous section (**ʟNUM**) is an *independent* list. You also can create *dependent* lists, which are dependent (or based) upon the contents of another defined numerical list.

You create a dependent list by attaching a *formula* to it. For example, **"2 + L1,"** where **L1** is already defined, is a formula. The formula always contains at least one other list. In addition, for a formula like **L3="2+L1+ L2,"** **L1** and **L2** must have the same number of elements. Then, each element in **L3** is the result of the attached formula.

When a formula is attached to a list, a small signifier (◆) appears next to the list name. You cannot edit a dependent list by simply typing over an element as with independent lists. You must highlight the element you want to change, press [ENTER], and then edit it. However, this changes the entire list back to an independent list, and the formula and the formula signifier disappear.

Also, it is possible to have multiple dependent lists all based on the same list (for example, **L2="2+ L1,"** **L3="3+L1,"** and **L4="4+L1"**).

An attached formula can be enclosed in quotation marks (located in the Text editor). A list whose formula:

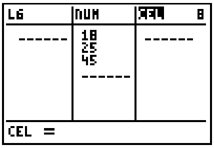
- Is *not* enclosed in quotation marks is *not* automatically updated if the independent list changes.
- Is enclosed in quotation marks is automatically updated if the independent list changes.

Convert the following six Celsius temperatures {⁻40,⁻15,⁻5,30,58,140} to Fahrenheit and display both lists in the List editor.

Independent List	ʟCEL={⁻40,⁻15,⁻5,30,58,140}
Dependent List	ʟFRHT="ʟCEL degC▶degF"

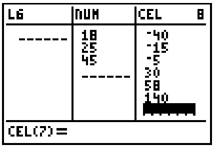
1. Create the independent list, **ʟCEL**.

   [LIST]
   [▶] or [◀] (as necessary to move to the blank list)
   [2nd] [TEXT] **C** [ENTER] **E** [ENTER] **L** [ENTER] **Done** [ENTER] [ENTER]



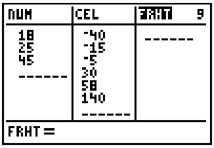
2. Enter the elements.

   [▼] [(-)] **4 0** [▼] [(-)] **1 5** [▼] [(-)] **5** [▼] **3 0** [▼] **5 8** [▼] **1 4 0** [▼]



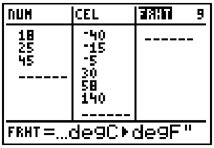
3. Create the dependent list, **ʟFRHT**.

   [▶] [2nd] [TEXT]
   F [ENTER] **R** [ENTER] **H** [ENTER] **T** [ENTER] **Done** [ENTER] [ENTER]



4. Attach the formula **"ʟCEL degC▶degF"** to **ʟFRHT**.

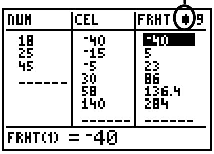
   [ENTER] [2nd] [TEXT]
   " [ENTER] **Done** [ENTER]
   [2nd] [STAT] **CEL** [ENTER]
   [2nd] [CONVERT] **5 1 2**
   [2nd] [TEXT] **"** [ENTER]
   Done [ENTER]



A small formula signifier appears.

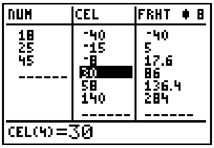
5. Display the elements of **ʟFRHT**.

   [ENTER]



6. Change **-5** in **ʟCEL** to **-8**.

   [◄] [▼] [▼] [ENTER] [(-)] **8** [ENTER]



Note: Since the formula is enclosed in quotation marks, element 3 in **ʟFRHT** is automatically updated.

Categorical Lists

Categorical lists usually contain words or letters (text elements). If they contain numerical elements, the numerical values of those elements are ignored. Categorical lists are usually used in statistical plotting, but they can allow you to label elements as explained in the following example. See Chapter 6: Statistical Plots for details about using categorical lists in stat plots.

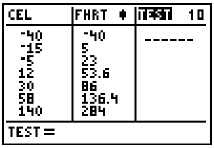
To define a categorical list, enclose the first element in quotation marks (found in the Text editor). Quotation marks are optional on the remaining text elements. A categorical list signifier, **c**, appears next to the list name.

A math class has 4 test scores: 2 tests, 1 midterm test, and 1 final exam. Ivan earned test scores of 85, 80, 74, and 82. Karen earned test scores of 90, 85, 92, and 79. Reflect this information in the List editor.

1 Categorical	ʟTEST={TEST1,TEST2,MDTRM,FINAL}
2 Numerical	ʟIVAN={85,80,74,82}
	ʟKAREN={90,85,92,79}

1. Display the List editor and create a list named **TEST**.

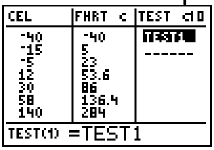
    [LIST] [▶] or [◀] (as necessary to move to the blank list)
    [2nd] [TEXT]
    T [ENTER] **E** [ENTER]
    S [ENTER] **T** [ENTER]
    Done [ENTER] [ENTER]



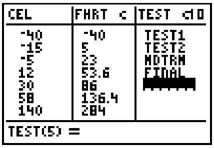
2. Enter the element **TEST1.**

    [▼] [2nd] [TEXT] **"** [ENTER]
    T [ENTER] **E** [ENTER]
    S [ENTER] **T** [ENTER] **1**
    " [ENTER] **Done**
    [ENTER] [ENTER]

The categorical list signifier, **c**.

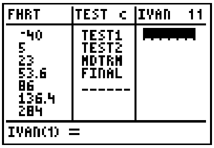


3. Repeat for the elements **TEST2**, **MDTRM**, and **FINAL** (quotation marks are optional after the first element).



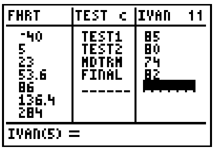
4. Create a list named **IVAN.**

    [▶] [2nd] [TEXT]
    I [ENTER] **V** [ENTER]
    A [ENTER] **N** [ENTER]
    Done [ENTER] [ENTER]



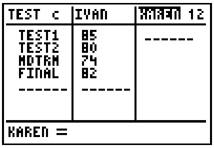
5. Enter **85**, **80**, **74**, and **82**.

    [▼] **8 5** [▼] **8 0** [▼] **7 4** [▼]
    8 2 [▼]



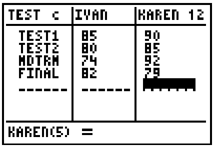
6. Create a list named **KAREN.**

   ▶ [2nd] [TEXT]
   K [ENTER] **A** [ENTER]
   R [ENTER] **E** [ENTER]
   N [ENTER] **Done** [ENTER]
   [ENTER]



7. Enter **90**, **85**, **92**, and **79.**

   ▼ **9 0** ▼ **8 5** ▼ **9 2** ▼
   7 9 ▼



Once you have these lists entered, you can display this data in various ways using related features on the calculator. For example, Chapter 6: Statistical Plots explains how you could easily convert this data into a bar chart. Chapter 7: Statistical Analyses explains ways to find each student's averages as well as doing other statistical analyses of their test scores.

Editing Lists in the List Editor

From the List editor, you can display, edit, insert, temporarily delete (not from memory), and move from view all lists stored in the calculator. You also can edit, insert, move, or delete list elements and attached formulas.

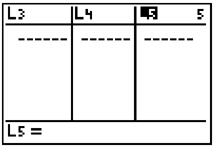
To see all list names that are stored into the calculator's memory (but not necessarily the List editor), display the [2nd] [STAT] **Ls** menu and use ▲ and ▼ to scroll the menu.

Inserting or Deleting a New List

Inserting a list into the List editor saves it in the calculator's memory. However, deleting a list from the List editor does not delete it from the calculator's memory. A deleted list's name still appears in the [2nd] [STAT] **Ls** menu.

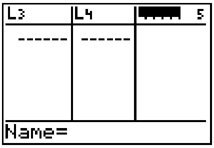
Therefore, if you would like to insert the deleted list back into the List editor, go to a blank list, select the list name from [2nd] [STAT] **Ls** menu and press [ENTER] [ENTER].

Insert **L1** between **L4** and **L5**.



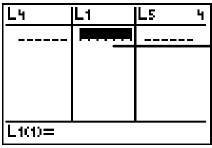
1. Move the cursor so that it highlights **L5**.

   [LIST] [▲]
   [▶] or [◀] (as necessary)



2. Insert a blank list.

   [2nd] [INS]



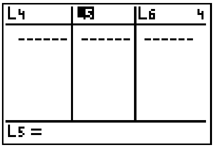
If elements exist in **L1**, then those appear also.

3. Identify it as **L1**.

   [2nd] [STAT] **1** [ENTER]

Delete **L1**.

[DEL]

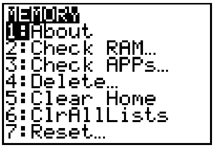


Deleting Lists from Calculator Memory

To delete a list from the calculator's memory, use the [2nd] [MEM] **4:Delete** menu. If you delete **L1**–**L6** from the calculator's memory, the names still appear in the [2nd] [STAT] **Ls** menu. If you delete a user-named list, its name is deleted from this menu.

Delete **L2** from the calculator's memory.

2nd [MEM] **4** **3** [▼] ENTER
2nd [QUIT] (to return to the Home screen)



Inserting or Deleting One Element in a List

To insert one element in a list:

1. Use the cursor keys as necessary to highlight the element space where you want to insert the element.
2. Press 2nd [INS] to insert the element space. All following elements move down one space.
3. Type the element, and press ENTER.

To delete one element from a list:

1. Use the cursor keys as necessary to highlight the element that you want to delete.
2. Press DEL to delete the element. All following elements move up one space.

Editing an Existing Element

You can edit any particular element in a list without having to reenter the entire list.

1. Use the cursor keys as necessary to highlight the element that you want to edit.
2. Press ENTER to move the element to the entry line.
3. Edit the element with 2nd [INS], CLEAR, or DEL, as necessary.
4. Press ENTER to replace the existing element with the edited element.

Clearing All Elements in a List

To clear all of the elements in a list when the List editor is displayed:

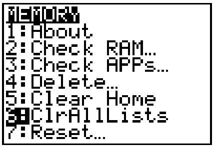
1. Use the cursor keys as necessary to highlight the list name. The list elements (or formula) are displayed on the entry line.
2. Press [CLEAR] [ENTER] to clear the list elements.

You also can clear elements from the Home screen using the [2nd] [STAT] **OPS 3:ClrList** menu item.

Clearing All Elements in All Lists

You clear all elements in all lists using the [2nd] [MEM] **6:ClrAllLists** instruction from the Home screen. When you press [ENTER], all elements in all lists are cleared from the calculator's memory, even for those lists not displayed in the List editor.

[2nd] [QUIT]
[2nd] [MEM] **6** [ENTER]



Editing a List Formula

To edit an attached formula:

1. Use the cursor keys as necessary to highlight the name of the list name that you want to edit.
2. Press [ENTER] to move the formula to the entry line.
3. Edit the formula with [2nd] [INS], [CLEAR], or [DEL], as necessary.
4. Press [ENTER] to replace the existing formula with the edited formula. The list elements are updated automatically according to the new formula.

Deleting a List Formula

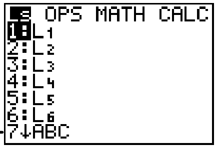
You can delete an attached formula in one of two ways. You can:

- Follow the preceding directions for editing a formula, but press [CLEAR] [ENTER] in place of step 3.
- Edit one of the elements in the dependent list as directed in the steps for editing an element. When you are finished, the formula signifier disappears, and the list becomes independent.

The [2nd] [STAT] Ls Menu

Use the [2nd] [STAT] **Ls** (lists) menu to access all list names stored in the calculator's memory. **L1-L6** are listed first followed by all user-named lists in alphabetical order. In this menu, the user-named lists appear as they do in the List editor (the List signifier, ʟ, does not precede the name). However, if you select a list to display it anywhere else on the calculator, such as the on Home screen, the ʟ automatically appears before the name.

[2nd] [STAT]



User-named lists follow **L1-L6** in alphabetical order.

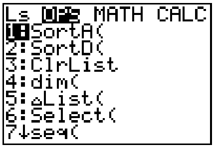
From the Home screen, you can type in a new list name directly using the Text editor (except for **L1- L6**); *however*, you *must* precede the list name with the list signifier, ʟ. Notice that the list signifier, ʟ, is smaller than the **L** in the Text editor. You can access ʟ by itself from [2nd] [CATALOG] or under the [2nd] [STAT] **OPS** menu.

If you try to use the **L** from the Text editor, the calculator reads that **L** plus any following characters as variables (representing numerical values), not as a list.

The [2nd] [STAT] OPS Menu

Use the [2nd] [STAT] **OPS** (options) menu to change defined lists from the Home screen.

[2nd] [STAT] [▸]



1:SortA((Ascending)	Sorts list elements from lowest to highest in numerical order or in alphabetical order.
2:SortD((Descending)	Sorts list elements from highest to lowest in numerical order or in reverse alphabetical order.
3:ClrList	Clears all elements in specified list(s).
4:dim(	Recalls, sets, or changes the dimension (number of elements) in a list.
5:ΔList(	Returns the differences between consecutive elements in a list.
6:Select(	Selects one or more specific data points from a Scatter or xyLine stat plot, and then updates the list(s) in memory. (Requires you to set up a statistical plot. See Chapter 6: Statistical Plots for more information.)
7:seq(	Returns a list that fulfills the requirements of 5 arguments (*expression*, *variable*, *begin*, *end*, and *increment*) which you specify.
8:augment(	Combines two lists to make a new list.
9:L	List signifier; all text characters or numbers following it are interpreted as a list name.

SortA(and SortD([2nd] [STAT] [►] 1 and 2

SortA((sort ascending) sorts numerical list elements from lowest to highest value and categorical list elements alphabetically. **SortD(** (sort descending) sorts the list elements from highest to lowest value or in reverse alphabetical order.

Enter the **SortA(** or **SortD(** instruction on the Home screen; and then enter all list names that you want to sort (separated by a comma), and press [ENTER].

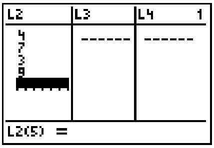
Sorting One List

SortA(*list***)**
SortD(*list***)**

Define **L2={4,7,3,9}** in the List editor, and sort in ascending order.

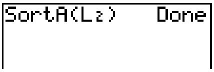
1. Define **L2** in the List editor.

   [LIST]



2. From the Home screen, sort **L2** in ascending order.

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] [►] **1**
   [2nd] [STAT] **2** [)] [ENTER]

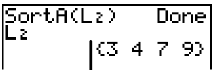


3. If desired, display **L2** on the Home screen or in the List editor to see the new order.

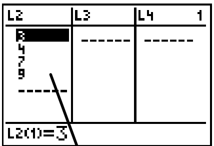
   [2nd] [STAT] **2** [ENTER]

   – or –

   [LIST]
   [►] (as necessary)



On the Home screen



In the List editor

Sorting Multiple Lists

You can specify more than one list when using **SortA(** and **SortD(**. In this case, the first list specified is the *independent* one; any following lists are *dependent*.

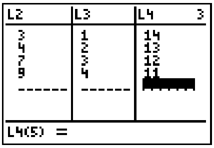
The calculator sorts the *independent* list first, and then sorts all the *dependent* lists by placing their elements in the same order as their corresponding elements in the *independent* list. This allows you to keep sets of related data in the same order when you sort lists.

SortA(*indpntlist*,*dependlist1*,*dependlist2*…**)**
SortD(*indpntlist*,*dependlist1*,*dependlist2*…**)**

Define **L2={3,4,7,9}** (independent), **L3={1,2,3,4}** (dependent), and **L4={14,13,12,11}** (dependent), and sort all three in descending order.

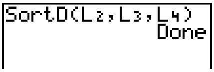
1. Define **L2, L3,** and **L4** in the List editor.

   [LIST]



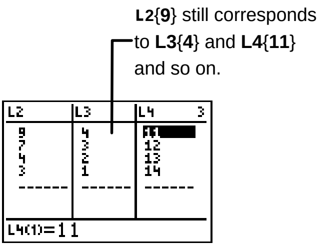
2. From the Home screen, sort the lists in descending order.

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] [▶] **2**
   [2nd] [STAT] **2** [,]
   [2nd] [STAT] **3** [,]
   [2nd] [STAT] **4** [)] [ENTER]



3. If desired, display the elements in the List editor to see the new order.

   [LIST]
   [▶] (as necessary)



ClrList [2nd] [STAT] [▸] 3

Clears all items in specified list(s) from the Home screen.

ClrList *list1*[,*list2*,*list3*,...]

From the Home screen, clear **L1** and **L2**.

[2nd] [QUIT] [CLEAR]
[2nd] [STAT] [▸] **3**
[2nd] [STAT] **1** [,] [2nd] [STAT] **2**
[ENTER]

dim([2nd] [STAT] [▸] 4

Use **dim(** from the Home screen to return the dimension (number of elements) of a defined list, to create a new list with a specified number of elements, or to change the dimension of a defined list.

When creating a new list with a specified dimension, you can assign a length from 1 to 999. The elements are set to zeros.

When changing the dimension of a defined list, all existing elements in the defined list within the new dimension are not changed.

- If you are increasing the number of elements, extra list elements are filled by 0.
- If you are decreasing the number of elements, all existing elements in the defined list outside the new dimension are deleted.

To return the dimension of a list:

dim(*list***)**

To create a new list with a specific dimension:

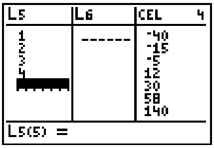
dimension# [STO▶] **dim(***newList***)**

To change the dimension of an existing list:

newDimension# [STO▶] **dim(***list***)**

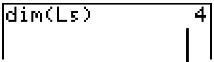
Define **L5={1,2,3,4}** in the List editor.

[LIST]



From the Home screen, return the dimension of **L5**.

[2nd] [QUIT] [CLEAR]
[2nd] [STAT] [▶] **4**
[2nd] [STAT] **5** [)] [ENTER]



There are 4 elements in **L5**.

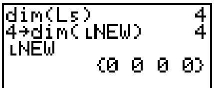
Create a new list, **ʟNEW**, with 4 elements.

1. Define the list on the Home screen.

   4 [STO▶] [2nd] [STAT] [▶] **4**
   [2nd] [STAT] [▶] **9**
   [2nd] [TEXT]
   N [ENTER] **E** [ENTER]
   W [ENTER] **Done** [ENTER] [)]
   [ENTER]

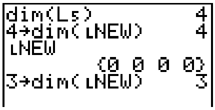
2. Display the elements in **ʟNEW** on the Home screen, if desired.

   [2nd] [STAT] **NEW** [ENTER]
   [ENTER]



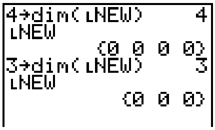
1. Change the dimension of **ʟNEW** to 3 elements.

   3 [STO▸] [2nd] [STAT] [▸] **4**
   [2nd] [STAT] **NEW** [ENTER]
   [)] [ENTER]



2. Display the elements in **ʟNEW**, if desired.

   [2nd] [STAT] **N E W** [ENTER]
   [ENTER]



ΔList([2nd] [STAT] [▸] 5

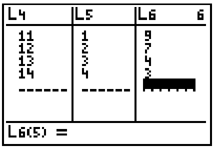
ΔList((delta list) returns a list containing the differences between consecutive elements in a list. It subtracts the first element in the list from the second element, subtracts the second element from the third, and so on. The resulting list is always one element shorter than the original list.

ΔList(*list***)**

Define **L6={9,7,4,3}** and calculate its **ΔList**.

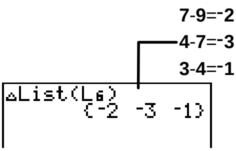
1. Enter the elements in the List editor.

   [LIST]



2. From the Home screen, calculate **ΔList** for **L6.**

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] [▸] **5**
   [2nd] [STAT] **6** [)] [ENTER]



Select([2nd] [STAT] [▶] 6

This instruction is used to select a certain portion of an existing Scatter or xyLine stat plot, both of which contain an *XList* and a *YList*. Before you can use **Select(**, you must define and select (turn on) the statistical plot you want to use; otherwise, you get an error message. For a detailed explanation on setting up Scatter and xyLine plots, see Chapter 6: Statistical Plots.

From the Home screen, enter **Select(** followed by two list names, *XList* and *YList*. These list names are where you want to store the selected data points. All *X*-values are stored in the first list and all *Y*-values are stored in the second list.

XList and *YList* can be the same two lists as the ones which set up the stat plot, or you can enter new list names. If you choose to enter new list names, entering the list signifier (**ʟ**) (found under the [2nd] [STAT] **OPS** menu) is optional. Enter the new list names using the Text editor ([2nd] [TEXT]).

Select(*XList*,*YList***)**

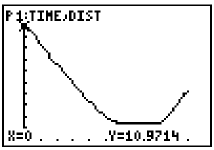
The calculator displays the stat plot and prompts you to select the left and the right bounds. The calculator then plots the selected points on the Graph screen for you to see. If desired, you can enter the List editor to see the lists with the *selected* data points.

The following example shows the steps you would follow when selecting a statistical plot. The data is acquired from a sample statistical xyLine plot. **ʟTIME** contains 94 *X*-values; **ʟDIST** contains 94 *Y*-values.

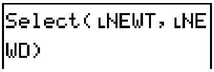
The example selects the first portion of the graph before Distance=0 and stores the *selected X*-values in **ʟNEWT** and the *selected Y*-values in **ʟNEWD**.

1. Display the graph or stat plot and determine the data points you want to select.

   [GRAPH]



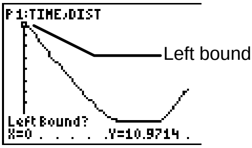
2. The **Select(** command and two new list names are entered from the Home screen.



[2nd] [STAT] [▸] **9** accesses the list signifier. [2nd] [TEXT] accesses the Text editor.

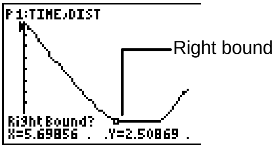
3. The left bound is chosen.

   [ENTER]

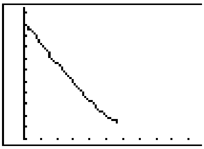


4. The right bound is chosen.

   [▸] (as necessary)
   [ENTER]



5. The plot is regraphed to include only the *selected* data points.



ʟNEWT and **ʟNEWD** now exist in the calculator's memory. To display newly selected lists in the List editor, insert them as you would insert any other list.

seq([2nd] [STAT] [▸] 7

seq(returns a list in which each element is the result of the evaluation of *expression* with regard to the *variable*. You also must specify a value range from *begin* to *end*. You can specify one optional argument, *increment*, which specifies the interval between each *variable* value used to solve *expression*.

variable need not be defined in memory. *increment* can be negative. The default value for *increment* is 1. **seq(** is not valid within expressions.

seq(*expression*,*variable*,*begin*,*end*[,*increment*]**)**

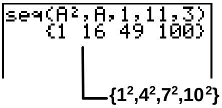
Solve *expression*, **A²**, with regard to *variable*, **A**. Use *variable* values ranging from 1 (*begin*) to 11 (*end*), and specify *increment* as 3.

1. Return to the Home screen, and clear it, if desired.

   [2nd] [QUIT] [CLEAR]

2. Enter the **seq(** expression.

   [2nd] [STAT] [▶] **7**
   [2nd] [TEXT] **A** [ENTER] [x^2] [,]
   A [ENTER] **Done** [ENTER] [,]
   1 [,] **1 1** [,] **3** [)] [ENTER]



augment([2nd] [STAT] [▶] 8

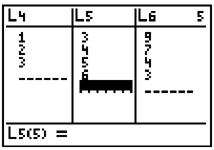
augment(combines the elements of two lists from the Home screen to create a new list. An augmented list is not saved in the calculator's memory unless you name it or store it to an existing list name. This is shown in the following example.

augment(*list1*,*list2***)**

Define **L4={1,2,3}** and **L5={3,4,5,6}** in the List editor, augment **L4** with **L5** and store the augmented list to **L6**.

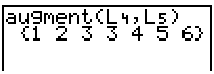
1. Define **L4** and **L5**.

   [LIST]



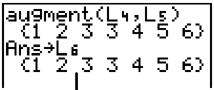
2. Return to the Home screen, and augment **L4** and **L5**.

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] [▶] **8**
   [2nd] [STAT] **4** [,]
   [2nd] [STAT] **5** [)] [ENTER]



3. Store the augmented list to **L6**.

   [2nd] [ANS] [STO▶] [2nd] [STAT] **6** [ENTER]



Press [LIST] to view **L6** in the List editor.

L (List Signifier) [2nd] [STAT] [▶] 9

The list signifier, **L**, which is not the same as the **L** from the Text editor, is especially useful in programming when you want to specify a group of numbers or text characters as a list name.

L*listname*

The list signifier does not appear in front of a list name in the List editor or in the [2nd] [STAT] **Ls** menu because it is obvious which groups of text characters or numbers are list names. Also, the list signifier is optional when entering commands that take only list names for arguments. For example,

Select(*XList*,*YList***)**

Although *XList* and *YList* are not preceded by the list signifier, the calculator interprets them as list names since no other types of arguments are accepted.

Also, when defining lists from the Home screen, the list signifier is optional.

{1,2,3}[STO▶]**ABC**

Since this command structure is only used with list names, the calculator interprets **ABC** as **LABC**.

List Commands from the Home Screen

You can create, copy, display, and edit lists directly from the Home screen. You also can perform mathematical functions on lists from the Home screen.

Creating a List

To create a list on the Home screen, you must enter the list elements surrounded by braces and store them to the list name. You can access the braces from the Text editor (2nd [TEXT]) or from the **CATALOG** (2nd [CATALOG]).

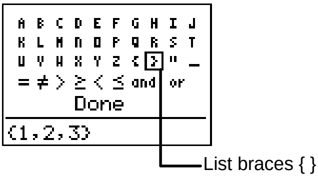
If you create a list on the Home screen, it is stored in the calculator's memory, but it won't show up in the List editor unless you specifically insert it there.

{*element1*,*element2*,…**}**STO▶*list*

Define **ʟABC={1,2,3}** on the Home screen.

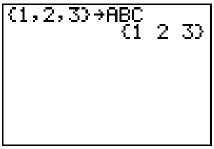
1. Enter the elements.

   2nd [QUIT] CLEAR
   2nd [TEXT]
   { ENTER **1** , **2** , **3 }**
   ENTER **Done** ENTER



2. Store to the list name.

   STO▶ 2nd [TEXT]
   A ENTER **B** ENTER
   C ENTER **Done** ENTER
   ENTER



Copying One List to Another

To copy a list on the Home screen, store it to another list name.

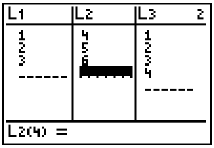
It is easiest to store the elements in the List editor. You then can review the results in the List editor. Otherwise, any lists you create on the Home screen are stored in memory, but they don't appear in the List editor unless you insert them there.

list STO▶ *newList*

Define **L1={1,2,3}** and **L2={4,5,6}** and copy **L1** to **L2**.

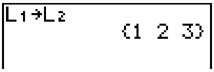
1. Enter the new elements.

   LIST



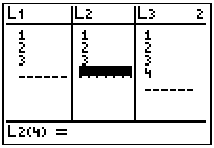
2. Return to the Home screen and copy **L1** to **L2**.

   2nd [QUIT] CLEAR
   2nd [STAT] **1** STO▶
   2nd [STAT] **2** ENTER



3. Display the copied list in the List editor.

   LIST

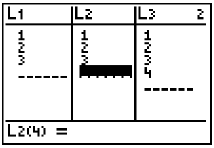


Displaying One List Element

From the Home screen, you can display one list element from a defined list.

list(*element#*)

Define **L_2={1,2,3}** in the List editor and display the second element from the Home screen.



1. Define **L_2**.

   [LIST]

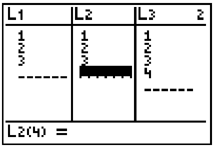
2. Display the 2nd element only.

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] **2** [(] **2** [)] [ENTER]

Inserting or Changing a List Element

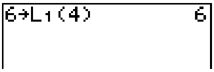
From the Home screen, you can insert or change elements in a defined list. You can only insert elements in order. For example, you can't insert a 3rd element if the 2nd and 1st elements are not defined.

Define **L_1={1,2,3}** and insert a fourth element, **6**. Then change the 4th element from **6** to **8**.



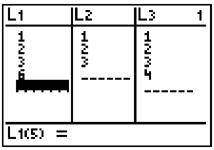
1. Define **L_1** in the List editor.

   [LIST] .



2. Return to the Home screen, and insert a 4th element, **6**.

   [2nd] [QUIT] [CLEAR]
   6 [STO▶] [2nd] [STAT] **1**
   [(] **4** [)] [ENTER]

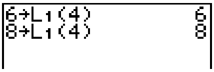


3. Display results in the List editor, if desired.

   [LIST]

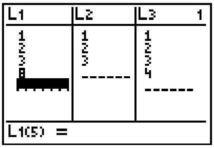
4. Change the 4th element, **6**, to **8**.

   [2nd] [QUIT] **8** [STO▶]
   [2nd] [STAT] **1** [(] **4** [)] [ENTER]



5. Display results in the List editor, if desired.

   [LIST]



Using Math Functions with Lists

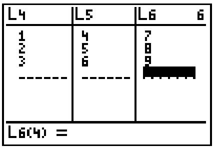
When a math function (see Chapter 2: Math Operations) is applied to a list, it is calculated for every element in the list. Therefore, the function must be valid for every element in the list.

You cannot perform a mathematical function on two lists of different sizes. For example, **{1,2,3}+{4,5,6,7}** results in an error. Mathematical rules always apply; for example, **1÷{0,1,2}** results in an error because 1 cannot be divided by 0.

Perform mathematical functions with **L5** and **L6** on the Home screen.

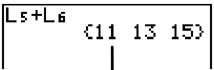
1. Define **L5={4,5,6}** and **L6={7,8,9}**.

   [LIST]



2. Return to the Home screen, and calculate **L5+ L6.**

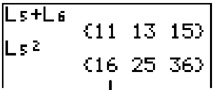
   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] **5** [+]
   [2nd] [STAT] **6** [ENTER]



{4,5,6}+{7,8,9}=
{4+7, 5+8, 6+9}=
{11,13,15}

3. Calculate **L5**^{2}.

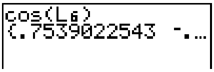
[2nd] [STAT] **5** [x^2] [ENTER]

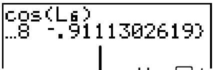


L5^2=
$\{4^2, 5^2, 6^2\}$=
{16,25,36}

4. Select **Radian** mode setting and calculate cos(**L6**).

[MODE] [▼] [▼] [▶] [ENTER]
[2nd] [QUIT] [CLEAR]
[2nd] [TRIG] **3**
[2nd] [STAT] **6** [)] [ENTER]





Use [▶] to scroll to see the entire answers.

For more information, see Chapter 11: Trigonometry.

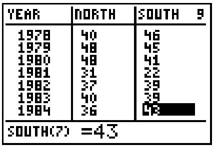
6 Statistical Plots

Steps for Defining a Stat Plot

Follow these basic steps when defining a statistical plot. You may not have to do all of them each time you graph the designated lists.

Define numerical and categorical lists in the List editor.

[LIST]

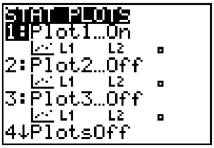


Deselect $\mathbf{Y}_n$ functions, if desired.

[2nd] [VARS] **2** **6**

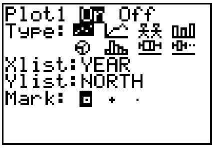
Define the stat plot by entering the **STAT PLOTS** menu and selecting **Plot1**, **Plot2**, or **Plot3**.

[2nd] [PLOT] **1**, **2**, or **3**



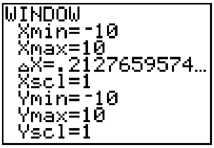
Turn on the stat plot, select the Plot Type and define all corresponding options.

[▶] [ENTER]



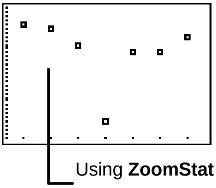
Adjust the **WINDOW** values and format, if necessary.

[WINDOW]



Graph the stat plot. Trace the plot with [TRACE], if desired.

[GRAPH]
[ZOOM] **7**
or [TRACE]



Using **ZoomStat WINDOW** values.

Defining Statistical Data in Lists

Statistical plots (stat plots) are graphical representations of data that has been stored in lists. Therefore, since you need to create your lists before you can define stat plots, review Chapter 5: Lists for information on naming and creating both numerical and categorical lists.

Note: All examples in this chapter assume that you know how to enter lists in the List editor.

Deselecting Y_n Functions

When you press [GRAPH] or a [ZOOM] command, the calculator graphs all selected **Y_n** functions (defined in the Y= editor) *and* graphs all stat plots that are defined and turned on. If you have defined and selected functions in the Y= editor and you don't want them displayed with your stat plots, deselect all defined functions with [2nd] [VARS] **2:Y-Vars** **6:FnOff**.

For more information on defining and selecting functions in the Y= editor, see Chapter 9: Function Graphing.

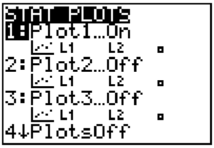
Defining a Stat Plot

Once you have data lists stored in the calculator, you need to define the stat plot. This requires two steps:

1. Press [2nd] [PLOT] to display the **STAT PLOTS** menu screen.
2. Select **1**, **2**, or **3** to enter the Stat Plot editor for **Plot1**, **Plot2**, or **Plot3**. Selecting **4** or **5** turns all stat plots off or on when you graph.

The Stat Plot Menu Screen

[2nd] [PLOT]



PlotsOff and PlotsOn [2nd] [PLOT] 4 and 5

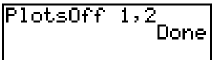
From the **STAT PLOTS** menu, you can choose to turn all stat plots off or on. This determines whether or not they are displayed on the Graph screen when you press [GRAPH] or select a [ZOOM] command. The TI-73 can graph all three stat plots at the same time, if desired. If you select either of these commands, the calculator returns you to the Home screen.

PlotsOff and **PlotsOn** accept three optional arguments, **1**, **2** or **3**, which represent their corresponding stat plot. If you do not include any arguments, the calculator automatically deselects (turns off) or selects (turns on) all three.

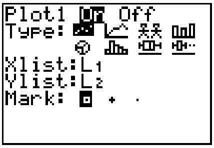
PlotsOff [**1,2,3**]
PlotsOn [**1,2,3**]

Turn off **Plot1** and **Plot2**.

[2nd] [PLOT] **4**
1 [,] **2** [ENTER]



The Stat Plot Editor

[2nd] [PLOT] **1 , 2 or 3**	

If the plot has been defined previously, that information is displayed when you select a plot number.

From the Stat Plot editor, you select (turn on) or deselect (turn off) the stat plot, and you can select one of eight plot types (represented as icons) as well as any options that go with the type.

Selecting Stat Plot Types

To select a stat plot type, display the Stat Plot editor. Use [▲] and [▼] to move to the **Type** line, and use [▶] and [◀] to highlight the individual **Type** icons. Once you have highlighted the **Type** icon that you want, press [ENTER] to select it. The options for the plot type then are displayed automatically.

Icon	Plot Type	Icon	Plot Type
	Scatter plot		Pie chart
	xyLine plot		Histogram
	Pictograph		Box plot
	Bar graph		Modified Box plot

Defining Stat Plot Options

The plot type you select determines which options you can select. Therefore, when you select a different type, the options adjust automatically, if necessary.

- To specify a list name, use the [2nd] [STAT] **Ls** menu. Highlight the list name you want with the cursor keys, and then press [ENTER]. The TI−73 inserts the name at the cursor location.
- To select an option, highlight the one you want with the cursor keys, and then press [ENTER].
- To enter a numerical value, use the number keys, and then press [ENTER].

Remember that when entering elements in a categorical list, you must surround the first element by quotation marks; they are optional for the remaining elements.

The following table includes a list of all possible options for all stat plot types. You only need to specify or select the options which apply to the stat plot type you are defining.

For option:	Do the following:
Xlist	Specify a defined numerical list.
Ylist	Specify a defined numerical list. **Ylist** must be the same length as **Xlist** and can be the same as **Xlist**. Plots which require you to specify both the **Xlist** and **Ylist** plot points from those lists as coordinate pairs.
Mark	Select one type (□ , **+**, or **•**) to specify appearance of data points or an outlier (Modified Box plot) on the graph screen.

For option:	Do the following:
CategList	Specify a defined categorical list. List dimension must be from 1 to 7 and must be the same length as all corresponding **Data Lists**.
Data List or DataList#	Specify a defined numerical list. All **Data Lists** must be the same length as the corresponding **CategList**.
Scale	Specify a number which represents the quantity of each Pictograph icon. 1≤**Scale**≤99999. **Scale** must be big enough so that it cannot be broken up into more than 7 icons. Using [ZOOM] **7:ZoomStat** to display the stat plot automatically adjusts **Scale** for you.
Vert/Hor	Select vertical or horizontal orientation for Pictograph icons or Bar graph bars.
Icons	Select one of 7 **Icons** to represent your Pictograph: 
1 2 3	Select number of bars you want graphed per category in a Bar graph. You must specify a corresponding **Data List** for each bar included in the graph.
Number/Percent	Select whether you want the values in **DataList** to be displayed as numbers or converted and displayed as percentages in a Pie chart.
Freq (optional) Default=1	Specify a frequency list that tells the calculator how many times each data point in **Xlist** occurs. **Freq** must have the same number of elements as **Xlist**.

Adjusting Window Values and Format

If you press [GRAPH] to display all selected stat plots, sometimes you see a blank screen. Try adjusting your viewing window. The easiest way to do this is with the [ZOOM] **7:ZoomStat** command. This adjusts the viewing window automatically so that all points of all turned on stat plots are visible. To adjust window values manually, press [WINDOW].

In addition, the calculator automatically selects the **AxesOff** option ([2nd] [FORMAT]) for Pictograph, Bar graph, Pie chart stat plots. However, any other selected options on the [2nd] [FORMAT] screen still apply to stat plots (as they do with function graphs).

For more information on adjusting **WINDOW** values and formatting the Graph screen, see Chapter 9: Function Graphing.

Displaying the Stat Plot

Press [GRAPH] to display a stat plot. (Pressing [GRAPH] also displays any **Y_n** functions that are defined and selected.) Once you have a plot displayed, you can press [TRACE] and use [▶] and [◀] to move from point to point.

If you have more than one plot turned on at the same time, you can trace all the points of each plot. Use [▲] and [▼] to move from plot to plot.

Stat Plot Examples

The following examples assume that all **Y_n** functions are deselected (turned off) ([2nd] [VARS] **2:Y-Vars 6:FnOff**).

Scatter Plot and xyLine Plot

Scatter plots () and xyLine plots () are especially useful for plotting data over a period of time to indicate trends. An xyLine plot () functions exactly like the Scatter plot, except that it connects the data points with a line.

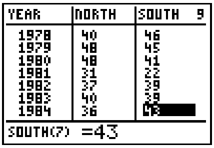
For the years 1978 -1984, determine in which baseball league, North or South, the homerun leader tends to hit more home runs. Use Scatter plots to find your solution.

Year	Home Runs		Year	Home Runs	
	NORTH	SOUTH		NORTH	SOUTH
1978	40	46	1982	37	39
1979	48	45	1983	40	39
1980	48	41	1984	36	43
1981	31	22			

1. Create three lists in the List editor, **YEAR**, **NORTH**, and **SOUTH**.

   [LIST]

   For more information on entering lists, see Chapter 5: Lists.



2. Turn off all stat plots.

   [2nd] [PLOT] **4** [ENTER]

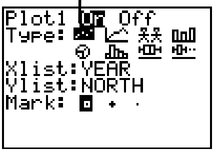
3. Display the **STAT PLOTS** menu.

   [2nd] [PLOT]

4. Define **Plot1** as a Scatter plot as shown to the right.

   1 [ENTER] [▼] [ENTER] [▼]
   [2nd] [STAT] **YEAR** [ENTER]
   [▼] [2nd] [STAT] **NORTH**
   [ENTER] [▼] [ENTER]

Select the icon.

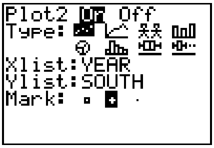


5. Display the **STAT PLOTS** menu.

   [2nd] [PLOT]

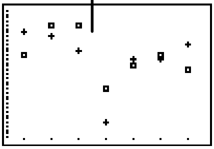
6. Define **Plot2** as shown to the right.

   2 [ENTER] [▼] [ENTER] [▼]
   [2nd] [STAT] **YEAR** [ENTER]
   [▼] [2nd] [STAT] **SOUTH**
   [ENTER] [▼] [▶] [ENTER]



7. Display the stat plots using the **ZoomStat** command.

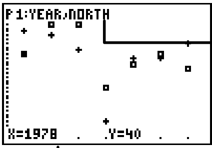
   [ZOOM] **7**



Using different marks helps you distinguish between **Plot1** and **Plot2**.

8. Trace the Scatter plots to find the solution to the question.

   [TRACE]
   [◀] and [▶] (to trace point to point)
   [▲] and [▼] (to move from plot to plot)



The plot that is being traced

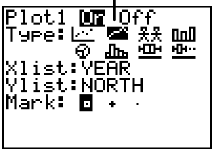
The X and Y coordinates

Solution	From 1978-1984, the North League's home run hitter led in 4 of the 7 years.

9. Redefining **Plot1** as an xyLine plot makes it even easier to follow the trends of its data.

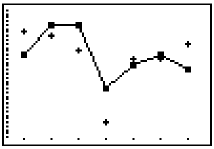
   [2nd] [PLOT] **1** [▼] [▶] [ENTER]

Select the icon.



10. Display **Plot1** and **Plot2** using the **ZoomStat** command. Trace, if desired.

    [ZOOM] **7**
    [TRACE] (optional)



Pictograph

In a Pictograph, an icon symbolizes the quantities being represented. Pictographs are useful for observing changes in quantity over time. They also can illustrate comparisons between similar situations.

The calculator displays no more than seven Pictograph icons for up to seven categories on the screen. Therefore, if **Scale** is not big enough (meaning that **Data List** is broken up by more than seven icons), you get an **INVALID DIM** error.

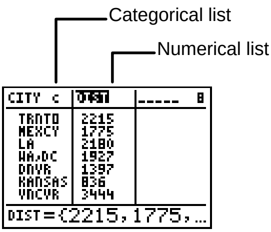
If an element in **Data List** is too large to fit the maximum scale (99999) so that the calculator can't make all icons fit in one screen, you get a **DOMAIN** error.

For your geography class, you want to compare distances (in kilometers) between Dallas, Texas, and seven other cities in North America. Use a vertical Pictograph to display your results.

City	km	City	km
Toronto, ON	2215	Denver, CO	1397
Mexico City, MX	1775	Kansas City, KS	836
Los Angeles, CA	2180	Vancouver, BC	3444
Washington, DC	1927		

1. Create two lists in the List editor, **CITY** and **DIST**. Remember to surround the first categorical list element with quotation marks (found in the Text editor).

   [LIST]



For more information on entering lists, see Chapter 5: Lists.

2. Turn off all stat plots.

   [2nd] [PLOT] **4** [ENTER]

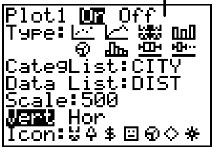
3. Display the **STAT PLOTS** menu.

   [2nd] [PLOT]

4. Define **Plot1** as a Pictograph as shown to the right.

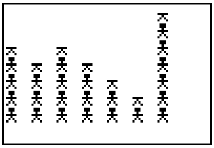
   1 [ENTER] [▼] [▶] [▶] [ENTER] [▼] [2nd] [STAT] **CITY** [ENTER] [▼] [2nd] [STAT] **MILES** [ENTER] [▼] **5 0 0** [▼] [ENTER] [▼] [ENTER]

Select the icon.



5. Display the stat plots.

   [GRAPH]

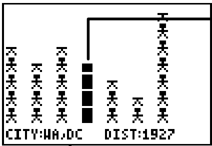


6. Trace, if desired.

   [TRACE]

   As you press [▶] and [◀], the calculator highlights whole columns. Both list names and list values are displayed at the bottom of the screen.

Highlighted column



Dallas, TX, to Washington, DC is 1,927 km.

Bar Graph

A Bar graph plots a group of up to three data lists (converted to bars) for comparison among one category. Bar graphs are especially useful for comparing data lists (especially when organized in categories) over a period of time.

The calculator adjusts all bars so that they fit within the graphing screen. Therefore, the data list with the largest values is scaled to fit the screen, and then all other bars are graphed relative to it. Each element in **CategList** defines a category. You can define up to seven categories with up to three data bars per category.

The **Xscl WINDOW** value specifies the range of values for each interval of a Bar graph. The **Yscl WINDOW** value specifies the height of a bar in a Bar graph; in other words, it acts as your bar scale. To adjust **Xscl** and **Yscl** manually, press [WINDOW] and enter the new values with the number keys. For more information about setting **WINDOW** values, see Chapter 9: Function Graphing.

If you want the calculator to adjust the **WINDOW** values for you automatically, press [ZOOM] **7:ZoomStat**.

Graph the data lists from the Scatter plot baseball example as a vertical Bar graph (see that section in this chapter, if necessary). Assign **ʟYEAR** as **CategList**, **ʟNORTH** as **DataList1** and **ʟSOUTH** as **DataList2**. Ignore **DataList3**. (By default, **L3** is assigned to **DataList3**, but if another list name is assigned you don't need to change it.)

1. Turn off all stat plots.

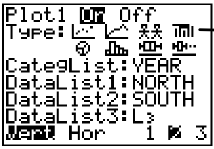
   [2nd] [QUIT] [CLEAR]
   [2nd] [PLOT] **4** [ENTER]

2. Display the **STAT PLOTS** menu.

   [2nd] [PLOT]

3. Define **Plot1** as a Bar graph as shown to the right.

   1 [ENTER] [▼] [▶] [▶] [▶]
   [ENTER]



Select the icon.

4. Specify **CategList**, **DataList1**, and **DataList2**.

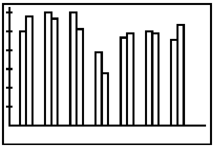
   [▼] [2nd] [STAT]
   Y E A R [ENTER] [▼]
   [2nd] [STAT] **NORTH**
   [ENTER] [▼] [2nd] [STAT]
   SOUTH [ENTER]

5. Select **Vert** and **2**, if necessary.

   [▼] [▼] [ENTER] [▶] [▶] [▶]
   [ENTER]

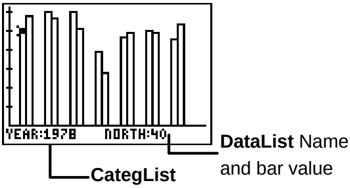
6. Display the stat plots.

   [GRAPH]



7. Trace the Bar graph, if desired.

   [TRACE]
   [◀] and [▶] (to trace bar to bar)



Pie Chart

A Pie chart is used to compare parts of a whole. The area of a "pie piece" is proportional to the part of 100% that it represents. You can display up to seven "pie pieces."

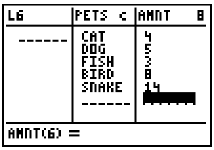
To trace the Pie chart with [TRACE], use [▶] to trace clockwise and [◀] to trace counterclockwise.

Keisha owns 4 cats, 5 dogs, 3 fish, 8 birds, and 14 snakes. Use a percentage Pie chart to illustrate this.

1. Create two lists in the List editor, **PETS** and **AMNT**.

   [LIST]

   For more information on entering lists, see Chapter 5: Lists.



2. Turn off all stat plots.

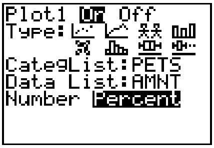
   [2nd] [PLOT] **4** [ENTER]

3. Display the **STAT PLOTS** menu and select **Plot1**.

   [2nd] [PLOT] **1** [ENTER]

4. Define **Plot1** as a Pie chart as shown to the right.

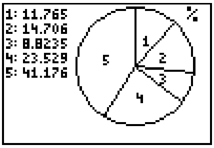
   [▼] [▶] [▶] [▶] [▶] [ENTER] [▼]
   [2nd] [STAT] **PETS** [ENTER] [▼]
   [2nd] [STAT] **AMNT** [ENTER]
   [▼] [▶] [ENTER]



5. Display the stat plot.

   [GRAPH]

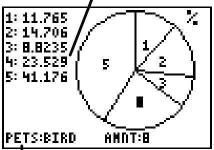
A percentage Pie chart



6. Trace the Pie chart, if desired.

   [TRACE]
   [▶] and [◀] (to trace from section to section)

8 birds is 23.529% of the total pets owned.



CategList

Data List name and section's numerical value

Histogram

Histograms are useful for representing data grouped in intervals, and it plots the data's frequency of occurrence for each interval.

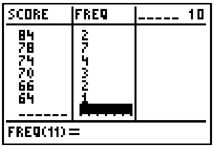
Thirty students recently took a math test. All scores between 100-90 are considered an A, 89-80 as a B, 79-70 as a C, 69-60 as a D, 59-0 as an F. Use a Histogram to show the scores grouped by their letter grade.

SCORE	{99,96,92,88,84,78,74,70,66,64}
FREQ	{1,2,3,5,2,7,4,3,2,1}

1. Create two lists in the List editor, **SCORE** and **FREQ**.

   [LIST]

   For more information on entering lists, see Chapter 5: Lists.



2. Turn off all stat plots.

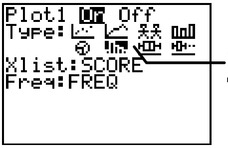
   [2nd] [PLOT] **4** [ENTER]

3. Display the **STAT PLOTS** menu.

   [2nd] [PLOT]

4. Define **Plot1** as a Histogram as shown to the right.

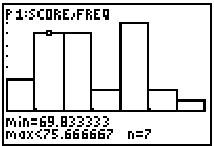
   1 [ENTER] [▼] [▶] [▶] [▶] [▶]
   [▶] [ENTER] [▼] [2nd] [STAT]
   SCORE [ENTER] [▼]
   [2nd] [STAT] **FREQ** [ENTER]



Select the icon.

5. Display the stat plot using the **ZoomStat** command and trace the Histogram.

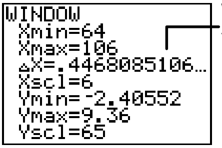
   [ZOOM] **7** [TRACE]
   [◀] and [▶] (to trace bar to bar)



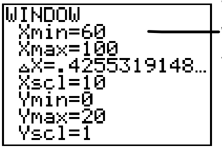
7 scores fall between the min and max.

6. Adjust the graphing window so that the data is grouped in intervals of 10 and so that the lowest test score is 60 and the highest is 100.

   [WINDOW] **6 0** [▼] **1 0 0** [▼] [▼]
   1 0 [▼] **0** [▼] **2 0** [▼] **1**



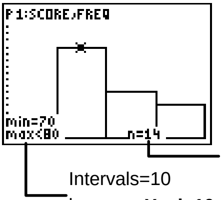
WINDOW values from **ZoomStat**



The adjusted **WINDOW** values

7. Graph and trace the adjusted Histogram.

   [GRAPH] [TRACE]
   [▶] and [◀] (to trace bar to bar)



14 scores fall between 70-80.

Intervals=10 because **Xscl=10**

```
Solution | 3 students received a D.
         | 14 students received a C.
         | 7 students received a B.
         | 6 students received an A.
```

Box Plot

A Box plot illustrates median applications of a data list. Lines on the plot, called whiskers, extend from the minimum data point in the set (**minX**) to the first quartile median point (**Q_1**) and from the third quartile median point (**Q_3**) to the maximum point (**maxX**). The middle vertical line is the median (**Med**) of all the data points.

The first quartile contains all data points between **minX** and **Med**; the third quartile contains all data points between **Med** and **maxX**.

When two Box plots are plotted, the first one plots at the top of the screen and the second plots in the middle. When three are plotted, the first one plots at the top, the second in the middle, and the third at the bottom.

Xmin and **Xmax** specify minimum and maximum *X*-axis values when a Box plot is displayed on the Graph screen. Box plots ignore **Ymin** and **Ymax** values. To adjust **Xmin** and **Xmax** manually, press [WINDOW] and enter the new values with the number keys. If you want the calculator to adjust the window values for you automatically, press [ZOOM] **7:ZoomStat**.

For more information about setting **WINDOW** values, see Chapter 9: Function Graphing.

Graph the test scores data from the Histogram example as a Box plot. (See previous section, if necessary.)

1. Turn off all stat plots.

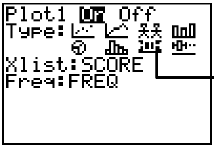
   [2nd] [PLOT] **4** [ENTER]

2. Display the Stat Plots menu.

   [2nd] [PLOT]

3. Define **Plot1** as a Box plot as shown to the right.

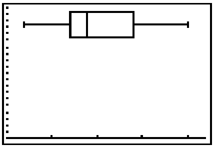
   1 [ENTER] [▼] [▶] [▶] [▶] [▶]
   [▶] [▶] [ENTER] [▼] [2nd] [STAT]
   SCORE [ENTER] [▼]
   [2nd] [STAT] **FREQ** [ENTER]



Select the icon.

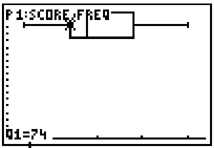
4. Display the stat plot using the **ZoomStat** command.

   [ZOOM] **7**



5. Trace the Box plot.

   [TRACE]
   [◀] and [▶] (to trace point to point)



Q1 Median point and its value

Modified Box Plot

The Modified Box plot functions exactly like the Box plot, except it separates outliers from the plot. Outliers are those data points which are 1.5∗Interquartile Range beyond the quartiles. The Interquartile Range is defined as the difference between the third quartile, Q_3, and the first quartile, Q_1.

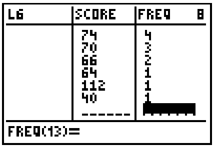
Outliers are plotted individually beyond the whisker, using the **Mark** you select from the Stat Plot editor. Outliers are included in plot traces with [TRACE].

Graph the test scores data from the Histogram and Box plot examples as a Modified Box plot. (See those sections, if necessary.) However, adjust **SCORE** and **FREQ** by inserting two outlier data points: 112 and 40, both at a frequency of 1.

1. Edit **SCORE** and **FREQ** in the List editor.

   [LIST]

   For more information on entering lists, see Chapter 5: Lists.



2. Turn off all **STAT PLOTS.**

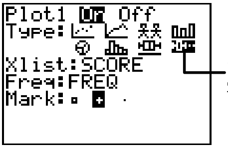
   [2nd] [PLOT] **4** [ENTER]

3. Display the **STAT PLOTS** menu.

   [2nd] [PLOT]

4. Define **Plot1** as a Modified Box plot as shown to the right.

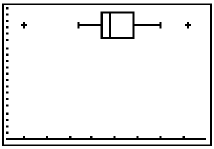
   1 [ENTER] [▼] [▶] [▶] [▶] [▶] [▶] [▶] [▶] [ENTER] [▼] [2nd] [STAT] **SCORE** [ENTER] [▼] [2nd] [STAT] **FREQ** [ENTER] [▼] [▶] [ENTER]



Select the ·□·· icon.

5. Display the stat plot using the **ZoomStat** command.

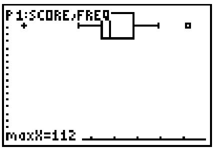
   [ZOOM] **7**



6. Trace the plot, if desired.

   [TRACE]

   [◀] and [▶] (to trace point to point)

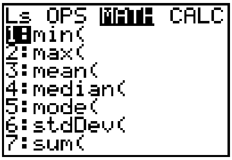


7 Statistical Analyses

The [2nd] [STAT] MATH Menu

The [2nd] [STAT] **MATH** menu allows you to calculate statistical analyses with lists (see chapter 5: Lists).

[2nd] [STAT] [▸] [▸]



1:min(	Returns the minimum of two real numbers, lists, or expressions.
2:max(	Returns the maximum of two real numbers, lists, or expressions.
3:mean(	Returns the calculated average of the values in a list.
4:median(	Returns the middle value occurring in a list.
5:mode(	Returns the most frequently occurring element in a list.
6:stdDev(	Returns the standard deviation of the elements in a list.
7:sum(	Returns the sum of the elements in a list.

min(and max([2nd] [STAT] [▸] [▸] 1 and 2

These are identical to the **min(** and **max(** commands found on the [MATH] **NUM** menu.

min((minimum) returns the smaller of two *values* or the smallest element in one *list*. *value* can be a real number, expression or a list.

If both arguments are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is compared with each element of the list, and a list is returned.

> **min(***valueA*,*valueB***)**
> **min(***list***)**

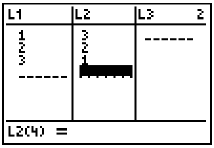
max((maximum) functions exactly like **min(**, but it always returns the *larger* of two *values* or the largest element in a list. Simply substitute **max(** in place of **min(** in the syntax models above.

Compare **L1** and **L2** to find the **min(** and **max(**. **L1={1,2,3}**, and **L2={3,2,1}**.

1. Define two lists in the List editor, **L1** and **L2**.

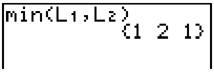
   [LIST]

   For more information on entering lists, see Chapter 5: Lists.



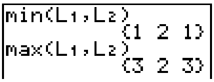
2. Find the list minimums.

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] [▶] [▶] **1**
   [2nd] [STAT] **1** [,]
   [2nd] [STAT] **2** [)] [ENTER]



3. Find the list maximums.

   [2nd] [STAT] [▶] [▶] **2**
   [2nd] [STAT] **1** [,]
   [2nd] [STAT] **2** [)] [ENTER]



mean(, median(, and mode(
[2nd] [STAT] [►] [►] 3, 4, and 5

median(returns the median (the middle element) of *list* when the elements, even if the list elements are not arranged in numerical order. With an even number of elements, the calculator returns the average of the two middle elements.

mean(returns the mean (mathematical average) of *list*. **mode(** returns the mode (element which occurs most frequently) of *list*.

If a second list, *freq*, is specified, it is interpreted as the frequency of the elements in the first list. *list* and *freq* must have the same number of elements. If *freq* is not included, then the default is 1 and every element in the first list is only counted once.

mean(*list*[,*freq*]**)**
median(*list*[,*freq*]**)**
mode(*list*[,*freq*]**)**

Calculate David's final course average for his math class.

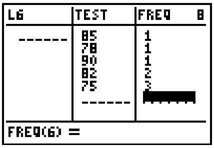
He received an 85 on Test 1, a 78 on Test 2, and a 90 on Test 3. He received an 82 on his Midterm Exam and a 75 on his Final Exam.

Tests count 1 time, the Midterm counts 2 times, and the Final Exam counts 3 times.

ʟTEST	{85,78,90,82,75}
ʟFREQ	{1,1,1,2,3}

1. Create two lists in the List editor, **TEST** and **FREQ**.

   [LIST]



For more information on entering lists, see Chapter 5:Lists.

2. Return to the Home screen, and calculate the average of the test scores.

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] [▶] [▶] **3**
   [2nd] [STAT] **TEST**
   [ENTER] [,] [2nd] [STAT]
   FREQ [ENTER] [)] [ENTER]

```
mean(ʟTEST,ʟFREQ
)
           80.25
```

Solution	David's final course average is 80.25.

stdDev([2nd] [STAT] [▶] [▶] 6

stdDev(returns the standard deviation of *list*. If a second list, *freq*, is specified, it is interpreted as the frequency of the elements in the first list. *list* and *freq* must have the same number of elements.

stdDev(*list*,*freq*[,*type*]**)**

type=**0** (population standard deviation) or **1** (sample population deviation). If *type* is not specified, the calculator returns sample population deviation.

Find the population standard deviation of **ʟTEST** (from the previous example). Use **ʟFREQ** as your *freq*.

   [2nd] [STAT] [▶] [▶] **6**
   [2nd] [STAT] **TEST** [ENTER]
   [,] [2nd] [STAT] **FREQ**
   [ENTER] [,] **0** [)] [ENTER]

```
stdDev(ʟTEST,ʟFR
EQ,0)
      5.14174095
```

sum([2nd] [STAT] [►] [►] 7

sum((summation) returns the sum of all elements in *list*. Specify the additional optional arguments to return the sum of the range of elements between *start* and *end*. *start* and *end* represent element places, not the element values.

To add the entire list:

sum(*list***)**

To add the range of elements from *start* to the last element in *list*:

sum(*list*,*start***)**

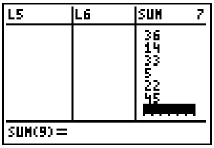
To add the range of elements between *start* and *end*:

sum(*list*,*start*,*end***)**

Find the sum of **ʟSUM** between elements 4 and 6, where **ʟSUM={3,10,36,14,33,5,22,45}**.

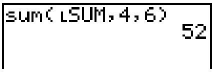
1. Create a list, in the List editor, **SUM**.

   [LIST]

   For more information on entering lists, see Chapter 5: Lists.

2. Return to the Home screen, and calculate the partial list sum.

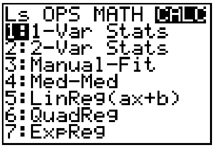
   

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] [►] [►] **7**
   [2nd] [STAT] **SUM** [ENTER]
   [,] **4** [,] **6** [)] [ENTER]

The [2nd] [STAT] CALC Menu

The [2nd] [STAT] **CALC** menu allows you to calculate statistical analyses on lists. When you choose an item from the menu, the calculator returns a list of statistical variables. Following the **1-Var Stats** and **2-Var Stats** explanation, a list and definition of all possible statistical variables is provided.

[2nd] [STAT] [▸] [▸] [▸] –or–
[2nd] [STAT] [◂]



1:1-Var Stats	Calculates 1-variable statistics.
2:2-Var Stats	Calculates 2-variable statistics.
3:Manual-Fit	Allows user to fit a line manually to plotted data.
4:Med-Med	Calculates a Median-Median line for the plotted data.
5:LinReg(ax+b)	Fits a linear model to plotted data.
6:QuadReg	Fits a quadratic model to plotted data.
7:ExpReg	Fits an exponential model to plotted data.

Using Frequency Lists with [2nd] [STAT] CALC Menu Items

For all menu items, you can specify a second list, *freq*, which is interpreted as the frequency of the elements in the first list. Each element in *freq* must be ≥ 0, and at least one element must be > 0.

Non-integer *freq* elements are valid. This is useful when entering frequencies expressed as percentages or parts that add up to 1. However, if *freq* contains non-integer frequencies, **Sx** and **Sy** (sample standard deviation) are undefined, and values are not displayed for **Sx** and **Sy** in the statistical results.

1-Var Stats and 2-Var Stats [2nd] [STAT] [◄] 1 and 2

1-Var Stats (one-variable statistics) analyzes data from one list with one measured variable (*X*). **1-Var Stats** accepts two optional arguments, *XList* and *freq*. If *XList* is not specified, the default list name is **L1**.

1-Var Stats [*XList*,*freq*]

2-Var Stats (two-variable statistics) analyzes paired data from two lists with two measured variables, *X*, the independent variable, and *Y*, the dependent variable. **2-Var Stats** accepts three optional arguments, *XList*, *YList*, and *freq*. If *XList* and *YList* are not specified, the default list names are **L1** and **L2**.

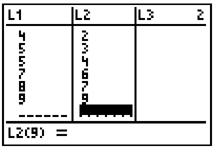
2-Var Stats [*XList*,*YList*,*freq*]

Find the **1-Var Stats** for **L1**, where **L1={1,3,4,5,5,7,8,9}**. Use **L2** as *freq*, where **L2={1,4,2,3,4,6,7,9}**.

1. Define two lists in the List editor, **L1** and **L2**.

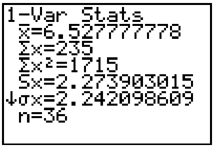
   [LIST]

   For more information on entering lists, see Chapter 5: Lists.

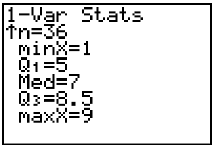


2. Return to the Home screen, and calculate the **1-Var Stats** for the lists.

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] [◄] **1**
   [2nd] [STAT] **1** [,]
   [2nd] [STAT] **2** [ENTER]



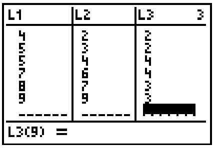
Press [▲] and [▼] to scroll all results.



Find the 2-Var Stats for **L1** (*XList*) and **L2** (*YList*), where **L1={1,3,4,5,5,7,8,9}** and **L2={1,4,2,3,4,6,7,9}**. Use **L3** as *freq*, where **{L3=1,2,2,2,4,4,3,3}**.

1. Define the three lists in the List editor, **L1**, **L2**, and **L3**.

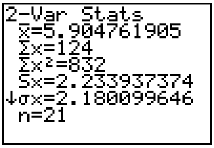
   [LIST]



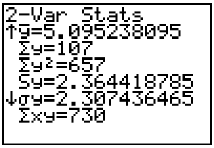
For more information on entering lists, see Chapter 5: Lists.

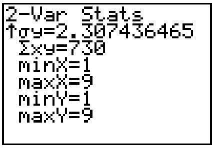
2. Return to the Home screen, and calculate the **2-Var Stats** for the lists.

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] [◄] **2**
   [2nd] [STAT] **1** [,]
   [2nd] [STAT] **2** [,]
   [2nd] [STAT] **3** [ENTER]



Press [▲] and [▼] to scroll all results.





What Do the Results Mean?

1-Var Stats and **2-Var Stats** variables are calculated and stored as indicated below. To access these variables for use in expressions, press [2nd] [VARS] **3:Statistics** and select the appropriate menu. If you edit a list or change the type of analysis, all statistical variables are cleared.

Variables	Definition	VARS Menu
$\bar{x}$ or $\bar{y}$	Mean of all **x** or **y** values.	**XY**
Σ**x** or Σ**y**	Sum of all **x** values or **y** values.	Σ
Σx^2 or Σy^2	Sum of all x^2 values or y^2 values.	Σ
Sx or **Sy**	Sample standard deviation of **x** or **y.**	**XY**
σ**x** or σ**y**	Population standard deviation of **x** or **y.**	**XY**
n	Number of **x** or **x,y** data points.	**XY**
minX **minY**	Minimum of **x** values or **y** values.	**XY**
maxX **maxY**	Maximum of **x** values or **y** values.	**XY**
Σ**xy**	Sum of **x**∗**y** for all **xy** pairs in two lists.	Σ
Q_1	Median of the elements between **minX** and **Med** (1st quartile). Only calculated for **1-Var Stats**.	**PTS**
Med	Median of all data points.	**PTS**
Q_3	Median of the elements between **Med** and **maxX** (3rd quartile). Only calculated for **1-Var Stats**.	**PTS**

Variables	Definition	VARS Menu
r	Correlation coefficient	**EQ**
r^2 or R^2	Coefficient of determination	**EQ**
RegEQ	Regression equation	**EQ**
x1,y1,x2,y2, x3,y3	Summary points	**PTS**
a, b, c	Regression/ fit coefficients	**EQ**

n (number of data points)

n=number of **x** data points in a **1-Var Stats** analysis or the number of **x** and **y** data points in a **2-Var Stats** analysis. Since both variable lists always have the same number of list elements in **2-Var Stats**, **n** for **x** is always equal to **n** for **y**. Therefore, **n** applies to both the **x** and **y** analyses.

freq (Frequency Lists)

If *freq* is specified, **n** is equal to the sum of the elements in that list. For example, if the *freq* is {2,2,3,1,2}, **n**={2+2+3+1+2}=10.

Q_1, Q_3, and Med

Q_1, **Q_3**, and **Med** are undefined if the *freq* contains non-integer values. They also are not calculated if the *freq* contains a value larger than 99.

RegEQ

The calculator stores the most recently generated regression equation (see [2nd] [STAT] **CALC** menu items 3–7) to the variable, **RegEQ**. If, for example, you execute **5: LinReg(ax+b)**, but you don't initially store **RegEQ** to a **Y_n** variable, you can later insert **RegEQ** into the Y= editor. The calculator graphs the regression equation when it is selected.

If the frequency for an element or data pair is 0, the element or data pair is ignored in the calculation.

Manual-Fit [2nd] [STAT] [◄] 3

Manual-Fit allows you to fit a line to plotted data on the Graph screen manually (as opposed to the calculator automatically drawing it for you). You can execute **Manual-Fit** from either the Graph screen or the Home screen.

From the Graph screen, select **Manual-Fit,** and then draw the line (steps provided below). The linear equation in the form **y=ax+b** is shown at the top of the Graph screen. You can use the cursors to adjust the line, if necessary, and the **a** and **b** equation values change accordingly.

From the Home screen, **Manual-Fit** accepts one optional argument, *Yn*. The calculator stores to *Yn* (in the Y= editor) the **ax+b** equation that manually fits the plotted data. To access the $\mathbf{Y_n}$ variables, press [2nd] [VARS] **2**.

Manual-Fit *Yn*

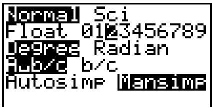
From either the Home screen or the Graph screen or Program editor, select **Manual-Fit** after you have plotted the stat plot. To draw the **Manual-Fit** line:

1. Position the cursor at the beginning of the line segment that you want to draw, and then press [ENTER].
2. As you press the cursor keys, the line is drawn and the slope is adjusted. When you have matched the plotted points as desired, press [ENTER].
3. The line segment is drawn across the entire screen and the **ax+b** equation is shown at the top of the Graph screen.
4. Continue to adjust the line's slope with [▲] and [▼], and the y-intercept with [◄] and [►], if desired.
5. If you specified a $\mathbf{Y_n}$ variable on the Home screen, you can view the selected and defined equation in the Y= editor ([Y=]). If you no longer want to view the Manual-Fit line, deselect it in the Y= editor by highlighting the **=** and pressing [ENTER].

Graph a scatter plot for **L1** and **L2**, where **L1={1,3,4,5,5,7,8,9}** and **L2={1,4,2,3,4,6,7,9}**, and use **Manual-Fit** to draw a line through the points.

1. Set Decimal Notation mode to **2**, if desired.

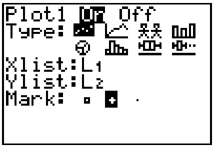
   [MODE] [▼] [▶] [▶] [▶] [ENTER]



2. After entering the lists, define **Plot1** as a scatter plot using **L1** and **L2**, as shown to the right.

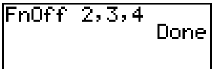
   [2nd] [PLOT] [ENTER]

   For more information on defining stat plots, see Chapter 6: Statistical Plots.



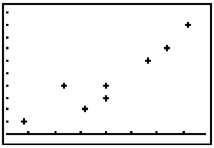
3. Turn off **Y₂**, **Y₃**, and **Y₄**, if they have been previously defined and selected.

   [2nd] [QUIT] [CLEAR]
   [2nd] [VARS] **2** **6**
   2 [,] **3** [,] **4** [ENTER]



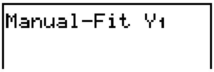
4. Plot **L1** and **L2**.

   [ZOOM] **7**



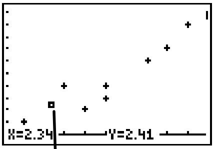
5. From the Home screen, assign the **Manual-Fit** (ax+b) line to **Y₁**.

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] [◀] **3**
   [2nd] [VARS] **2** **1** [ENTER]



6. Move the cursor to the beginning point of line.

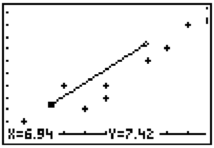
   [▶] [◀] [▲] [▼] (as necessary)
   [ENTER]



Cursor changes to a small box.

7. Move the cursor to the end point of line.

   [▶] [◀] [▲] [▼] (as necessary)



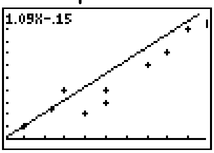
8. Draw the line.

   [ENTER]

9. Adjust line with cursor keys, if necessary.

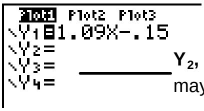
   [▶] [◀] [▲] [▼]
   [ENTER] (when finished)

ax+b



10. View the equation in the Y= editor, if desired.

    [Y=]



Y$_2$, **Y**$_3$, and **Y**$_4$ may vary.

Med-Med [2nd] [STAT] [◀] 4

Med-Med (Median-Median) fits the model equation, **y=ax+b**, to the data using the median-median line (resistant line) technique, calculating the summary points **x1**, **y1**, **x2**, **y2**, **x3**, and **y3**. **Med-Med** displays values for **a** (slope) and **b** (y-intercept). You can execute **Med-Med** from either the Graph screen, the Home screen, or the Program editor.

From the Home screen or the Program editor, **Med-Med** accepts four optional arguments. Enter up to two list names, *XList* and *YList*; a frequency list, *freq*; and an equation variable, *Yn*. *freq* is the frequency of occurrence for each corresponding data point in *XList* and *YList*.

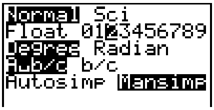
If *freq* is omitted, all values are used once. If *XList* and *YList* are not specified, the default list names are **L1** and **L2**. To access **Y_n** variables, press [2nd] [VARS] **2:Y-Vars**.

Med-Med [*XList*,*YList*,*freq*,*Yn*]

Graph a scatter plot for **L1** and **L2**, where **L1={1,3,4,5,5,7,8,9}** and **L2={1,4,2,3,4,6,7,9}**, and use **Med-Med** to draw the median-median line through the points.

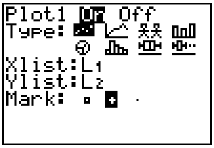
1. Set Decimal Notation mode to **2**, if desired.

   [MODE] [▼] [▶] [▶] [▶] [ENTER]



2. After entering the lists, define **Plot1** as a scatter plot using **L1** and **L2**, as shown to the right.

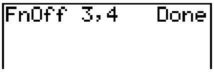
   [2nd] [PLOT]



For more information on defining stat plots, see Chapter 6: Statistical Plots.

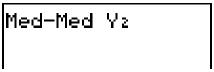
3. Turn off **Y_3** and **Y_4**, if they have been previously defined and selected.

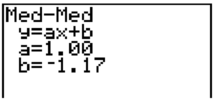
   [2nd] [QUIT] [CLEAR]
   [2nd] [VARS] **2** **6**
   3 [,] **4** [ENTER]



4. Find the **Med-Med** line, and store the results to **Y_2**.

   [CLEAR] [2nd] [STAT] [◀] **4**
   [2nd] [VARS] **2** **2** [ENTER]

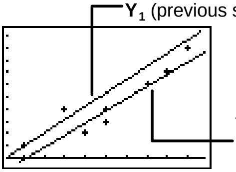




Specifying **L1** and **L2** is optional since they are the default *listnames*. However, if you were using other list names, you would have to enter them before the *Yn* variable.

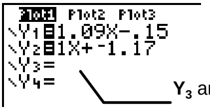
5. View the line on the Graph screen.

   [ZOOM] **7**



6. View the equation stored to **Y2**, if desired.

   [Y=]



LinReg(ax+b) [2nd] [STAT] [◄] 5

LinReg(ax+b) (linear regression) fits the model equation **y=ax+b** to the data using a least-squares fit. It displays the value for **a** (slope) and **b** (y-intercept); when **DiagnosticOn** is set, it also displays values for **r²** (coefficient of determination) and **r** (correlation coefficient). The **DiagnosticOn** command is in the **CATALOG** ([2nd][CATALOG]).You can execute **LinReg(ax+b)** from the Graph screen, Home screen, or the Program editor.

It is also helpful to compare the slope of the line you draw with **Manual-Fit** to the slope of the line the calculator calculates with the **LinReg(ax+b)** command.

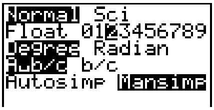
From the Home screen or the Program editor, **LinReg(ax+b)** accepts four optional arguments. Enter up to two list names, *XList* and *YList*; a frequency list, *freq*; and an equation variable, *Yn*. *freq* is the frequency of occurrence for each corresponding data point in *XList* and *YList*. If *freq* is omitted, all values are used once. If *XList* and *YList* are not specified, the default list names are **L1** and **L2**. To access **Yn** variables, press [2nd] [VARS] **2:Y-Vars**.

LinReg(ax+b) [*XList*,*YList*,*freq*,*Yn*]

Graph a scatter plot for **L1** and **L2**, where **L1={1,3,4,5,5,7,8,9}** and **L2={1,4,2,3,4,6,7,9}**, and use **LinReg(ax+b)** to draw the linear regression line through the points.

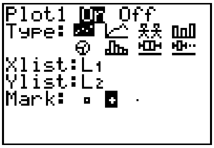
1. Set Decimal Notation mode to **2**, if desired.

   [MODE] [▼] [▶] [▶] [▶] [ENTER]



2. After entering the lists, define **Plot1** as a scatter plot using **L1** and **L2**, as shown to the right.

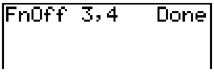
   [2nd] [PLOT]



For more information on defining stat plots, see Chapter 6: Statistical Plots.

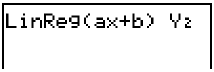
3. Turn off $\mathbf{Y_3}$ and $\mathbf{Y_4}$, if they have been previously defined and selected.

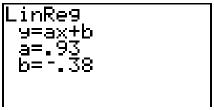
   [2nd] [QUIT] [CLEAR]
   [2nd] [VARS] **2** **6**
   3 [,] **4** [ENTER]



4. Find the **LinReg(ax+b)** line, and store the results to $\mathbf{Y_2}$.

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] [◀] **5**
   [2nd] [VARS] **2** **2** [ENTER]

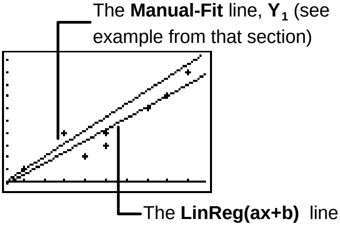




Specifying **L1** and **L2** is optional since they are the default *listnames*. However, if you were using other list names, you would have to enter them before the *Yn* variable.

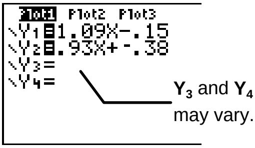
5. View the line on the Graph screen.

   [ZOOM] **7**



6. View the equation stored to Y_2, if desired.

   [Y=]



QuadReg [2nd] [STAT] [◄] 6

QuadReg (quadratic regression) fits the second-degree polynomial **y=ax²+bx+c** to the data. It displays values for **a**, **b**, and **c**; when **DiagnosticOn** is set, it also displays a value for **r²** (coefficient of determination). The **DiagnosticOn** command is in the **CATALOG** ([2nd] [CATALOG]). You can execute the **QuadReg** command from the Graph screen, the Home screen, or the Program editor.

For three data points, the equation is a polynomial fit; for four or more, it is a polynomial regression. At least three data points are required.

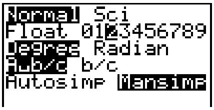
From the Home screen or the Program editor, **QuadReg** accepts four optional arguments. Enter up to two list names, *XList* and *YList*; a frequency list, *freq*; and an equation variable, *Yn*. *freq* is the frequency of occurrence for each corresponding data point in *XList* and *YList*. If *freq* is omitted, all values are used once. If *XList* and *YList* are not specified, the default list names are **L1** and **L2**. To access Y_n variables, press [2nd] [VARS] **2**.

QuadReg [*XList*,*YList*,*freq*,*Yn*]

Graph a scatter plot for L_1 and L_2, where **L_1={1,3,4,5,5,7,8,9}** and **L_2={1,4,2,3,4,6,7,9}**, and use **QuadReg** to draw the quadratic regression curve through the points.

1. Set Decimal Notation mode to **2**, if desired.

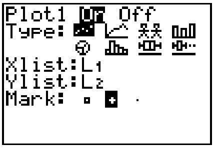
   [MODE] [▼] [▶] [▶] [▶] [ENTER]



2. After entering the lists, define **Plot1** as a scatter plot using L_1 and L_2, as shown to the right.

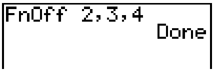
   [2nd] [PLOT]

   For more information on defining Stat plots, see Chapter 6: Statistical Plots.



3. Turn off Y_2, Y_3 and Y_4, if they have been previously defined and selected.

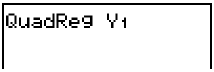
   [2nd] [QUIT] [CLEAR]
   [2nd] [VARS] **2** **6**
   2 [,] **3** [,] **4** [ENTER]

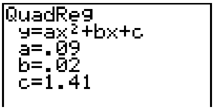


4. Find the **QuadReg** curve, and store the results to Y_1.

   [2nd] [QUIT] [CLEAR]
   [2nd] [STAT] [◀] **6**
   [2nd] [VARS] **2** **1** [ENTER]

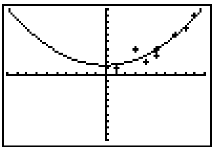
   Specifying **L1** and **L2** is optional since they are the default *listnames*. However, if you were using other list names, you would have to enter them before the *Yn* variable.





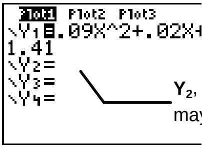
5. View the curve on the Graph screen.

   [ZOOM] **6**



6. View the equation stored to **Y**$_1$, if desired.

   [Y=]



Y$_2$, **Y**$_3$, and **Y**$_4$ may vary.

ExpReg [2nd] [STAT] [◄] 7

ExpReg (exponential regression) fits the model equation **y=ab^x** to the data using a least-squares fit and transformed values x and ln(y). It displays values for **a** and **b**; when **DiagnosticOn** is set, it also displays values for **r²** (coefficient of determination) and **r** (correlation coefficient). The **DiagnosticOn** command is in the **CATALOG** ([2nd] [CATALOG]). You can execute **ExpReg** from the Graph screen, the Home screen, or the Program editor.

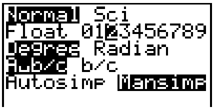
From the Home screen or the Program editor, **ExpReg** accepts four optional arguments. Enter up to two list names, *XList* and *YList*; a frequency list, *freq*: and an equation variable, *Yn*. *freq* is the frequency of occurrence for each corresponding data point in *XList* and *YList*. If *freq* is omitted, all values are used once. If *XList* and *YList* are not specified, the default list names are **L1** and **L2**. To access **Y**$_n$ variables, press [2nd] [VARS] **2**.

ExpReg [*XList*,*YList*,*freq*,*Yn*]

Graph a scatter plot for **L1** and **L2**, where **L1={1,3,4,5,5,7,8,9}** and **L2={1,4,2,3,4,6,7,9}**, and use **ExpReg** to draw the exponential regression curve through the points.

1. Set Decimal Notation mode to **2**, if desired.

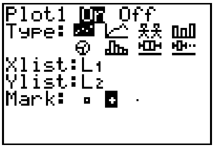
   [MODE] [▼] [►] [►] [►] [ENTER]



2. After entering the lists, define **Plot1** as a scatter plot using **L1** and **L2**, as shown to the right.

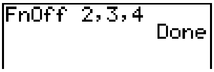
   [2nd] [PLOT]

   For more information on defining stat zplots, see Chapter 6: Statistical Plots.



3. Turn off $\mathbf{Y_2}$, $\mathbf{Y_3}$, and $\mathbf{Y_4}$, if they have been previously defined and selected.

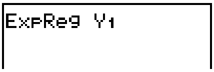
   [2nd] [QUIT] [CLEAR]
   [2nd] [VARS] **2** **6**
   2 [,] **3** [,] **4** [ENTER]

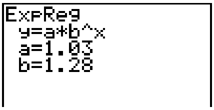


4. Find the **ExpReg** curve, and store the results to $\mathbf{Y_1}$.

   [2nd] [QUIT] [CLEAR] [2nd]
   [STAT] [◄] **7** [2nd] [VARS] **2**
   1 [ENTER]

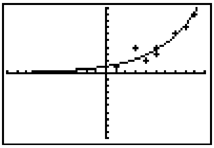
   Specifying **L1** and **L2** is optional since they are the default *list names*. However, if you were using other list names, you would have to enter them before the *Yn* variable.





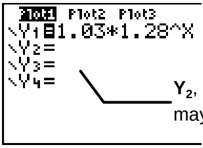
5. View the curve on the Graph screen.

   [ZOOM] **6**



6. View the equation stored to $\mathbf{Y_1}$, if desired.

   [Y=]



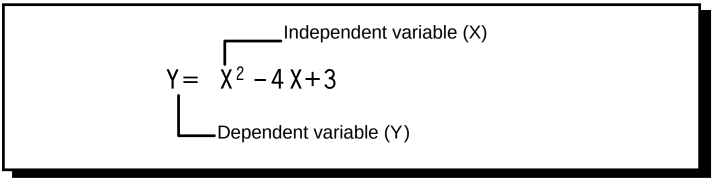
$\mathbf{Y_2}$, $\mathbf{Y_3}$, and $\mathbf{Y_4}$ may vary.

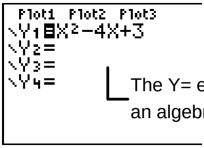
8 Tables

What Is a Table?

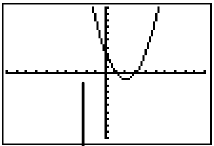
A table displays coordinate pair (**X**,**Y**) solutions for a defined function. One column displays independent variable values (**X**), and all others display corresponding dependent variable values (**Y**).

On the TI−73, functions can be displayed in one of three ways, as shown here with the function, $\mathbf{Y_1=X^2\text{-}4X+3}$.

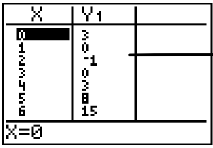




The Y= editor ([Y=]) shows an algebraic representation.



The Graph screen ([GRAPH]) shows a graphic representation.



The Table screen ([2nd] [TABLE]) shows a numeric representation in coordinate pairs.

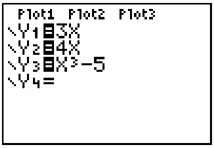
For more information about the Y= editor and function graphing, see Chapter 9: Function Graphing.

Steps for Creating a Table

Follow these basic steps when defining a table.

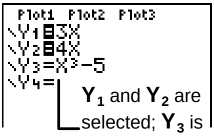
Define or edit up to four functions in the Y= editor.

[Y=]



Select the $\mathbf{Y}_n$ function(s) that you want to be included in the table.

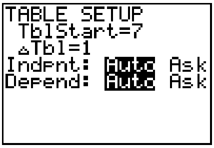
Highlight =, and press [ENTER].



$\mathbf{Y}_1$ and $\mathbf{Y}_2$ are selected; $\mathbf{Y}_3$ is deselected.

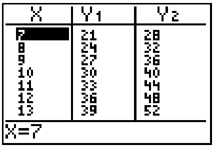
Set up the table.

[2nd] [TBLSET]



Display the table. Numeric Notation, Decimal Notation, and Angle mode settings determine the display of the elements.

[2nd] [TABLE]



Defining and Selecting Functions in the Y= Editor [Y=]

To create a table of values for a function, you first must define the function in the Y= editor. Press [Y=] to display the Y= editor; then define up to four functions, Y_1, Y_2, Y_3, and Y_4, in terms of the independent variable, **X**.

For every *selected* function in the Y= editor, the calculator automatically creates a column of Y_n values. Because the Y= editor holds up to four functions, the TI-73 can create up to four Y_n columns in a table, one for each function.

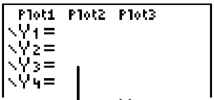
When you first enter a function, it is selected automatically. To select or deselect a function, highlight the **=** with the cursor, and then press [ENTER].

For more details on entering functions, see Chapter 9: Function Graphing.

Define $Y_1=X^2-4X+3$.

1. Display the Y= editor.

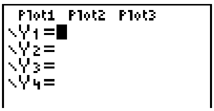
   [Y=]



Your screen may vary.

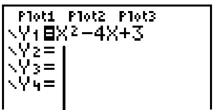
2. Move the cursor to Y_1, and clear if necessary.

   [▼] [CLEAR]



3. Enter **Y_1= X^2 -4X+3**.

   [x] [x²] [−] **4** [x] [+] **3**

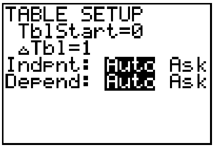


When you first enter a function, it is selected automatically.

Setting Up the Table [2nd] [TBLSET]

Use the **TABLE SETUP** screen to specify the initial settings for your table. To select an **Indpnt** or **Depend** setting, highlight the one you want with the cursor, and then press [ENTER].

[2nd] [TBLSET]



TblStart
Default=**0**

Specifies the first value displayed in the independent variable (**X**) column and can be any real number.

ΔTbl
Default=**1**

Specifies the increment by which the **X** values increase or decrease.

Indpnt:
Default=**Auto**

Refers to the *independent* variable (**X**) column values. You must select one of two choices:

- **Auto** — **X** values are automatically displayed in the independent variable column when you view the Table screen.
- **Ask** — No **X** values are shown when you view the Table screen. Instead, you enter the values for the **X** column.

Depend:
Default= **Auto**

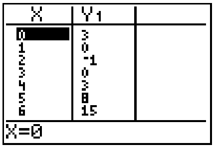
Refers to all *dependent* variable (Y_n) column values. You must select one of two choices:

- **Auto** — Y_n values of all selected functions are automatically displayed in their respective columns when you view the Table screen.
- **Ask** — No Y_n values are shown when you view the Table screen. Instead you select which Y_n values you want the calculator to display.

Displaying the Table [2nd] [TABLE]

Once your functions are defined and selected in the Y= editor and you have set up your table in the **TABLE SETUP** screen, if necessary, you can display the table with [2nd] [TABLE].

[2nd] [TABLE]



On the Table screen, you can see lower **X** values by placing the cursor anywhere in the **X** column and pressing [▲], as necessary (you can't scroll up from the $\mathbf{Y_n}$ columns). To see higher **X** values, use [▼] from anywhere on the Table screen.

Only two $\mathbf{Y_n}$ columns appear at a time on the Table screen. Use [▶] to display a third or fourth $\mathbf{Y_n}$ column.

When you highlight a table element, the entry line displays the value in its entirety.

The values displayed in the table are affected by the mode settings. If the calculator is set to the **Sci** Numeric Notation mode, all applicable values in all columns are displayed in scientific notation. If your calculator is set to **Radian** Angle mode and a defined function is a trig function, all the table values for that function are interpreted as radians, not degrees.

Indpnt=Auto and Depend=Auto

Select these settings on the **TABLE SETUP** screen when you want all **X** and $\mathbf{Y_n}$ values to appear automatically.

You have two dogs, Rover and Spot. You feed Rover 3 times a day. You feed Spot 4 times a day. How many times will Spot and Rover have eaten after 3 and 5 days?

$Y_1=3X$	X=number of days Y=total times Rover has eaten
$Y_2=4X$	X=number of days Y=total times Spot has eaten

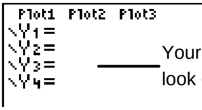
1. Reset default settings.

   [2nd] [MEM] **7 2 2**

   Note: This resets table settings and all mode settings, and deselects any previously defined and selected $\mathbf{Y_n}$ functions.

2. Display the Y= editor.

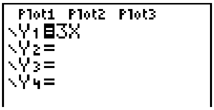
   [Y=]



Your Y= editor may look different.

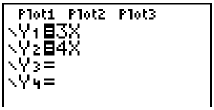
3. Clear $\mathbf{Y_1}$, if necessary. Enter $\mathbf{Y_1=3X}$.

   [CLEAR] **3** [x]



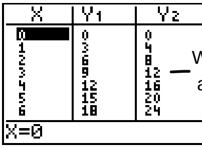
4. Clear $\mathbf{Y_2}$, if necessary. Enter $\mathbf{Y_2=4X}$.

   [▼] [CLEAR] **4** [x]



5. Display the table (using default table settings).

   [2nd] [TABLE]



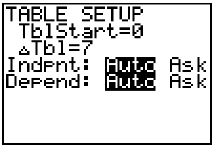
When **X=3**, $\mathbf{Y_1=9}$ and $\mathbf{Y_2=12}$.

After Day 3	Rover has eaten 9 times. Spot has eaten 12 times.
After Day 5	Rover has eaten 15 times. Spot has eaten 20 times.

How many times will Spot and Rover have eaten after 1, 3, and 4 weeks? (Refer to the previous example, if necessary.)

1. Set up the table where **TblStart=0**, **ΔTbl=7**, **Indpnt=Auto**, and **Depend=Auto**.

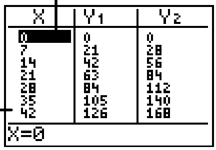
   [2nd] [TBLSET]
   0 [▼] **7** [▼] [ENTER]
   [▼] [ENTER]



2. Display the table.

   [2nd] [TABLE]

TblStart=0



X values change by 7 since **ΔTbl=7.**

After Day 7 (End of Week 1)	Rover has eaten 21 times. Spot has eaten 28 times.
After Day 21 (End of Week 3)	Rover has eaten 63 times. Spot has eaten 84 times.
After Day 28 (End of Week 4)	Rover has eaten 84 times. Spot has eaten 112 times.

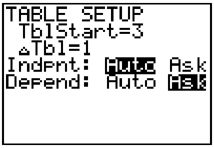
Indpnt=Auto and Depend=Ask

Select these settings on the **TABLE SETUP** screen when you want **X** values to appear automatically, but you want to be able to reveal $\mathbf{Y_n}$ values one at a time. It is also helpful in recognizing patterns between different $\mathbf{Y_n}$ solutions.

Display the number of times Rover has eaten after 4 days and 8 days, and display the number of times Spot has eaten after 3 days and 6 days. (Refer to the previous example, if necessary.)

1. Setup the table where **TblStart=3**, **ΔTbl=1**, **Indpnt=Auto**, and **Depend=Ask**.

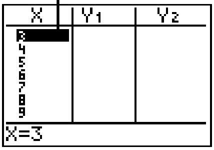
   [2nd] [TBLSET]
   3 [▼] **1** [▼] [ENTER] [▼] [▶]
   [ENTER]



2. Display the table.

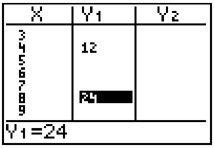
   [2nd] [TABLE]

X starts with 3 because **TblStart=3**.



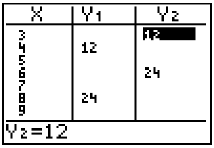
3. Display how many times Rover (**Y$_1$**) has eaten after 4 and 8 days.

   [▼] [▶] [ENTER]
   [▼] [▼] [▼] [▼] [ENTER]



4. Display how many times Spot (**Y$_2$**) has eaten after 3 and 6 days.

   [▶] [▲] [▲] [ENTER]
   [▲] [▲] [▲] [ENTER]



After Day 3	Spot has eaten 12 times.
After Day 4	Rover has eaten 12 times.
After Day 6	Spot has eaten 24 times.
After Day 8	Rover has eaten 24 times.

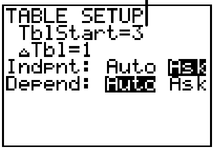
Indpnt=Ask

Select these settings on the **TABLE SETUP** screen when you want to find specific table values, especially those that are not in chronological order or which span across a large range of numbers. **TblStart** and **ΔTbl** do not apply when **Indpnt=Ask**.

How many total times will Spot and Rover have eaten after 16 days, 37 days, 52 days, and 74 days? (Refer to the previous examples, if necessary.)

1. Setup the table where **Indpnt=Ask** and **Depend=Auto**.

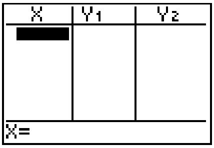
   [2nd] [TBLSET]
   [▼] [▼] [▶] [ENTER]
   [▼] [ENTER]



TblStart and **ΔTbl** values are ignored.

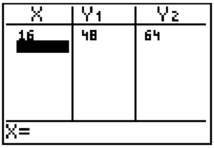
2. Display the table.

   [2nd] [TABLE]



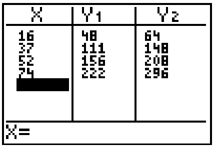
3. Enter **X=16**.

   1 6 [ENTER]



4. Enter **X=37**, **X=52**, and **X=74**.

   3 7 [ENTER] **5 2**
   [ENTER] **7 4** [ENTER]



After Day 16	Rover has eaten 48 times. Spot has eaten 64 times.
After Day 37	Rover has eaten 111 times. Spot has eaten 148 times.
After Day 52	Rover has eaten 156 times. Spot has eaten 208 times.
After Day 74	Rover has eaten 222 times. Spot has eaten 296 times.

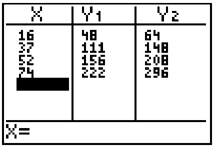
Editing X Values from the Table Screen

You can edit **X** values from the Table screen when **Indpnt=Ask**.

Change X=37 to X=36. (Refer to the previous example, if necessary.)

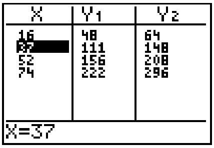
1. Display the current table.

   [2nd] [TABLE]



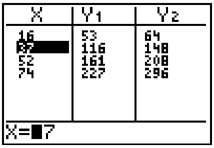
2. Highlight **X=37**.

   [▼] or [▲] (as necessary)



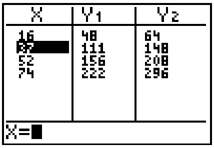
3. Move the cursor to the entry line.

   [ENTER]



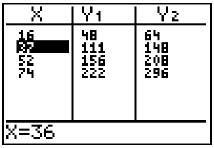
4. Clear the entry line.

   [CLEAR]



5. Enter **36** and insert it into table.

   3 6 [ENTER]



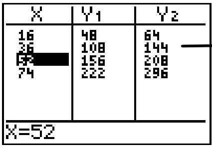


Table values are adjusted.

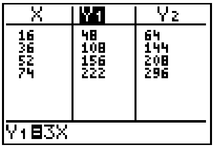
Editing Y_n from the Table Screen

At any time you can edit $\mathbf{Y_n}$ from the Table screen without returning to the Y= editor.

Change Y_1 =3x to Y_1 =3x+5. (Refer to the previous example, if necessary.)

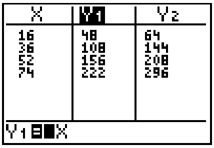
1. Display the Table screen, and highlight $\mathbf{Y_1}$ with the cursor.

   [2nd] [TABLE]
   [▶] a nd [▲] (as necessary)



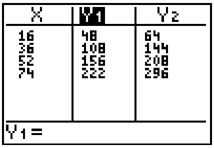
2. Move the cursor to the entry line.

   [ENTER]



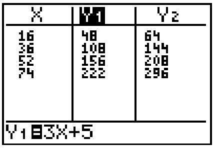
3. Clear the entry line.

   [CLEAR]



4. Enter **3X+5**.

   3 [x] [+] **5**



5. Insert the equation back into the table.

   [ENTER]

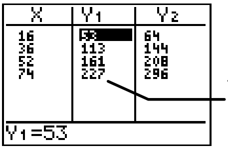


Table values are adjusted.

6. If desired, display the Y= editor to confirm that $\mathbf{Y_1}$ has indeed been changed.

   [Y=]

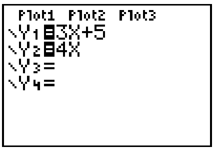


Table Setup from the Home Screen

You can store values to **TblStart** and **ΔTbl** from the Home screen or the Program editor. These table variable names are on the [2nd] [VARS] **5:Table** menu.

You also can select **DependAsk**, **DependAuto**, **IndpntAsk**, and **IndpntAuto** from a Program editor to turn on these settings during program execution.

Assign 6 to **TblStart** and 3 to **ΔTbl** from the Home screen.

1. Go to Home screen and clear, if desired.

   [2nd] [QUIT] [CLEAR]

2. Store 6 to **TblStart**.

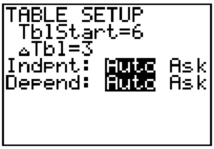
   6 [STO▸] [2nd] [VARS] **5**
   1 [ENTER]

3. Assign 3 to **ΔTbl**.

   3 [STO▸] [2nd] [VARS] **5**
   2 [ENTER]

4. Display the **TABLE SETUP** screen to confirm that the values you entered have indeed been set.

   [2nd] [TBLSET]



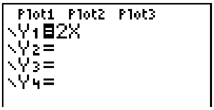
9 Function Graphing

Steps for Graphing a Function

Follow these basic steps when graphing a function. You may not have to do all of them each time.

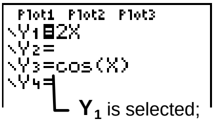
Define or edit up to four functions in the Y= editor.

[Y=]



Select the Y_n function(s) that you want to graph. Deselect statistical plots, if desired (Chapter 7).

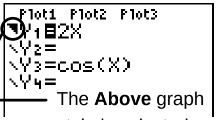
Highlight **=**, and then press [ENTER] to select or deselect.



$\mathbf{Y_1}$ is selected; $\mathbf{Y_3}$ is deselected.

Set the graph style for each selected function.

Highlight the area to the left of $\mathbf{Y_n}$. [ENTER] cycles through 7 styles.



The **Above** graph style is selected for $\mathbf{Y_1}$

Set the window format.

[2nd] [FORMAT]

Define the viewing Window values. This may include using the **ZOOM** menu.

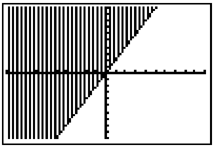
[WINDOW]
or
[ZOOM]

[ZOOM] **6** values

Xmin=⁻10
Xmax=10
Δx=.21276595744681
Xscl=1
Ymin=⁻10
Ymax=10
Yscl=1

Graph the selected functions with [GRAPH]. [TRACE] and [ZOOM] also automatically display the graph.

[GRAPH],
[TRACE],
or
[ZOOM]



Example of Function Graphing

For every cookie Tham eats, Antonio eats two. How many cookies does Antonio eat if Tham eats 1 cookie, 2 cookies, 3 cookies, and 4 cookies?

Find the equation that represents the relationship between how many cookies Tham eats and how many Antonio eats, and represent your answers in the form of a function graph.

Y=2X	X=number of cookies Tham eats Y=number of cookies Antonio eats

These steps explain what the calculator does internally when you define a function graph. The next page shows how to use the TI−73 to find the answers to this example.

1. This example uses these *X* values:

   X=1
   X=2
   X=3
   X=4

2. The TI−73 solves for *Y* using specific *X* values.

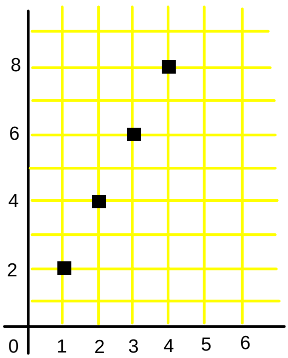
   Y = 2∗X

   Y = 2∗1 = 2
   Y = 2∗2 = 4
   Y = 2∗3 = 6
   Y = 2∗4 = 8

3. It generates a table of (*X*, *Y*) coordinate pairs for you to look at.

X	Y
1	2
2	4
3	6
4	8

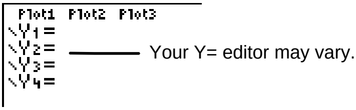
4. It graphs the (*X*,*Y*) pairs.



Graph Y=2X on your calculator and find the solutions to the word problem.

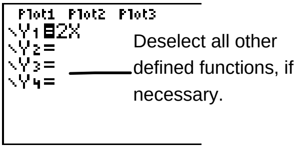
1. Display the Y= editor.

   [Y=]



2. Clear **Y$_1$=,** if necessary. Enter **Y$_1$=2X**.

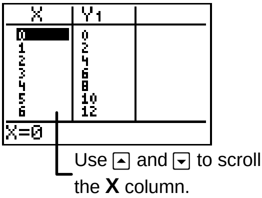
   [CLEAR] **2** [x]



3. Show the table of (X,Y) coordinate pairs, if desired; use **TblStart=0** and **ΔTbl=1**.

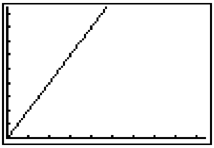
   [2nd] [TABLE]

   See Chapter 8: Tables for more information about function tables.



4. Define the viewing window for Quadrant 1 only.

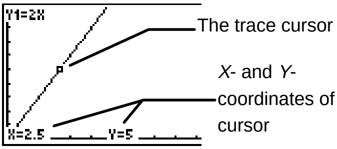
   [ZOOM] **4**



5. Trace the graph with the cursor keys.

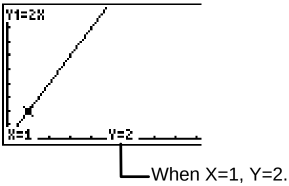
   [TRACE]

   (Use [◄] and [►] to move the cursor along the graph.)



6. Find the Y values when X=1, 2, 3,and 4.

   1 [ENTER]
   2 [ENTER]
   3 [ENTER]
   4 [ENTER]



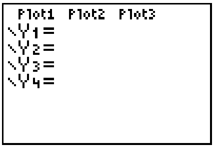
Defining Functions in the Y= Editor [Y=]

Use the Y= editor to define up to four functions, $\mathbf{Y_1}$, $\mathbf{Y_2}$, $\mathbf{Y_3}$, and $\mathbf{Y_4}$, in terms of the independent variable, *X*.

Press [Y=] to display the Y= editor. The TI−73 graphs up to four defined functions at the same time.

If the result of an expression is not a real number, that point is not plotted. You do not get an error.

[Y=]



Entering Functions

Functions can consist of variables, lists, trigonometric or logarithmic expressions, or variations of already defined functions (for example, $\mathbf{Y_2=2*Y_1}$). Access a $\mathbf{Y_n}$ variable by pressing [2nd] [VARS] **2:Y-Vars**.

Define Y_2=3X+5.

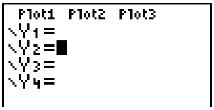
1. Display the Y= editor.

   [Y=]

If you tried the chapter example problem, then $\mathbf{Y_1=2X}$.

2. Move the cursor to the function that you want to define, $\mathbf{Y_2}$.

   [▼]

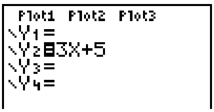


3. Clear $\mathbf{Y_2}$, if necessary.

   [CLEAR]

4. Enter $\mathbf{Y_2=3X+5}$.

   3 [*x*] [+] **5**



Editing Functions

You can edit or delete functions at any time in the Y= editor. Move the cursor to the function in the Y= editor that you want to change.

You can:

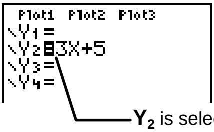
- Use the edit keys such as [DEL] and [2nd] [INS] to delete and insert characters.
- Overwrite current entries.
- Delete a function with [CLEAR]. Position the cursor anywhere on the function.

Selecting Functions

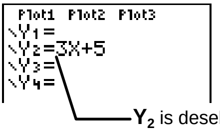
Even if a function is defined in the Y= editor, the TI−73 only graphs the function if it is selected (turned on). You know that a function is selected because the background behind a function's equal sign (=) is dark.

When you first define a function, it is selected automatically.

To select or deselect a function, highlight its = using the cursor keys, and then press [ENTER].



Y_2 is selected.

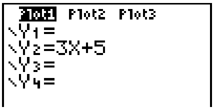


Y_2 is deselected.

You can change the on/off status of a statistical plot in the Y= editor. To select or deselect **Plot1**, **Plot2**, or **Plot3**, highlight the name (across the top of the Y= editor) using the cursor keys, and then press [ENTER]. A plot is selected (on) if the background behind its name is dark.

See Chapter 6: Statistical Plots for more information on defining and graphing stat plots.

Plot1 is selected. **Plot2** and **Plot3** are deselected.



Exiting the Y= Editor

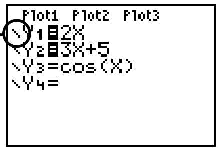
To select another screen, press the appropriate key, such as [GRAPH] or [WINDOW]. Press [2nd] [QUIT] to return to the Home screen.

Selecting a Graph Style

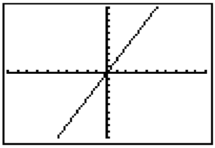
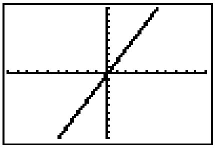
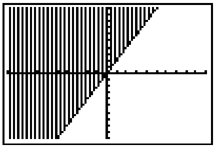
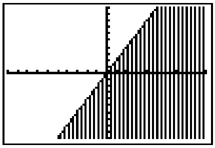
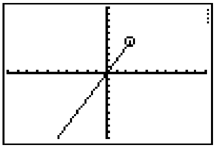
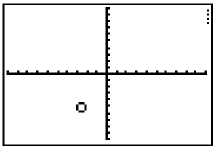
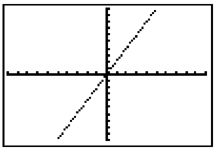
For a defined function, you can set one of seven styles that specify the appearance of a function graph. The graph style icons described below are located to the left of Y_n in the Y= editor. If you do not select a style, the calculator graphs all defined functions with the default style, **Line**.

To select a style, press [◄] from the Y_n equal sign (**=**) to highlight the graph style icon, and then press [ENTER], as necessary, to cycle through the seven styles. Press [►] [►] to return to the Y_n entry line.

The graph style icon.



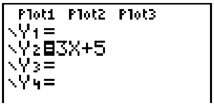
Graph styles are especially useful when graphing multiple functions. For example, you can set Y_1 as a solid line, Y_2 as a dotted line, and Y_3 as a thick line.

Icon	Style	Description	Example (Y_1=2x)
	Line	Connects plotted points with a line. This is the default.	
	Thick	Connects plotted points with a thick line.	
	Above	Shades the area above the graph.	
	Below	Shades the area below the graph.	
	Path	A circular cursor traces the graph and draws the path.	
	Animate	A circular cursor traces the graph without drawing the path.	
	Dot	Displays a dot at each plotted point.	

Set the **Below** graph style for Y_2=3X+5.

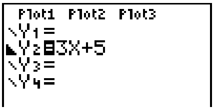
1. Enter the Y= editor and define **Y_2=3X+5**.

   [Y=] [▼] [CLEAR] **3** [x] [+] **5**



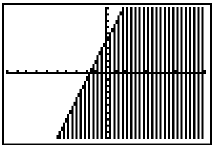
2. Highlight the graph style icon (to the left of the **Y_2**) and select the graph style, **Below**.

   [◀] [◀] [◀] [◀] [◀] [◀]
   [ENTER] [ENTER] [ENTER]



3. Display the graph.

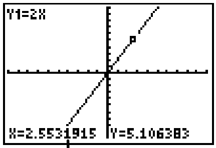
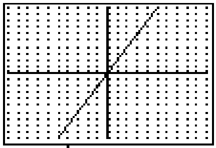
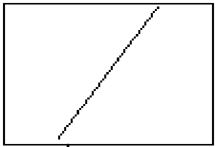
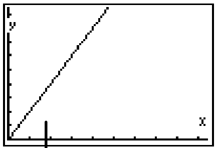
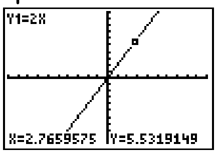
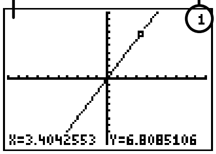
   [ZOOM] **6**



Setting the Window Format
[2nd] [FORMAT]

The window format screen lets you choose display settings. These apply to function graphing *and* statistical plotting.

[2nd] [FORMAT]

Setting	Turns these on or off:	Example:
CoordOn/ CoordOff	*X*- and *Y*-coordinates of the cursor at the bottom of the screen. Useful when tracing a graph.	 **CoordOn**
GridOff/ GridOn	Grid lines that correspond to the axes tick marks.	 **GridOn**
AxesOn/ AxesOff	*X*- and *Y*-axes.	 **AxesOff**
LabelOff/ LabelOn	Labels for the *X*- and *Y*-axes. These settings are disregarded when **AxesOff** is selected. **LabelOn** is especially helpful when displaying Quadrant I (ZOOM **4**) graphs.	 A Quadrant I graph with **LabelOn** selected
ExprOn/ ExprOff	Expression which is currently being traced. The expression is shown in the top left corner of a graph.	**ExprOn** 
	When **CoordOn** and **ExprOff** are both selected, the number in the top-right corner specifies which function is being traced.	Y_1 is being traced. **ExprOff** 

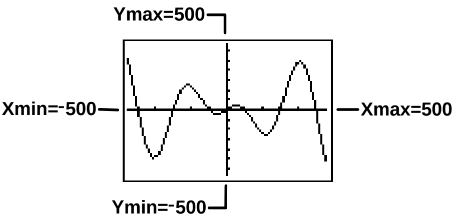
Defining Window Values

If you enter a function in the Y= editor and press [GRAPH], but nothing happens or the graph doesn't look the way you expect it to, you may need to adjust the **WINDOW** values ([WINDOW]).

Depending upon which section of a graph you specify through the **WINDOW** values, the display on your calculator screen can look very different.

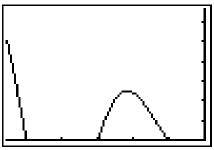
In the example below, the first calculator screen uses **WINDOW** values which include all four quadrants for the function, **Y_1=X∗cos(X)**. (Calculator is in **Degree** mode.) Then, Quadrants I, II, III, and IV are shown separately, so you can see how **WINDOW** values affect the display. The next section explains how to redefine the values.

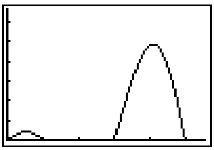
Y_1=X∗cos(X)



All Quadrants
Xmin=⁻500
Xmax=500
Xscl=90
Ymin=⁻500
Ymax=500
Yscl=75

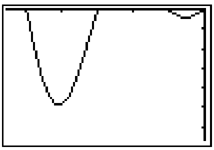
Quadrant II
Xmin=⁻500
Xmax=0
Xscl=90
Ymin=0
Ymax=500
Yscl=75

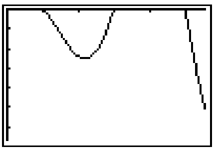




Quadrant I
Xmin=0
Xmax=500
Xscl=90
Ymin=0
Ymax=500
Yscl=75

Quadrant III
Xmin=⁻500
Xmax=0
Xscl=90
Ymin=⁻500
Ymax=0
Yscl=75



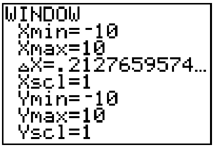


Quadrant IV
Xmin=0
Xmax=500
Xscl=90
Ymin=⁻500
Ymax=0
Yscl=75

The Window Values Screen [WINDOW]

WINDOW values put specific boundaries on the display. For an explanation of **ΔX**, see the section in this chapter entitled “Controlling the Increments of a Trace.”

To exit the **WINDOW** menu, select another screen by pressing the appropriate key, or press [2nd] [QUIT] to return to the Home screen.

[WINDOW]	

Xmin	The minimum value on the *X*-axis; must be less than **Xmax**.
Xmax	The maximum value on the *X*-axis.
Δx	When tracing the graph with [TRACE], this determines the increments between *X* values.
Xscl	The distance between tick marks on the *X*-axis. To turn off the tick marks, set **Xscl=0**.
Ymin	The minimum value on the *Y*-axis; must be less than **Ymax**.
Ymax	The maximum value on the *Y*-axis.
Yscl	The distance between tick marks on the *Y*-axis. To turn off the tick marks, set **Yscl=0**.

Determining Window Values for a Specific Graph

The following example shows how you can adjust the **WINDOW** values manually (as opposed to using the standard **WINDOW** values set by ZOOM **6:ZStandard**).

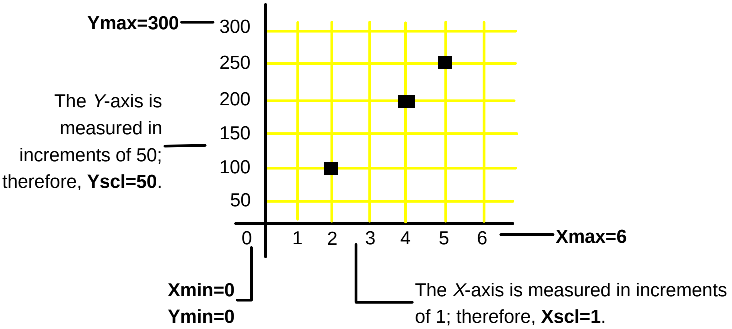
Yuko practices the piano 50 minutes per day. How many minutes has he practiced after 2, 4, and 5 days? Graph your answer.

`Y=50X`	`X=number of days` `Y=number of total minutes`

1. A table of coordinate pairs would look like this:

X	Y
2	100
4	200
5	250

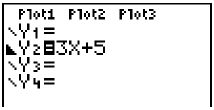
2. A possible graph of the ordered pairs would look like this (the **WINDOW** values are labeled):



Graph the function, Y_1=50X, on your calculator.

1. Display the Y= editor.

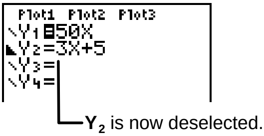
   [Y=]



2. Enter **Y_1=50X.**

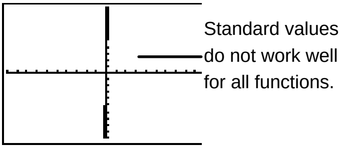
   [CLEAR] **5 0** [x]

   Note: Deselect any other functions by highlighting the corresponding = and pressing [ENTER].



3. Graph the function using standard window values (**ZStandard**).

   [ZOOM] **6**

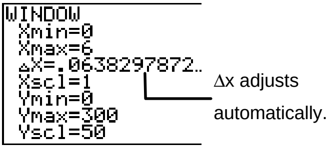


4. Adjust the **WINDOW** values to match the sample graph from the previous page.

   [WINDOW]

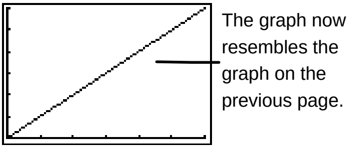
   0 [▼] **6** [▼] [▼] **1** [▼] **0** [▼]

   3 0 0 [▼] **5 0**



5. Graph **Y_1.**

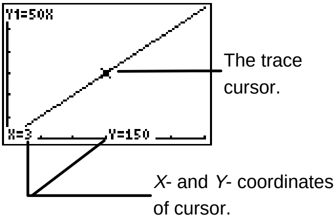
   [GRAPH]



6. Trace the graph.

   [TRACE]

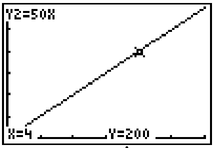
   (Use [◄] and [►] to move the cursor along the graph.)



7. Find the Y values when X= 2, 4, and 5.

   2 [ENTER] **4** [ENTER] **5** [ENTER]

   Note: Consider using the **CONVERSIONS** menu ([2nd] [CONVERT] **4**) to convert your answers (in minutes) to seconds, hours, days, weeks, or years.



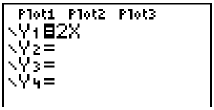
When **X=4**, **Y=200**.

If you trace ([TRACE]) the graph with the cursor keys to an X value greater than **Xmax** or less than **Xmin**, the cursor goes off the Graph screen, but the corresponding Y values are still displayed since they exist. However, you cannot enter X values (as you did in step 7 above) that are greater than **Xmax** or less than **Xmin**.

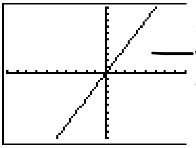
Displaying a Graph [GRAPH]

Press [GRAPH] to display the graph of the selected function(s). (Some operations, such as [TRACE] and [ZOOM], display the graph automatically.) As a graph is plotted, the busy indicator comes on (upper right corner) until the graph is completely drawn and X and Y are updated.

Pressing [GRAPH], [TRACE], or selecting a [ZOOM] function graphs all defined and selected functions.



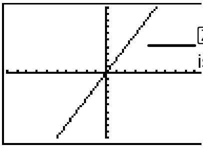
- If the desired **WINDOW** values are already set, press [GRAPH] or [TRACE].



Standard default window values are set.

- Press [ZOOM] to change the **WINDOW** values and graph all selected functions.

  [ZOOM] , then select a function from the menu



[ZOOM] **6:ZStandard** is selected.

To pause while a graph is being drawn, press [ENTER]; press [ENTER] again to resume plotting.

Press [ON] to stop graphing. Press [GRAPH] to start over and plot again.

Smart Graph

When you press [GRAPH], the Graph screen immediately displays (instead of replotting) the previous function graph(s) if no changes were made. If changes were made, the functions are replotted.

The graph is replotted if you have:

- Changed a function.
- Selected or deselected a function.
- Changed the value of a variable in a selected function.
- Changed a **WINDOW** variable or a [2nd] [FORMAT] setting.
- Cleared drawings by selecting **ClrDraw** (Chapter 10: Draw).
- Changed a stat plot definition (Chapter 6: Statistical Plots).

Exploring the Graph with the Free-Moving Cursor

Use [▶], [◀], [▲], and [▼] to move the cursor around the Graph screen. When you first display the graph, the cursor is in the middle of the screen but is not visible. When you press a cursor key, the cursor moves from that point and can be seen. (Remember to use the [2nd] [FORMAT] **CoordOn** setting if you want to see the (X,Y) coordinates at the bottom of the screen.)

Exploring a Function Graph with [TRACE]

Pressing [TRACE] allows you to move the [◀] and [▶] cursor keys from one plotted point to another and displays the cursor coordinates at the bottom of the screen (if **CoordOn** is set). If **ExprOn** ([2nd] [FORMAT]) is set, the expression being traced appears in the top left corner.

When more than one function (or stat plot) is selected and graphed, press [▲] and [▼] to move the cursor from one function graph to another.

The cursor movement is based on the order of the functions as they appear in the Y= editor and not on the appearance of the functions as graphed on the screen. (However, the TI−73 starts with selected statistical plots first.)

The function number in the upper right corner of the display changes as you move to the various graphs.

To quit [TRACE] mode, select another screen by pressing the appropriate key, such as [WINDOW] or [ZOOM], or press [2nd] [QUIT] to return to the Home screen. Press [CLEAR] to stay on the Graph screen.

Using QuickZoom

While tracing, you can press [ENTER] to adjust the viewing window. The cursor location then becomes the center of the new viewing window, and the cursor remains in [TRACE] mode. This is called **QuickZoom**. If you do a **QuickZoom** accidentally, and you want to return to the zoom settings in the previous window, select [ZOOM] **MEMORY 1:ZPrevious**.

Controlling the Increments of a Trace

By assigning a specific value to **ΔX** (which is optional), you can control the *X* coordinates of a trace. **ΔX** is a **WINDOW** value; change it by pressing [WINDOW].

The TI−73 automatically calculates **ΔX** as:

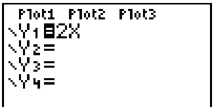
$$\Delta X = \frac{(\text{Xmax-Xmin})}{94}$$

If standard window values are set (**ZStandard**), **ΔX = .21276595744681**. If you assign a value to **ΔX**, the values for **Xmin** and **Xmax** are adjusted automatically according to the formula above.

Graph Y_1=2X with **ZStandard**.

1. Define **Y_1=2X** in the Y= editor.

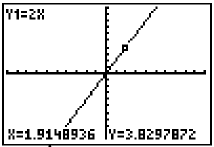
   [Y=] [CLEAR] **2** [x]



Note: Deselect any other functions by highlighting the corresponding = and pressing [ENTER].

2. Graph and trace the function.

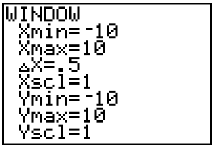
   [ZOOM] **6**
   [TRACE] [▶] [◀] (as necessary)



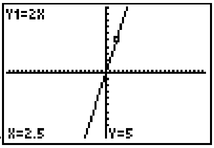
The TI‑73 chooses the **X**-value increments.

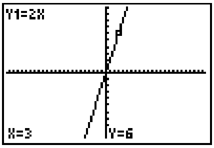
Assign .5 to **ΔX,** and graph and trace **Y_1=2X**.

[WINDOW] [▼] [▼] **. 5**



[TRACE]
[◀] and [▶]





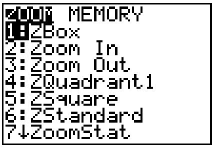
X- coordinates change in increments of .5.

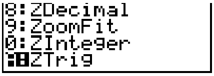
Adjusting Window Values with the [ZOOM] ZOOM Menu

The [ZOOM] **ZOOM** menu items allow you to adjust the viewing **WINDOW** of a graph quickly in a variety of ways. From the Graph screen, press [WINDOW] to see the adjusted **WINDOW** values.

1:ZBox, **2:Zoom In**, and **3:Zoom Out**, require you to move the cursor first to define the viewing window.

[ZOOM]





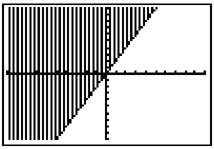
1:ZBox	Lets you draw a box around a specific section of the Graph screen. The calculator then zooms in on the area inside the box.
2:Zoom In	Lets you select a point with the cursor keys. The calculator then zooms in around the point by an amount defined by **SetFactors** (found on the [ZOOM] **MEMORY** menu).
3:Zoom Out	Lets you select a point with the cursor keys. The calculator then zooms out around the point by an amount defined by **SetFactors**.
4:ZQuadrant1	Displays Quadrant I only. Replots the graph immediately.
5:ZSquare	Adjusts **WINDOW** variables so that a square or a circle is shown in correct proportion (instead of a rectangle or an ellipse). Replots the graph immediately.

6:ZStandard	Sets the standard (default) **WINDOW** variables. Replots the graph immediately.
7:ZoomStat	Sets the **WINDOW** values for the current stat lists. Replots the graph immediately.
8:ZDecimal	Sets **ΔX** and **ΔY** to 0.1 and centers the origin. Replots the graph immediately; press [TRACE] to view the new coordinate values.
9:ZoomFit	Adjusts **Ymin** and **Ymax** so that the Graph screen displays the full range of *Y* variable values. Replots the graph immediately.
10:ZInteger	Lets you select a new center point, and then sets **ΔX** and **ΔY** to 1 and sets **Xscl** and **Yscl** to 10. Replots the graph immediately; press [TRACE] to view the new coordinate values.
11:ZTrig	Sets **WINDOW** variables to preset values that are often appropriate for graphing trig functions. Replots the graph immediately.

ZBox [ZOOM] 1

With **ZBox**, use the cursor keys to draw a box around a specific section of the Graph screen that you would like to view up close. The calculator then zooms in on the area inside the box with the cursor in the center of the screen.

Explore the function graph, Y_1=2X with **ZBox**.

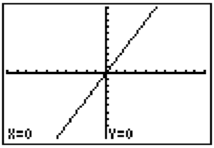


1. Display the graph of a selected function (the example shows **Y_1=2X**).

   [ZOOM] **6**

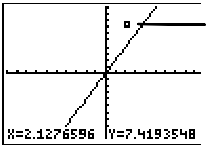
2. Select the **ZBox** function and return to the function graph.

   [ZOOM] **1**



3. Move the cursor to one corner of the box you want to define.

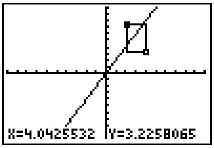
   [▶] [◀] [▲] [▼] [ENTER]



Cursor changes to a small square.

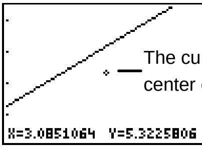
4. Move the cursor to the corner diagonally opposite from the first one.

   [▶] [◀] [▲] [▼]



5. Replot the graph.

   [ENTER]



The cursor is in the center of the new screen.

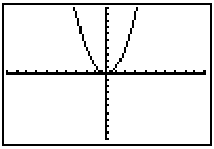
Zoom In and Zoom Out [ZOOM] 2 and 3

Zoom In magnifies the graph around the cursor location. **Zoom Out** displays a greater portion of the graph, centered on the cursor location, to provide a more global view. (The procedure is the same for both.)

After a **Zoom In** or **Zoom Out** operation is selected, move the cursor, as necessary, and press [ENTER] to select the new center point. Repeat the operation until another operation is selected or you exit the Graph screen.

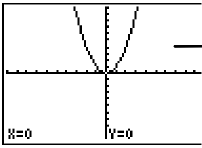
1. Display the graph of a selected function (the example shows $Y_1=X^2$).

   [ZOOM] **6**



2. Select the **Zoom In** operation for the function graph.

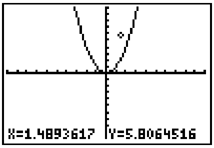
   [ZOOM] **2**



You want to zoom in on this side of the graph.

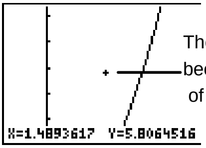
3. Move the cursor to the point that you want as the center of the new viewing window.

   [▶] [◀] [▲] [▼]



4. Replot the graph.

   [ENTER]



The cursor point becomes the center of the new window.

Zoom Out works exactly the same way as **Zoom In**. The calculator zooms out automatically around the center point.

ZStandard [ZOOM] 6

ZStandard is one of the more popular zoom commands because many function graphs look good when graphed according to the standard (default) **WINDOW** values: **Xmin=-10**, **Xmax=10**, **Xscl=1**, **Ymin=--10**, **Ymax=10**, **Yscl=1**.

If you select the **ZStandard** operation, either from the Graph screen or another screen, all selected functions are immediately replotted according to these standard **WINDOW** values.

ZInteger [ZOOM] 0

ZInteger requires you first to select a new center point. The calculator then replots the graph immediately using the adjusted **WINDOW** values which set **ΔX** and **ΔY** to 1, and **Xscl** and **Yscl** to 10.

Select the center point (as you would do for **ZoomIn** and **ZoomOut**) by moving the cursor with the cursor keys, and then pressing [ENTER]. Press [TRACE] to view the new coordinate values.

Other Zoom Operations

All other Zoom commands, **ZQuadrant**, **ZSquare, ZoomStat, ZDecimal**, **ZoomFit**, and **ZTrig**, replot immediately all selected functions and adjust **WINDOW** values according to their definitions. For **ZDecimal**, press [TRACE] to view the new coordinate values.

Examples of these operations are included in Appendix A: Function and Instruction Reference.

The [ZOOM] MEMORY Menu

[ZOOM] [▶]

1:ZPrevious	Replots all selected function graphs using the **WINDOW** variables of the graph that was displayed before you executed the last **ZOOM** operation.
2:SetFactors	Define the magnification or reduction factor used to **Zoom In** or **Zoom Out** around a cursor point. There are two: **XFact** and **YFact**.

ZPrevious [ZOOM] [▶] 1

Selecting **ZPrevious** automatically replots all selected functions and stat plots and adjusts **WINDOW** values according to the definition of the previous graph.

SetFactors [ZOOM] [▸] 2

The zoom factors, **XFact** and **YFact**, are positive real numbers ≥1. They define the magnification or reduction factor used to **Zoom In** or **Zoom Out** around a cursor point. The default values for both **XFact** and **YFact** are 4. Highlight the factor you want to change, press [CLEAR], and then enter the new value. **XFact** and **YFact** do not affect any other **Zoom** operations.

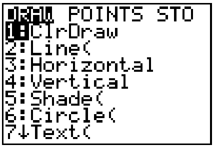
10 Drawing

The [DRAW] DRAW Menu

The [DRAW] **DRAW** menu items let you draw on top of function graphs and stat plots (see Chapter 9: Function Graphing and Chapter 6: Statistical Plots). The way the TI-73 interprets draw instructions depends on whether you accessed the menu items from the Home screen or the Program editor, or directly from a graph.

Note: Redefining **WINDOW** values, graphing a $\mathbf{Y_n}$ function or stat plot, or pressing [ZOOM] erases all drawn items from the Graph screen.

[DRAW]



1:ClrDraw	Clears all drawn elements.
2:Line(	Draws a line segment between two points.
3:Horizontal	Draws a horizontal line.
4:Vertical	Draws a vertical line.
5:Shade(	Shades an area between two functions.
6:Circle(	Draws a circle.
7:Text(	Draws text on a Graph screen.
8:Pen	Activates the free-form drawing tool.

When using a [DRAW] **DRAW** menu item or [DRAW] **POINTS** menu item to draw directly on a graph, the cursor coordinates are displayed if **CoordOn** is selected ([2nd] [FORMAT]). If a graph is not displayed when you select a [DRAW] **DRAW** menu item, the Home screen is displayed.

ClrDraw [DRAW] 1

ClrDraw clears all drawn elements from the Graph screen. All points, lines, and shading drawn with [DRAW] **DRAW** menu items are temporary. Therefore, if you leave the Graph screen, and then return, all drawings are erased.

If you select **ClrDraw** from the Graph screen, the current graph is replotted and displayed with no drawn elements. You can save drawings and recall them with the [DRAW] **STO** menu.

If you select **ClrDraw** from the Home screen or a program, it is pasted to the cursor location. Pressing [ENTER] executes the instruction, all drawings on the current graph are erased, and the message **Done** is displayed. When you display the graph again, all drawn elements disappear.

Line([DRAW] 2

Line(draws a line from point (X_1,Y_1) to (X_2,Y_2). You can execute the **Line(** instruction from the Graph screen, the Home screen or Program editor.

Line(from the Graph Screen

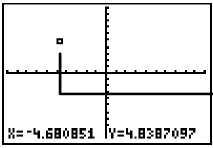
To draw a line on the Graph screen:

1. From the Graph screen, select [DRAW] **2**. The cursor appears in the middle of the Graph screen. The *X*- and *Y*-coordinates are shown at the bottom of the screen. If they are not, you can turn them on by selecting **CoordOn** ([2nd] [FORMAT]).
2. Position the cursor at the beginning point of the line segment that you want to draw, and then press [ENTER]. The cursor becomes a small box.
3. Move the cursor to the end point of the line segment, and then press [ENTER]. The line segment is drawn as you move the cursor.
4. Repeat steps 2 and 3, as necessary. To cancel **Line(**, press [CLEAR].

Draw a line segment from the Graph screen.

1. Clear all previous drawings, and select the beginning point of the line segment.

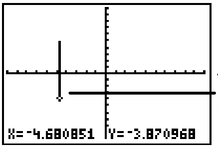
   [ZOOM] **6** [DRAW] **1** [DRAW] **2**
   [▼] [▲] [▶] [◀]
   (as necessary)
   [ENTER]



The cursor becomes a small box.

2. Select the ending point of the line segment.

   [▼] [▲] [▶] [◀]
   (as necessary)
   [ENTER]



The line segment is finished.

Line(from the Home Screen or Program Editor

From the Home screen or the Program editor, **Line(** can draw or erase a line segment from point (X_1,Y_1) to (X_2,Y_2) on the Graph screen.

You follow the **Line(** instruction with the coordinates of the beginning point (X_1,Y_1) and the ending point (X_2,Y_2) of the line segment. Including the argument, **0**, after the X and Y coordinates erases a line from (X_1,Y_1) to (X_2,Y_2).

To draw the line segment:

Line(X_1, Y_1, X_2, Y_2**)**

To erase a line segment:

Line(X_1, Y_1, X_2, Y_2**,0)**

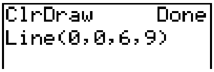
From the Home screen, draw a line segment from (0,0) to (6,9).

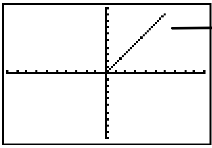
1. From the Home screen, clear the Graph screen.

   [2nd] [QUIT] [CLEAR]
   [DRAW] **1** [ENTER]

2. Specify the (X,Y) coordinates and draw the line segment.

   [DRAW] **2**
   0 [,] **0** [,] **6** [,] **9** [)] [ENTER]

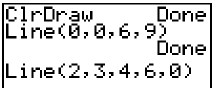


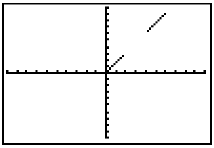


[WINDOW] is set to standard default values.

Erase the portion of the line from (2,3) to (4,6).

[2nd] [QUIT] [DRAW] **2**
2 [,] **3** [,] **4** [,] **6** [,] **0** [)]
[ENTER]





Horizontal and Vertical [DRAW] 3 and 4

Horizontal and **Vertical** draw a horizontal or vertical line on the Graph screen. You can execute both instructions from the Graph screen, Home screen or the Program editor.

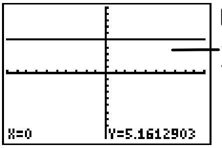
Horizontal and Vertical from the Graph Screen

To draw a horizontal or vertical line on the Graph screen:

1. From the Graph screen, select [DRAW] **3** or **4**. The cursor appears in the middle of the Graph screen. The *X*- and *Y*-coordinates are shown at the bottom of the screen.
2. A line is displayed that moves as you move the cursor. Place the cursor on the *Y*-coordinate (for horizontal lines) or the *X*-coordinate (for vertical lines) through which you want the line to pass.
3. Press [ENTER] to draw the line on the graph.
4. Repeat steps 2 and 3, as necessary. To cancel **Horizontal** or **Vertical**, press [CLEAR].

Draw a horizontal line from the Graph screen.

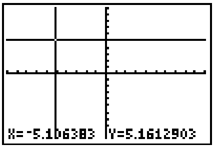
[GRAPH] [DRAW] **1** [DRAW] **3**
[▼] [▲] (as necessary)
[ENTER]



[WINDOW] is set to standard default values.

Draw a vertical line from the Graph screen.

[GRAPH] [DRAW] **4**
[▶] [◀] (as necessary)
[ENTER]



Horizontal and Vertical from the Home Screen or Program Editor

From the Home screen or the Program editor, **Horizontal** draws a horizontal line at $Y=y$. y can be an integer or an expression.

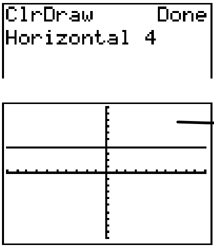
Horizontal y

Vertical draws a vertical line at X=x. x can be an integer or an expression.

Vertical x

From the Home screen, draw a horizontal line at Y=4.

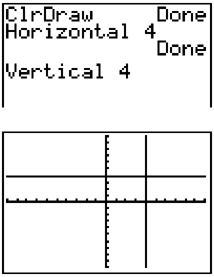
2nd [QUIT] CLEAR
DRAW **1** ENTER DRAW **3**
4 ENTER



WINDOW is set to standard default values.

Draw a vertical line at X=4.

2nd [QUIT] DRAW **4**
4 ENTER



Shade(DRAW 5

With **Shade(**, you can shade areas above and below functions on the Graph screen.

You can execute **Shade(** only from the Home screen or in a programming instruction. **Shade(** accepts two mandatory arguments and four optional arguments. However, you cannot skip any arguments. For example, if you want to specify the 5th argument, *pattern*, you also must specify the 3rd and 4th arguments, *left* and *right*.

Shade(*lower*,*upper*[,*left*,*right*,*pattern*,*res*]**)**

To use **Shade(** from the Home screen or a program:

1. Select [DRAW] **5**.
2. Enter two functions, *lower* and *upper*, in terms of *X*. After the instruction is executed, the calculator graphs the functions and shades above *lower* and below *upper*.
3. Enter *left* and *right*, the left and right *X* boundaries, if desired. **Xmin** and **Xmax** are the defaults.
4. Enter the shading pattern number, *pattern*, if desired. The four shading patterns are:

   1=Vertical (default)
   2=Horizontal
   3=Diagonal upper left to lower right
   4=Diagonal lower left to upper right

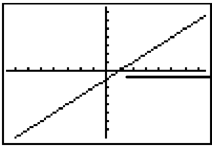
5. Specify the pattern resolution, *res*, an integer number between **1** and **8**, if desired.

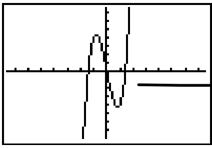
   res=**1** is the default and represents the lowest resolution (lines drawn very close together). *res*=**8** represents the highest resolution (lines drawn very far apart).

6. Press [ENTER] to execute the instruction.

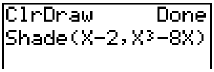
Shade above the function $Y=X-2$ (*lower*) and below the function $Y=X^3-8X$ (*upper*).

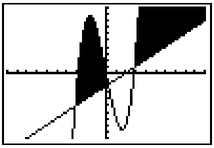
(The functions are shown to the right as they would look if graphed individually.)


Y=X-2


$Y=X^3-8X$

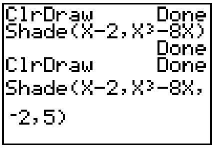
[2nd] [QUIT] [CLEAR]
[DRAW] **1** [ENTER]
[DRAW] **5** [x] [−] **2** [,] [x]
[MATH] **3** [−] **8** [x] [)] [ENTER]

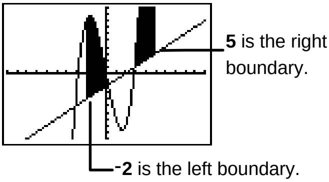




Enter a *left X* boundary, ⁻2, and a *right X* boundary, 5, for the same functions.

[2nd] [QUIT] [DRAW] **1** [ENTER]
[2nd] [ENTRY] [2nd] [ENTRY]
[◄] [,] [(−)] **2** [,] **5** [)] [ENTER]





Circle([DRAW] 6

You can execute the **Circle(** instruction from the Graph screen, Home screen, or the Program editor.

Circle(from the Graph Screen

To draw a circle on the Graph screen:

1. From the Graph screen, select [DRAW] **6**. The cursor appears in the middle of the Graph screen. The *X*- and *Y*-coordinates are shown at the bottom of the screen.

2. Place the cursor at the center point of the circle you want to draw. Press [ENTER].

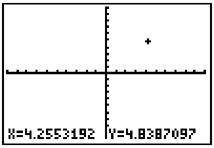
3. Move the cursor to a point on the circumference. Press [ENTER]. The circle is drawn automatically on the graph.

4. Repeat steps 2 and 3, as necessary. To cancel **Circle(**, press [CLEAR].

Draw a circle from the Graph screen.

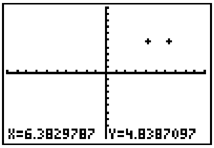
1. Clear all previous drawings, and select the center point of the circle.

   [GRAPH] [DRAW] **1** [DRAW] **6**
   [▼] [▲] [▶] [◀] (as necessary)
   [ENTER]



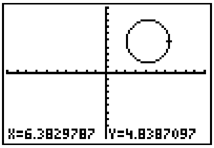
2. Move the cursor to a point on the circumference.

   [▲] [▼] [▶] [◀]
   (as necessary)



3. Draw the circle.

   [ENTER]



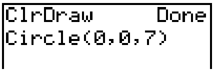
Circle(from the Home Screen or Program Editor

From the Home screen or the Program editor, you can draw a circle on the Graph screen. **Circle(** accepts three mandatory arguments: *X* and *Y*, the coordinates of the center point of the circle, and *radius*, the radius length which must be a positive real number.

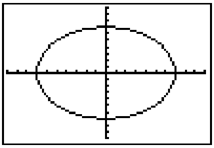
Circle(*X*,*Y*,*radius***)**

Draw a circle with center point=(0,0) and radius=7.

[2nd] [QUIT] [CLEAR] [DRAW]
1 [ENTER] [DRAW] **6**
0 [,] **0** [,] **7** [)] [ENTER]



Note: Use [ZOOM] **5:ZSquare** to adjust them and make the circle circular.



Text([DRAW] 7

You can access **Text(** from the Graph screen, Home screen, or the Program editor. **Text(** allows you to draw text on the Graph screen when a graph is displayed. Use the Text editor ([2nd] [TEXT]) to access all text characters. You may enter TI-73 functions, variables, and instructions as text. The font is proportional, so the exact number of characters you can place on the graph varies.

Text(from the Graph Screen

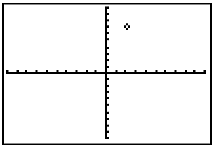
To draw text on the Graph screen:

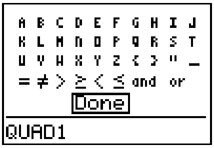
1. From the Graph screen, select [DRAW] **7**. The cursor appears in the middle of the Graph screen.
2. Place the cursor at the point where you want the text to begin.
3. Press [2nd] [TEXT] to display the Text editor. Select the text characters. Highlight **Done** with the cursor, and then press [ENTER]. The selected text is pasted onto the Graph screen.
4. Repeat steps 2 and 3, as necessary. To cancel **Text(**, press [CLEAR].

From the Graph screen, Label Quadrant I with **QUAD1**.

1. Clear all previous drawings, and select the beginning point where you want the text to start.

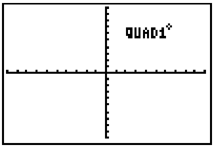
   [GRAPH] [DRAW] **1** [DRAW] **7**
   [▼] [▲] [▶] [◀]
   (as necessary)





2. Using the Text editor, enter **QUAD1**.

   [2nd] [TEXT] **Q** [ENTER]
   U [ENTER] **A** [ENTER]
   D [ENTER] **1** **Done** [ENTER]



Text(from the Home Screen or the Program Editor

From the Home screen or the Program editor, you can draw text on the Graph screen.

Text(accepts three mandatory arguments: *row* and *column*, which specify the pixel value of the top-left corner of the first character, and *text*, which can be functions, variables, or text instructions.

> **Text(***row*,*column*,*text***)**
> **Text(***row*,*column*,**"***text***")**

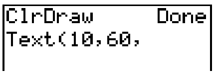
row is an integer between 0 and 57 and *column* is an integer between 0 and 94. Therefore, (0,0) is the top left corner, (0,94), is the top right corner, (57,0) is the lower left corner, and (57,94) is the lower right corner. If you try to draw *text* on any edge of the Graph screen, the calculator only displays *text* that fits; *text* does not wrap to the next *row*.

If *text* is surrounded by quotation marks (**" "**) (found in the Text editor), the calculator interprets any characters, numbers, or expressions as text. If the quotation marks are omitted, the TI−73 calculates and displays the result, if applicable, with up to 10 characters.

Label Quadrant I with **QUAD1** from the Home screen. Start the text at the pixel value of (10,60).

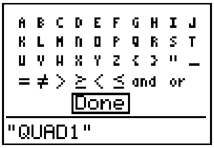
1. Clear all previous drawings, and select the beginning point of the text.

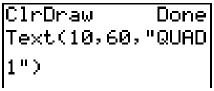
   [2nd] [QUIT] [CLEAR]
   [DRAW] **1** [ENTER]
   [DRAW] **7** **1 0** [,] **6 0** [,]

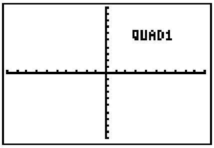


2. Using the Text editor, enter **"QUAD1"**.

   [2nd] [TEXT] **"** [ENTER]
   Q [ENTER] **U** [ENTER]
   A [ENTER] **D** [ENTER] **1**
   " [ENTER] **Done** [ENTER] [)]
   [ENTER]







Pen [DRAW] 8

Pen draws any shape you want, including irregular or unusual ones.

You can execute **Pen** only from the Graph screen. You cannot execute **Pen** from the Home screen or the Program editor.

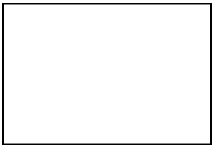
To draw your own shape on the Graph screen:

1. From the Graph screen, select [DRAW] **8**. The cursor appears in the middle of the Graph screen. The *X*- and *Y*-coordinates are shown at the bottom of the screen.
2. Place the cursor at the point where you want to begin drawing. Press [ENTER] to turn on the pen.
3. Move the cursor. As you move the cursor, you draw on the graph, shading one pixel at a time.
4. Press [ENTER] to turn off the pen.
5. Repeat steps 2, 3, and 4, as necessary. To cancel **Pen**, press [CLEAR].

Draw a happy face on the Graph screen.

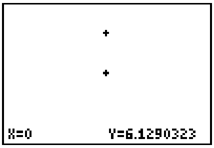
1. Clear all previous drawings, and then select **AxesOff**.

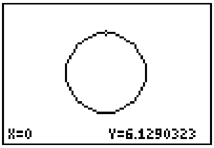
   [GRAPH] [DRAW] **1**
   [2nd] [FORMAT] [▼] [▼] [▶]
   [ENTER]



2. First draw a circle.

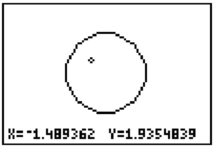
   [GRAPH] [DRAW] **6** [ENTER]
   [▲] (as necessary)
   [ENTER]

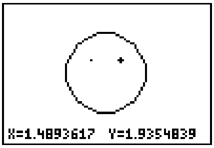




3. Use **Pen(** to draw the eyes.

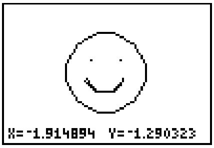
   [DRAW] **8**
   [▼] and [◀]
   (as necessary)
   [ENTER] [ENTER]
   [▶] (as necessary)
   [ENTER] [ENTER]





4. Draw the mouth.

   [▼] [ENTER] (to begin smile)
   [◀] [▼] (repeat as necessary)
   [◀] (repeat as necessary)
   [◀] [▲] (repeat as necessary)



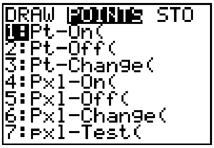
The [DRAW] POINTS Menu

The [DRAW] **POINTS** menu items let you draw or erase individual points or pixels on top of function graphs and stat plots (see Chapter 9: Function Graphing and Chapter 6: Statistical Plots). The way the TI-73 interprets the point instructions depends on whether you accessed the instructions from the Home screen or the Program editor, or directly from a graph.

Redefining **WINDOW** values ([ZOOM] **6:ZStandard**), graphing a $\mathbf{Y_n}$ function or stat plot, or pressing [ZOOM] erases all drawn items from the Graph screen.

Note: All examples in this section show the Graph screen set to standard **WINDOW** values and with all $\mathbf{Y_n}$ functions and stat plots deselected.

[DRAW] [▸]



1:Pt-On(	Turns on a point.
2:Pt-Off(	Turns off a point.
3:Pt-Change(	Toggles a point on or off.
4:Pxl-On(	Turns on a pixel.
5:Pxl-Off(	Turns off a pixel.
6:Pxl-Change(	Toggles a pixel on or off.
7:pxl-Test(	Returns 1 if pixel is on, 0 if pixel is off.

Pt-On(, Pt-Off(, and Pt-Change([DRAW] [▶] 1, 2, and 3

Pt-On(, **Pt-Off(**, and **Pt-Change(** turn on, off, or change the status of a point from the Graph screen, Home screen, or Program editor.

A point (as opposed to a pixel) is tied directly to the *X*- and *Y*-axes. The screen is divided into *X*- and *Y*-coordinates as specified by (*X*,*Y*). The points that you can view depend upon how the **WINDOW** values are defined.

For example, if standard **WINDOW** values are set, $^-10 \leq X \leq 10$ and $^-10 \leq Y \leq 10$. This does not mean that points outside these boundaries do not exist, only that you cannot see any turned on points outside these boundaries.

Pt-On(, Pt-Off(and Pt-Change(from the Graph Screen

To use **Pt-On(**, **Pt-Off(**, and **Pt-Change(** on the Graph screen:

1. From the Graph screen, select [DRAW] [▶] **1**, **2**, or **3**. The cursor appears in the middle of the Graph screen. The *X*- and *Y*-coordinates are shown at the bottom of the screen.
2. Move the cursor:
   - To the position where you want to draw the point (**Pt-On(**).
   - To the position of the point you want to erase (**Pt-Off(**).
   - To the position of the point you want to change (toggle on or off) (**Pt-Change(**).
3. Press [ENTER] to draw, erase, or change the point.
4. Repeat steps 2 and 3, as necessary. To cancel **Pt-On(**, **Pt-Off(**, or **Pt-Change(**, press [CLEAR].

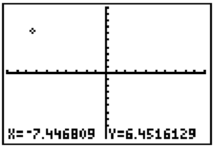
Draw points from the Graph screen.

1. Select **AxesOn,** if desired, and then clear all previous drawings.

   [2nd] [FORMAT] [▼] [▼] [ENTER]
   [GRAPH] [DRAW] **1**

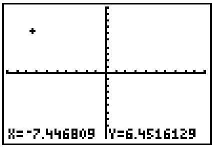
2. Select the beginning point where you want to draw the point.

   [DRAW] [▶] **1** [▶] [◀] [▲] [▼]
   (as necessary)

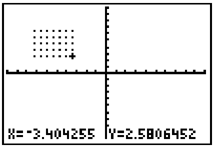


3. Draw the point.

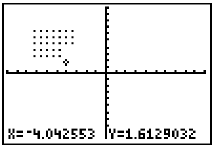
   [ENTER]



4. Repeat as necessary.



Erase four points from the Graph screen.



1. Move the cursor to the point you want to erase.

   [GRAPH] [DRAW] [▶] **2**
   [▶] [◀] [▲] [▼]
   (as necessary) [ENTER]

2. Repeat as necessary.

Pt-On(, Pt-Off(, and Pt-Change(from the Home Screen and Program Editor

From the Home screen or the Program editor, you can draw, erase, or change a point's status on the Graph screen.

Pt-On(, **Pt-Off(**, and **Pt-Change(** accept two mandatory arguments: *X* and *Y*, which specify the coordinates of the point that you want to draw, erase, or change. **Pt-On(** and **Pt-Off(** have one optional argument, *mark*, which determines the point's appearance. Specify **1** (default), **2**, or **3**, where:

1(default)= • (dot) **2**= □ (box) **3**= **+** (cross)

If you specify *mark* to turn on a point with **Pt-On(**, you must specify the same *mark* when you turn off the point with **Pt-Off(**. **Pt-Change(** does not have the *mark* argument.

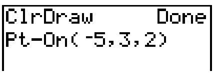
Also note that if, for example, you specify the point (20,30) but your viewing window is set to the standard values, you do not see the point since the viewing window does not include the specific part of the graph where (20,30) exists. Press [WINDOW] to redefine the **WINDOW** values.

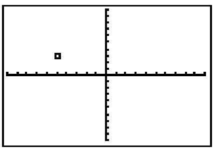
Note: Redefining **WINDOW** values, graphing a **Y**$_n$ function or stat plot, or pressing [ZOOM] erases all drawn items from the Graph screen.

Pt-On(*X*, *Y*[,*mark*]**)**
Pt-Off(*X*, *Y*[,*mark*]**)**
Pt-Change(*X*, *Y***)**

Turn on point (⁻5,3) and assign the box mark to it.

[2nd] [QUIT] [CLEAR]
[DRAW] **1** [ENTER] [DRAW] [▶]
1 [(-)] **5** [,] **3** [,] **2** [)] [ENTER]





Pxl-On(, Pxl-Off(, and Pxl-Change([DRAW] [▶] 4, 5, and 6

Pxl-On(, **Pxl-Off(**, and **Pxl-Change(** turn on, off, or change the status of a pixel only from the Home screen or the Program editor.

When you select a pixel instruction from the [DRAW] **POINTS** menu, the TI-73 returns you to the Home screen or the Program editor. Since the pixel instructions are not interactive, they cannot be used from the Graph screen.

A pixel is independent of the *X*- and *Y*- axes. It is based instead on the physical size of the screen. The screen is divided into pixels specified as (*row*,*column*). $0 \le row \le 62$ and $0 \le column \le 94$.

Pxl-On(, **Pxl-Off(**, and **Pxl-Change(** accept two mandatory arguments: *row* and *column*, which specify the pixel that you want to draw, erase, or change.

Pxl-On(*row*,*column***)**
Pxl-Off(*row*,*column***)**
Pxl-Change(*row*,*column***)**

Turn on the pixel at (45,35).

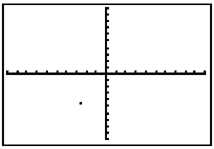
1. From the Home screen, clear the Graph screen.

   [2nd] [QUIT] [CLEAR]
   [DRAW] **1** [ENTER]

2. Turn on the pixel.

   [DRAW] [▶] **4**
   4 5 [,] **3 5** [)] [ENTER]

```
ClrDraw      Done
Pxl-On(45,35)
```



pxl-Test([DRAW] [▶] 7

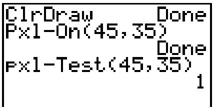
You can execute **pxl-Test(** only from the Home screen or the Program editor.

pxl-Test(tests a pixel at (*row*,*column*) to see if it is turned on or off. If it is on, **pxl-Test(** returns 1. If it is off, **pxl-Test(** returns 0. 0≤*row*≤57 and 0≤*column*≤94.

pxl-Test(*row*,*column***)**

Test to see if the pixel at (45,35) is turned on or off.

[2nd] [QUIT] [DRAW] [▶] **7**
4 5 [,] **3 5** [)] [ENTER]

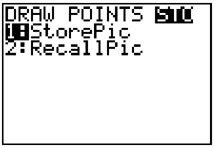


The [DRAW] STO Menu

The [DRAW] **STO** (store) menu lets you store or recall up to three pictures in memory. When you select an instruction from the [DRAW] **STO** menu, the TI-73 returns to the Home screen or the Program editor. The picture instructions are not interactive, which means you cannot use them from the Graph screen.

Note: All examples in this section show the Graph screen set to standard **WINDOW** values ([ZOOM] **6:ZStandard**) and with all $\mathbf{Y_n}$ functions and stat plots deselected.

[DRAW] [▶] [▶]



1:StorePic	Stores the current picture.
2:RecallPic	Recalls a stored picture.

StorePic [DRAW] [▶] [▶] 1

You can execute **StorePic** only from the Home screen or Program editor. You can store up to three pictures, each of which is an image of the current graph display, in picture variables **Pic1**, **Pic2**, or **Pic3**. Later, you can superimpose the stored picture onto a displayed graph from the Home screen or a program.

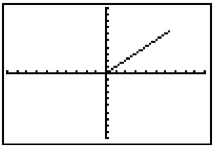
A picture includes drawn elements, plotted functions, axes, and tick marks. The picture does not include axes labels, lower and upper bound indicators, prompts, or cursor coordinates. Any parts of the display hidden by these items are stored with the picture.

StorePic accepts one mandatory argument, *number*, which specifies the number of the picture variable to which you want to store the picture. For example, if you enter **3**, the TI-73 stores the picture to **Pic3**. Pressing [ENTER] displays the current graph and stores the picture.

StorePic *number*

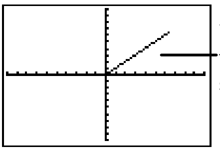
To see which graph variables have pictures stored to them, use the **PICTURE** secondary menu ([2nd] [VARS] **4:Picture**). Each variable **Pic1**, **Pic2**, and **Pic3** is marked as either **Defined** or **Empty**. If selected, the variable is pasted next to **StoPic**.

Store the drawn picture (to the right) in picture variable 2.



For more information on drawing lines, see the previous section entitled "Line(."

[2nd] [QUIT] [CLEAR]
[DRAW] [▶] [▶] 1 2 [ENTER]



You return to the Graph screen.

RecallPic [DRAW] [▸] [▸] 2

You can execute **RecallPic** only from the Home screen or Program editor. Use **RecallPic** to recall the graph picture stored in the picture variables **Pic1**, **Pic2**, or **Pic3**.

RecallPic accepts one mandatory argument, *number*, which specifies the number of the picture variable that you want to recall. For example, if you enter **3**, the TI−73 recalls **Pic3**. Pressing [ENTER] displays the current graph and superimposes **Pic3** on it. Since pictures are drawings, you cannot trace a curve that is part of a picture.

RecallPic *number*

To see which graph variables have pictures stored to them, use the **PICTURE** secondary menu ([2nd] [VARS] **4:Picture**). Each variable **Pic1**, **Pic2**, and **Pic3** is marked as either **Defined** or **Empty**. If selected, the variable is pasted next to **RecallPic**.

1. Clear the Graph screen.

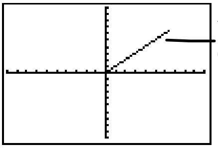
   [2nd] [QUIT]
   [DRAW] **1** [ENTER] [GRAPH]

```
StorePic 2    Done
ClrDraw       Done
```

2. Recall picture variable **2**. (A picture was stored to it in the previous example.)

   [DRAW] [▸] [▸] **2** **2** [ENTER]

```
StorePic 2    Done
ClrDraw       Done
RecallPic 2
```



You return to the Graph screen.

Deleting a Graph Picture

To delete graph pictures from memory, use the **MEMORY DELETE:Pic** menu ([2nd] [MEM] **4:Delete 7:Pic**).

11 Trigonometry

The [2nd] [TRIG] TRIG Menu

The [2nd] [TRIG] **TRIG** (trigonometry) menu accesses the trigonometric (trig) functions (**sin(**, **cos(**, **tan(**) and their inverses (**sin⁻¹(**, **cos⁻¹(**, **tan⁻¹(**).

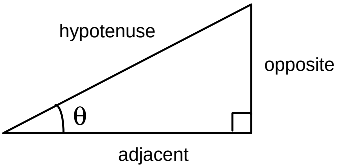
[2nd] [TRIG]

The sine, cosine, and tangent of an angle (θ) are defined by the lengths of the sides of a right triangle.

$$\sin\theta = \frac{\text{opposite}}{\text{hypotenuse}}$$

$$\cos\theta = \frac{\text{adjacent}}{\text{hypotenuse}}$$

$$\tan\theta = \frac{\text{opposite}}{\text{adjacent}}$$



Trig Functions [2nd] [TRIG] 1, 3, and 5

All trig functions return the sine, cosine, or tangent of a real number, expression, or a each element in a list. If *value* is a list, the calculator calculates the trig function of each element in the list, and a list is returned.

sin(*value***)**
cos(*value***)**
tan(*value***)**

For **tan**, *value* cannot be 90, 270, etc., or ⁻90, ⁻270, etc. In other words, since tan θ=sin/cos by definition, tan θ is undefined when cos θ=0.

Hint: This chapter's section entitled "Graphing Trig Functions" contains an example which graphs and traces **Y_1=tan(X)** to show undefined *Y* values for the function.

Inverse Trig Functions [2nd] [TRIG] 2, 4, and 6

The inverse trig functions calculate the smallest angle that gives a particular sine, cosine, or tangent. For example, **$\sin^{-1}$(.5)** calculates the angle whose sine is .5.

$\sin^{-1}$(*value***)**
$\cos^{-1}$(*value***)**
$\tan^{-1}$(*value***)**

For **$\cos^{-1}$** (also called arccosine) and **$\sin^{-1}$** (also called arcsine), $^-1 \le value \le 1$.

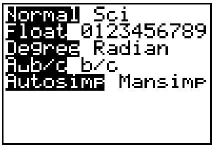
All inverse trig functions return the arcsine, arccosine, or arctangent of *value* or of each element in a list. If *value* is a list, the calculator calculates the inverse trig function of each element in the list, and a list is returned.

Angle Mode Settings

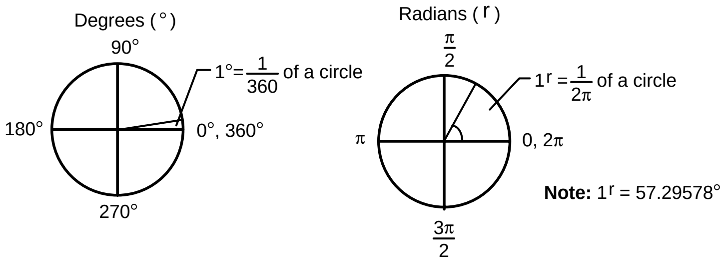
In trig calculations, angles are interpreted as degrees (°) or radians (^{r}), depending on the Angle mode setting, **Degree** or **Radian.**

Set the Angle mode from the mode screen.

[MODE]



Depending on the Angle mode, sin(1) is the sine of either 1° or 1^r. As you can see in the following illustration, 1° is not the same as 1^r. Therefore, sin(1°) ≠ sin(1^r). For correct results, enter angle values in the same units (degrees or radians) as the Angle mode setting.

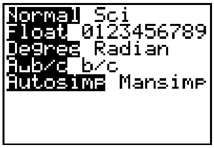


To perform a trig calculation, select the Angle mode for your value and then select the function. In **Radian** Angle mode, angles are often defined in terms of π.

Calculate sin(30) in both degrees and radians.

1. Select the **Degree** Angle mode.

   MODE ▼ ▼ ENTER

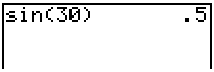


2. Return to the Home screen, and clear it, if desired.

   2nd [QUIT] CLEAR

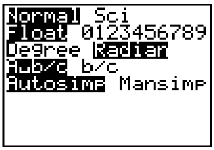
3. Enter **sin(30)**.

   2nd [TRIG] **1**
   3 0) ENTER



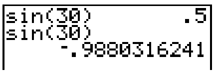
4. Change to the **Radian** Angle mode, and return to the Home screen.

   MODE ▼ ▼ ▶ ENTER
   2nd [QUIT]



5. Recall the previous entry to recalculate **sin(30)**.

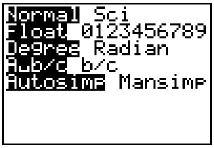
   2nd [ENTRY] ENTER



In **Degree** Angle mode, calculate $\tan^{-1}(1)$. Check your answer.

1. Select the **Degree** Angle mode.

   [MODE] [▼] [▼] [ENTER]

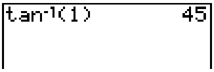


2. Return to the Home screen, and clear it, if desired.

   [2nd] [QUIT] [CLEAR]

3. Enter **tan⁻¹(1)**.

   [2nd] [TRIG] **6**
   1 [)] [ENTER]



4. Using the result, enter **tan(45)**.

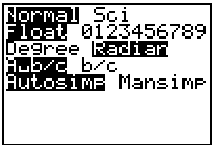
   [2nd] [TRIG] **5**
   4 5 [)] [ENTER]

This confirms the previous result.

In Radian mode, calculate $\cos(\pi/4)$.

1. Select the **Radian** Angle mode.

   [MODE] [▼] [▼] [▶] [ENTER]

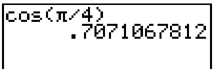


2. Return to the Home screen, and clear it, if desired.

   [2nd] [QUIT] [CLEAR]

3. Enter **cos(π/4)**.

   [2nd] [TRIG] **3**
   [2nd] [π] [÷] **4** [)] [ENTER]



Graphing Trig Functions

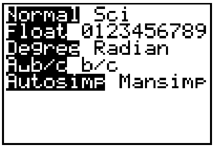
In addition to using the calculator to solve trig functions numerically, as described so far in this chapter, you can solve trig functions graphically.

For more information on generating function tables or graphing functions, see Chapter 8: Tables and Chapter 9: Function Graphing.

In **Degree** Angle mode, find four Y values where $\mathbf{Y_1}$**=tan(X)** is undefined. Check your answer by displaying the table for $\mathbf{Y_1}$.

1. Select **Degree** Angle mode, if necessary.

   [MODE] [▼] [▼] [ENTER]

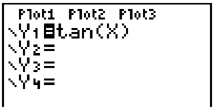


2. Deselect all $\mathbf{Y_n}$ functions.

   [2nd] [VARS] **2** **6** [ENTER]

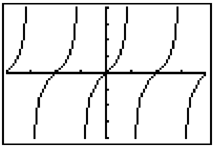
3. Define $\mathbf{Y_1}$**=tan(X)** in the Y= editor.

   [Y=] [CLEAR] [2nd] [TRIG] **5**
   [x] [)]



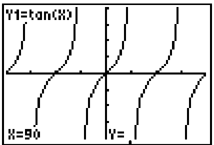
4. Graph the function using the **ZTrig** command.

   [ZOOM] [▲] [ENTER]



5. Trace the graph and observe where the Y value is undefined.

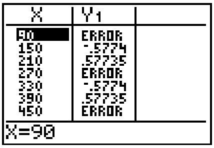
   [TRACE]
   [▶] and [◀], as necessary



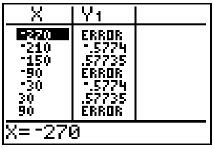
When **X=90**,
Y is undefined.

6. Use the table to check your result. Set **TblStart=90**, **ΔTbl=60**, **Indpnt=Auto** and **Depend=Auto**.

   [2nd] [TBLSET] **9 0** [▼] **6 0**
   [2nd] [TABLE]
   [▲] or [▼], as necessary



From these two screens, you know that **tan(X)** is undefined at **X**=-270, -90, 90, 270, 450.



The [2nd] [TRIG] ANGLE Menu

The **ANGLE** menu lets you specify the unit (degrees, radians, or DMS) of an angle, and it lets you convert an angle from one unit to another.

[2nd] [TRIG] [▶]

1:°	Designates an angle as degrees, regardless of the current Angle mode setting or DMS notation.
2:'	In DMS (degrees° minutes' seconds") notation, specifies the minutes.
3:"	In DMS (degrees° minutes' seconds") notation, specifies the seconds.
4:r	Specifies an angle as radians, regardless of the current Angle mode setting.
5:▶DMS	Converts an angle to DMS (degrees° minutes' seconds") notation.

Using ° and r to Specify Degrees and Radians [2nd] [TRIG] [▸] 1 and 4

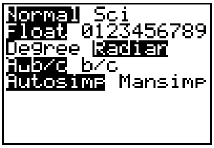
Normally, angles are interpreted according to the Angle mode setting. However, you can specify an angle as degrees or radians regardless of the Angle mode.

Suppose a series of trig calculations uses radians, but a few use degrees. Rather than change from **Radian** to **Degree** Angle mode and then back again, you can stay in the **Radian** Angle mode and specify some angles as degrees.

In **Radian** Angle mode, calculate sin(π/3). Then, without changing to the **Degree** Angle mode, calculate sin(60°).

1. Select **Radian** Angle mode.

   [MODE] [▾] [▾] [▸] [ENTER]

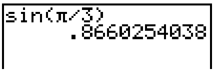


2. Return to the Home screen, and clear it, if desired.

   [2nd] [QUIT] [CLEAR]

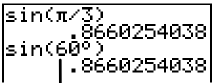
3. Enter **sin(π/3)**.

   [2nd] [TRIG] **1** [2nd] [π] [÷] **3** [)] [ENTER]



4. Use the ° designator to enter **sin(60°)**.

   [2nd] [TRIG] **1**
   6 0 [2nd] [TRIG] [▸] **1**
   [)] [ENTER]



60 is specified as degrees even in **Radian** angle mode. $\pi/3^r$=60°.

Likewise, you can use r to specify an angle as radians in the **Degree** Angle mode.

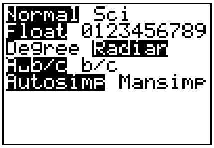
Converting between Degrees and Radians

Set the Angle mode to the unit you want to convert *to* because results are displayed according to the Angle mode setting. Then use ° or ^{r} to designate the unit to convert *from*.

Convert 50° to radians.

1. Set Angle mode to **Radian**.

   [MODE] [▼] [▼] [▶] [ENTER]

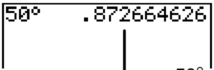


2. Return to the Home screen, and clear it, if desired.

   [2nd] [QUIT] [CLEAR]

3. Enter the value to convert, **50**. Use ° to specify it as degrees.

   5 0 [2nd] [TRIG] [▶] **1** [ENTER]

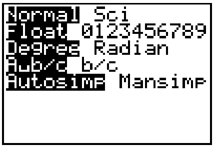


50° converted to radians.

Convert 50^r to degrees.

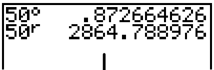
1. Set the Angle mode to **Degree**.

   [MODE] [▼] [▼] [ENTER]
   [2nd] [QUIT]



2. Enter the value to convert, **50**. Use ^{r} to specify it as radians.

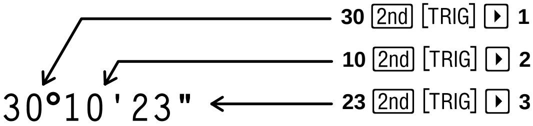
   5 0 [2nd] [TRIG] [▶] **4** [ENTER]



50^r converted to degrees.

Entering Angles in DMS Notation 2nd [TRIG] ▶ 1, 2, and 3

DMS (degrees° minutes' seconds") is often used for angles involving latitude and longitude. The degrees can be any real number; minutes and seconds must be ≥ 0. To enter an angle in DMS notation, use the 2nd [TRIG] **ANGLE** menu.

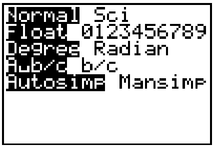


If you enter the angle *of a trig function*, where the angle is in DMS notation (as shown in the following example), the angle is interpreted as degrees, even in **Radian** mode.

Calculate sin(30°10'23") in **Degree** and **Radian** mode.

1. Select **Degree** Angle mode.

   MODE ▼ ▼ ENTER

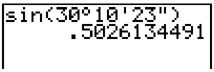


2. Return to the Home screen, and clear it, if desired.

   2nd [QUIT] CLEAR

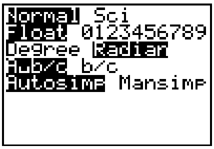
3. Enter **sin(30°10'23")**.

   2nd [TRIG] **1**
   3 0 2nd [TRIG] ▶ **1**
   1 0 2nd [TRIG] ▶ **2**
   2 3 2nd [TRIG] ▶ **3**)
   ENTER



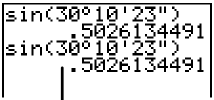
4. Select **Radian** Angle mode.

   MODE ▼ ▼ ▶ ENTER



5. Calculate **sin(30°10'23")**.

   2nd [QUIT]
   2nd [ENTRY] ENTER



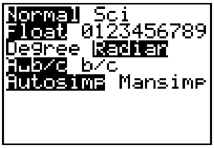
DMS notation overrides **Radian** mode.

In **Radian** mode, if you enter an angle only (*without a trig function*) in DMS notation (as shown in the following example), the angle is interpreted as degrees, but converted to a result in radians.

Convert 20°10'14" to radians.

1. Select **Radian** Angle mode.

   MODE ▼ ▼ ▶ ENTER

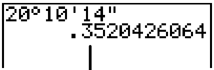


2. Return to the Home screen, and clear it, if desired.

   2nd [QUIT] CLEAR

3. Enter **20°10'14"**.

   2 0 2nd [TRIG] ▶ **1**
   1 0 2nd [TRIG] ▶ **2**
   1 4 2nd [TRIG] ▶ **3**
   ENTER



The result is in radians.

▸DMS [2nd] [TRIG] [▸] 5

To convert angles to DMS notation, use **▸DMS** from the [2nd] [TRIG] **ANGLE** menu.

*angle***▸DMS**

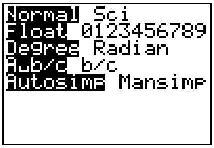
Entering ° overrides **Radian** mode. For example, if you enter **50°▸DMS** in **Radian** mode, the calculator still interprets 50 as degrees and displays the DMS equivalent.

In **Radian** mode, if you enter **50▸DMS** (no °), the calculator interprets 50 as radians, and then displays the DMS equivalent. For example, **50▸DMS** in **Radian** mode shows **2804°47'20.312"**. Likewise, in **Degree** mode, if you enter **50▸DMS** (no °), the calculator interprets 50 as degrees, and then displays the DMS equivalent. For example, **50▸DMS** in **Degree** mode shows **50°0'0"**.

In **Degree** Angle mode, convert 50.672° to DMS.

1. Select **Degree** Angle mode.

   [MODE] [▾] [▾] [ENTER]

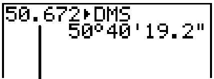


2. Return to the Home screen, and clear it, if desired.

   [2nd] [QUIT] [CLEAR]

3. Convert **50.672°** to DMS.

   50.672
   [2nd] [TRIG] [▸] **5**
   [ENTER]



Entering ° after 50.672 is optional in **Degree** mode.

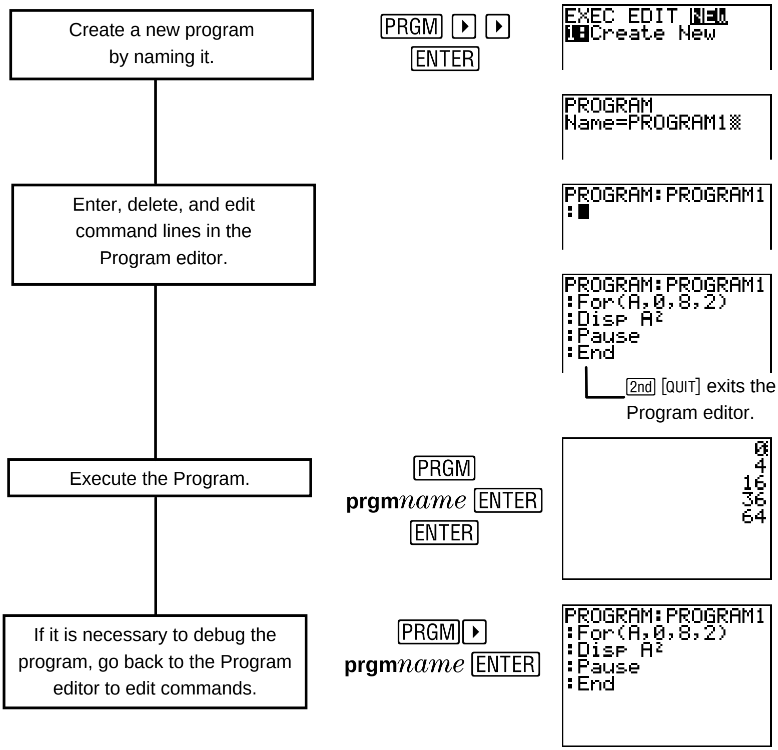
12 Programming

What Is a Program?

A program is a series of one or more programming commands to be executed by the calculator. Each command is an expression or instruction and begins with a colon (:). The number and size of programs that the TI–73 can store is limited only by available memory.

Steps for Creating a Program

Follow these basic steps when creating and executing a program. You may not have to do all of them each time.



Creating and Naming a New Program

You create a new program by selecting **1:Create New** from the [PRGM] **NEW** menu. You then are prompted to name the new program.

[PRGM] [▶] [▶]	

1:Create New	Creates a new program and displays the **PROGRAM Name=** screen, prompting you to name the new program.

Create New [PRGM] [▶] [▶] 1

After you select **1:Create New** from the [PRGM] **NEW** menu, the TI-73 displays **Name=** to prompt you to name the new program. A program name can be one to eight characters long. The first character must be a letter from A to Z. The second through eighth characters can be any combination of letters and numbers.

Access letters from the Text editor ([2nd] [TEXT]). If you type a name with more than eight characters, the calculator accepts the first eight characters and disregards the rest.

Create a new program and name it **PROGRAM1**.

1. Display the [PRGM] **NEW** menu.

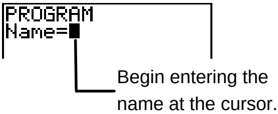
   [PRGM] [▶] [▶]

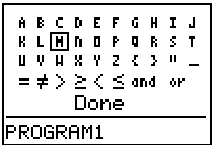
2. Select **1:Create New**.

   [ENTER]

3. Enter **PROGRAM1** at the cursor.

   2nd [TEXT] **P** ENTER
   R ENTER **O** ENTER
   G ENTER **R** ENTER
   A ENTER **M** ENTER **1**
   Done ENTER





4. Display the Program editor with the name of the program on the top line.

   ENTER

The Program Editor

You use the Program editor to enter and edit program commands. Enter the Program editor in one of two ways:

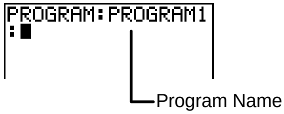
- Create and name a new program from the PRGM **NEW** menu with **1:Create New**. Once the name is entered, the calculator automatically enters the Program editor with the name of the program on the top line.

- Select a program to edit from the PRGM **EDIT** menu (PRGM ▸). The calculator automatically enters the Program editor with the name of the program on the top line.

Each new command line in the Program editor begins with a colon (:). Enter programming commands at the cursor. You can enter more than one command per line. To do this, separate the two commands with a colon (2nd [CATALOG] ▴ ▴ ▴ ENTER). If the command line is longer than the screen is wide, the command wraps to the next screen line.

To exit the Program editor, press 2nd [QUIT]. All commands are automatically saved.

[PRGM] [▸] *prgmName*
– or –
[PRGM] [▸] [▸] **1** [2nd] [TEXT]
prgmName **D o n e**
[ENTER] [ENTER]



Entering Program Commands

The calculator contains built-in programming commands on three menus. You access these menus by pressing [PRGM] from the Program editor. The first two menus, the [PRGM] **CTL** menu and the [PRGM] **I/O** menu, are discussed extensively in the next two sections. The third menu, the [PRGM] **EXEC** menu, lets you call existing programs as subroutines. It is discussed in the section entitled, "Executing a Program."

Entering Functions, Instructions, and Variables

In the Program editor, you also can select from function menus on the calculator ([MATH], [2nd] [CONVERT], etc.), change settings ([MODE], [2nd] [TBLSET], etc.), as well as select function keys ([b/c], [x^2], etc.). Simply press the appropriate key, and the function, instruction, or mode setting is pasted at the cursor location in the Program editor. Also, remember that all instructions and functions are listed in the **CATALOG** ([2nd] [CATALOG]).

Programs can access variables and lists saved in memory. If a program stores a new value to a variable or list, the program changes the value in memory during execution.

The following menus or keystroke sequences change appearance or operate differently when accessed from the Program editor:

- [PRGM] (accesses programming command menus)
- [2nd] [PLOT] (changes appearance)
- [2nd] [SET] (changes appearance)
- [2nd] [TBLSET] (changes appearance)

- [DRAW] **DRAW** (excludes **8:Pen)**
- [MATH] **MATH** (excludes **6:Solver)**
- [ZOOM] **MEMORY** (excludes **2:SetFactors)**

Exiting the Program Editor

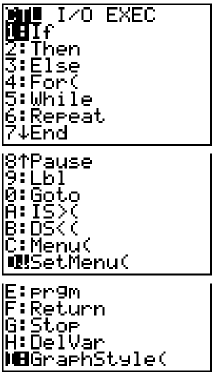
Pressing [Y=], [WINDOW], [GRAPH], [2nd] [MEM], [2nd] [QUIT], or [LIST] exits the Program editor and displays the applicable screen. The calculator automatically saves all command lines in memory whenever you exit the Program editor.

The [PRGM] CTL Menu

You can only access the [PRGM] **CTL** (control) menu by pressing [PRGM] from the Program editor. These programming commands help control the flow of an executing program. They make it easy to repeat or skip a group of commands (*block*) during program execution.

If, **For(**, **While**, **Repeat, IS>(,** and **DS<(** check a defined condition to determine which command to execute next. Conditions frequently use relational or Boolean tests (Chapter 2: Math Operations). When you select an item from the menu, the name is pasted to the cursor location on a command line in the program. To return to the Program editor without selecting an item, press [CLEAR].

[PRGM] (from the Program editor only)



1:If	Creates a conditional test.
2:Then	Executes commands when **If** condition is true.
3:Else	Executes commands when **If** condition is false.
4:For(	Creates an incrementing loop.
5:While	Creates a conditional loop.
6:Repeat	Creates a conditional loop.
7:End	Signifies the end of a block.
8:Pause	Pauses program execution.
9:Lbl	Defines a label.
0:Goto	Goes to a label.
A:IS>(	Increments and skips if greater than.
B:DS<(	Decrements and skips if less than.
C:Menu(	Defines menu items and branches.
D:SetMenu(	Views and modifies variables on a menu.
E:prgm	Executes a program as a subroutine.
F:Return	Returns from a subroutine.
G:Stop	Stops execution.
H:DelVar	Deletes a variable from within program.
I:GraphStyle(	Designates the graph style to be drawn.

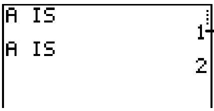
If [PRGM] 1

Use **If** to execute one *command* depending upon *condition*. If *condition* is true (non-zero), then *command1* is executed. If *condition* is false (zero), then *command1* is skipped. **If** instructions can be nested.

:**If** *condition*
:*command1* (if true)
:*command2*

Write a program named **COUNT** that adds one to variable *A* and displays the current value until *A*≥2.

```
PROGRAM:COUNT
:0→A
:Lbl Z
:A+1→A
:Disp "A IS",A
:Pause
:If A≥2
:Stop
:Goto Z
```



Press [ENTER].

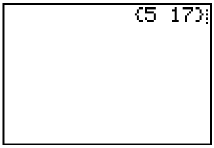
If-Then [PRGM] 1 and 2

Use **If** with **Then** to execute more than one command (*block*) depending upon *condition*. If *condition* is true (non-zero), then *block* is executed. If *condition* is false (zero), then *block* is skipped. **End** identifies the end of the *block*. Both **Then** and **End** must be on a line by itself.

:**If** *condition*
:**Then**
:*block* (if true)
:**End**
:*command*

Write a program named **TEST** that tests the values of variable *X*. If *X*<10, manipulate *X* and *Y* and then display both values. If *X*≥10, then display *X* and *Y* (without manipulating them).

```
PROGRAM:TEST
:1→X:10→Y
:If X<10
:Then
:2X+3→X
:2Y-3→Y
:End
:Disp {X,Y}
:Pause
```



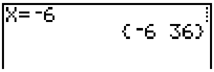
If-Then-Else PRGM 1, 2, and 3

Use **If** with **Then** and **Else** to execute only one of two *blocks* of commands depending upon *condition*. If *condition* is true (non-zero), then *block1* is executed. If *condition* is false (zero), then *block2* is executed. **End** identifies the end of *block2*. **Then**, **Else**, and **End** each must be on a line by itself.

:**If** *condition*
:**Then**
:*block1* (if true)
:**Else**
:*block2* (if false)
:**End**
:*command*

Write a program named **TESTELSE** that tests an input value, X. If $X<0$, then square it and store it to Y. If $X\geq0$, then store it to Y. Display X and Y.

```
PROGRAM:TESTELSE
:Input "X=",X
:If X<0
:Then
:X²→Y
:Else
:X→Y
:End
:Disp {X,Y}
:Pause
```



For([PRGM] 4

Use **For(** to control how many times a loop is repeated. A **For(** command loops to repeat the same group of commands (*block*) and increments to control the number of times the loop is repeated.

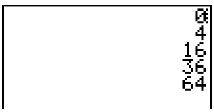
It executes commands in *block* through *end*, increasing *variable* from *begin* by *increment* until *variable*>*end*. *increment* is optional (default=1) and can be negative (*end*<*begin*). *end* is a maximum or minimum value not to be exceeded, which identifies the end of the loop. **End** identifies the end of *block*. When *variable*>*end*, the program executes each *command* following **End**. **For(** loops can be nested.

:**For(***variable*,*begin*,*end*[,*increment*]**)**
:*block* (while *variable* ≤ *end*)
:**End**
:*command*

Write a program named **SQUARE** that displays A^2, where 0=*begin*, 8 =*end*, and 2=*increment*.

```
PROGRAM:SQUARE
:For(A,0,8,2)
:Disp A²
:Pause
:End
```

Press [ENTER] between results.



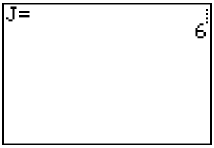
While [PRGM] 5

Use **While** to test *condition* before the commands in the loop are executed. **While** performs a *block* of commands WHILE *condition* is true (non-zero). *condition* is frequently a relational test (Chapter 2: Math Operations) and is tested when **While** is encountered. **End** identifies the end of *block*. When *condition* is false (zero), the program executes each command following **End**. **While** instructions can be nested.

:While *condition*
:*block* (while *condition* is true)
:End
:*command*

Write a program named **LOOP** that increments two variables, *I* and *J*, and displays the value of *J* when *I*≥6.

```
PROGRAM:LOOP
:0→I
:0→J
:While I<6
:J+1→J
:I+1→I
:End
:Disp "J=",J
:Pause
```



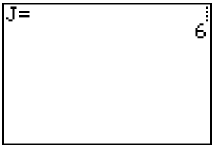
Repeat [PRGM] 6

Use **Repeat** to test *condition* after the commands in the loop are executed. **Repeat** executes *block* UNTIL *condition* is true (non-zero). It is similar to **While**, but *condition* is tested when **End** is encountered; therefore, the group of commands is always executed at least once. When condition is *false* (zero), **Repeat** instructions can be nested.

:Repeat *condition*
:*block* (until *condition* is true)
:End
:*command*

Write a program named **RPTLOOP** that increments two variables, *I* and *J*, and displays the value of *J* while *I*≥6.

```
PROGRAM:RPTLOOP
:0→I
:0→J
:Repeat I≥6
:J+1→J
:I+1→I
:End
:Disp "J=",J
:Pause
```



End [PRGM] 7

End identifies the end of a group of commands. You must include an **End** instruction at the end of each **For(**, **While**, or **Repeat** loop. Also, you must enter an **End** instruction at the end of each **If-Then** group and each **If-Then-Else** group.

:End

Pause [PRGM] 8

After a program has been executed, the screen is erased. Therefore, **Pause** is useful to suspend program execution until you press [ENTER], or to display *value* (such as answers or graphs) and suspend program execution until you press [ENTER]. During the pause, the pause indicator is on in the top-right corner. Press [ENTER] to resume execution.

Pause without *value* temporarily pauses the program. If the **DispGraph** or **Disp** instruction has been executed, the appropriate screen is displayed.

:Pause

Pause with *value* displays *value* on the Home screen. *value* can be scrolled.

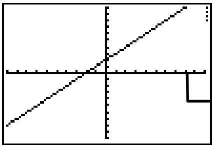
:Pause *value*

Write a program named **PAUSE** that stores a value to A, an equation to **Y₁**, graphs **Y₁** using standard **WINDOW** values (**ZStandard**), pauses, and then displays A.

```
PROGRAM:PAUSE
:2→A
:FnOff
:"X+A"→Y1
:ZStandard
:Pause
:Disp "A=",A
:Pause
```

For **FnOff** and **Y₁**, press [2nd] [VARS] **2** **1**.

For **ZStandard**, press [ZOOM] **6**.



Press [ENTER] to get to next screen.

Press [ENTER] to end execution.

Lbl and Goto [PRGM] 9 and 0

Lbl (label) and **Goto** are used together for branching.

Lbl gives a name (*label*) to a particular location in a program. *label* can be one or two text characters (**A** through **Z**, **0** through **99**).

:Lbl *label*

Goto causes the program to branch to *label* when **Goto** is encountered.

:Goto *label*

Write a program named **SQUARE2** that asks for an input, A, squares A, and then displays A until $A \geq 100$.

```
PROGRAM:SQUARE2
:Lbl 99
:Input A
:If A≥100
:Stop
:Disp A²
:Pause
:Goto 99
```

Press [ENTER] after answer is displayed.

IS>([PRGM] A

IS>((increment and skip if greater than) is used for testing and branching. IS>(adds 1 to *variable*. If the answer is > *value* (which can be an expression), then *command1* is skipped; if the answer is ≤ *value*, then *command1* is executed. *command2* is always executed. *variable* cannot be a system variable. **IS>(** is not a looping instruction.

:IS>(*variable*,*value***)**
:*command1* (if answer ≤ *value*)
:*command2*

Write a program named **ISKIP** that displays *A* until *A*>5.

```
PROGRAM:ISKIP
:0→A
:Lbl S
:Disp A
:Pause
:IS>(A,5)
:Goto S
:Disp "A IS NOW >5"
:Pause
```

Press [ENTER] between **A** values.

DS<([PRGM] B

DS<((decrement and skip if less than) is used for testing and branching. **DS<(** subtracts 1 from *variable*. If the answer is < *value* (which can be an expression), then *command1* is skipped; if the answer is ≥ *value*, then *command1* is executed. *command2* is always executed. *variable* cannot be a system variable. **DS<(** is not a looping instruction.

:DS<(*variable*,*value***)**
:*command1* (if answer ≥ *value*)
:*command2*

Write a program named **DSKIP** that displays *A* until *A*<5.

```
PROGRAM:DSKIP
:9→A
:Lbl S
:Disp A
:Pause
:DS<(A,5)
:Goto S
:Disp "A IS NOW <5"
:Pause
```

Press [ENTER] between **A** values.

Menu([PRGM] C

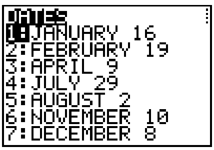
Menu(generates a menu of up to seven items during program execution. The pause indicator stays on until you select a menu item. The calculator then branches to the *label* corresponding with that *item*.

The menu *title* is enclosed in quotation marks (**" "**) and can have up to 16 characters. Up to seven pairs of menu *items* follow. Each pair consists of a text *item* (also enclosed in quotation marks) to be displayed as a menu selection, and a *label* item to which to branch if you select the corresponding menu selection.

:Menu("*title***","***item1***",***label1*[,**"***item2***"**,*label2*,…]**)**

Write a program named **DATES** that displays a menu of dates. Label the title "**DATES**", and label option one "**JANUARY 16**" with **A**, label option two "**FEBRUARY 19**" with **B**, label option three "**APRIL 9**" with **C**, label option four "**JULY 29**" with **D**, label option five "**AUGUST 2**" with **E**, label option six "**NOVEMBER 10**" with **F**, and label option seven "**DECEMBER 8**" with **F.**

```
PROGRAM:DATES
:Menu("DATES","JANUARY
16",A,"FEBRUARY 19",B,"APRIL
9",C,"JULY 29",D,"AUGUST
2",E,"NOVEMBER 10",F,"DECEMBER
8",G)
```



The program above pauses until you select 1, 2, 3, 4, 5, 6, or 7. If you select **2:FEBRUARY 19**, for example, the menu disappears and the program continues execution at **Lbl B**.

SetMenu([PRGM] D

Like **Menu(**, **SetMenu(** sets up a menu of up to seven *items*. During program execution, the user assigns (and edits, as necessary) numerical values to each item. To assign a value, enter the value using the number keys, and then press [ENTER] or [▼].

Long values do not wrap; they scroll off the screen, and an ellipsis (…) is displayed. Use [▶] and [◀] to scroll the whole value. Use [▲] and [▼] to move between menu items as necessary.

:SetMenu("*title***","***item1***",***variable1*[**,"***item2***",***variable2*,…]**)**

The menu *title* is enclosed in quotation marks (**" "**) and can have up to 16 characters. Up to seven menu *items* (also enclosed in quotation marks) follow. During program execution, the menu displays the first 10 characters of *item*. Each *item* needs a corresponding *variable* where the entered value is stored.

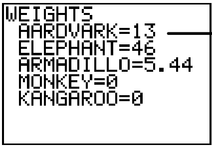
The values you enter for the *variables* (assigned to the menu items) are stored in the calculator's memory. Also, if you assign to a menu item a *variable* that has been previously defined in the calculator's memory, that value displays when you first execute the program.

Press [2nd] [QUIT] to exit the menu and end program execution.

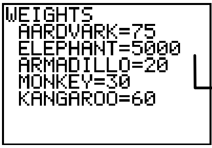
Write a program named **SETMENU** that displays a menu of animal weights. Label the title "**WEIGHTS**", show weight values of five different animals, and allow the user to change the weight values.

```
PROGRAM:SETMENU
:SetMenu("WEIGHTS","AARDVARK",
A,"ELEPHANT",B,"ARMADILLO",
C,"MONKEY",D," KANGAROO",E)
```

For example, the **ARMADILLO**'s weight is assigned to variable **C**.



A, **B**, and **C** were previously defined.



Enter new weights.

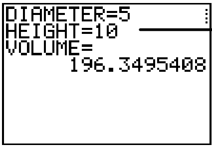
prgm [PRGM] E

Use **prgm** to execute other programs as subroutines. When you select **prgm**, it is pasted to the cursor location. Use the Text editor to enter the characters needed to spell a program name. Using **prgm** is equivalent to selecting existing programs from the [PRGM] **EXEC** menu (see the section in this chapters entitled "*Calling a Program from Another Program*"; however, it allows you to enter the name of a program that you have not yet created.

:prgm*name*

Write two programs named **CALCAREA** and **VOLUME**. **CALCAREA** calculates the area of a circle. **VOLUME** inputs the circle diameter *D*, and height *H*, calls **CALCAREA** as a subroutine, which calculates the area using *D* and *H*, and then displays the volume of a cylinder .

```
PROGRAM:VOLUME
:Input "DIAMETER=",D
:Input "HEIGHT=",H
:prgmCALCAREA
:A*H→V
:Disp "VOLUME=",V
:Pause
```



Press [ENTER] after inputs.

```
PROGRAM:CALCAREA
:D/2→R
:π*R²→A
:Return
```

Return [PRGM] F

Return quits the subroutine and returns execution to the calling program, even if it is encountered within nested loops. Any loops are ended. An implied **Return** exists at the end of any program that is called as a subroutine. Within the main program, **Return** stops execution and returns to the Home screen.

:Return

See the program examples (on the previous page), **CALCAREA** and **VOLUME**, explaining the programming command, **prgm**. The subroutine, **CALCAREA**, ends with a **Return** command.

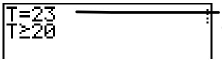
Stop [PRGM] G

Stop ends program execution and returns to the Home screen. **Stop** is optional at the end of a program.

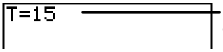
:Stop

Write a program named **STOP** that inputs T. If $T \geq 20$, then the program displays $T \geq 20$. If $T < 20$, then the program stops execution. (**Note:** The example screens show two program executions so that you can see what happens with both types of input.)

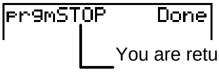
```
PROGRAM:STOP
:Input "T=",T
:If T≥20
:Then
:Disp "T≥20"
:Pause
:Else
:Stop
```



Press [ENTER] after input.



Press [ENTER] after input.



You are returned to the Home screen.

DelVar [PRGM] H

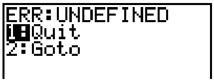
DelVar (delete variable) deletes the contents of *variable* from memory. You cannot delete a program or a system variable.

:DelVar *variable*

Write a program named **DELVAR** that deletes the value for variable *A* from the calculator's memory.

```
PROGRAM:DELVAR
:{1,2}→L1
:Disp L1
:Pause
:DelVar L1
:Disp L1
:Pause
```

Press [ENTER].



GraphStyle([PRGM] I

GraphStyle(defines one of seven graph style *types* for Y_n. Y_n=**1**, **2**, **3**, or **4** (for $\mathbf{Y_1}$, $\mathbf{Y_2}$, $\mathbf{Y_3}$, or $\mathbf{Y_4}$). The *type* icons described below are located to the left of $\mathbf{Y_n}$ in the Y= editor.

1 = (line)

2 = (thick line)

3 = (shade above)

4 = (shade below)

5 = (path)

6 = (animate)

7 = (dot)

:GraphStyle(Y_n,*type***)**

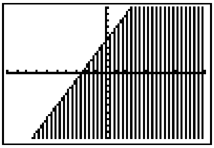
For a detailed description of each graph style, see Chapter 9: Function Graphing.

Write a program named **GRPHSTYL** that defines the *shade below* graph style for **Y_1=2X+5** and graphs it.

```
PROGRAM:GRPHSTYL
:FnOff
:"2X+5"→Y1
:GraphStyle(1,4)
:ZStandard
```

For **FnOff** and **Y_1**, press [2nd] [VARS] **2 1.**

For **ZStandard**, press [ZOOM] **6**.

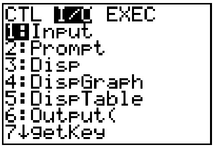


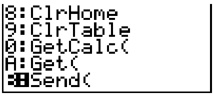
The [PRGM] I/O Menu

You can only access the [PRGM] **I/O** (input/output) menu by pressing [PRGM] [▸] from the Program editor. The [PRGM] **I/O** menu instructions allow you to input values and output answers during program execution.

To return to the Program editor without selecting an item, press [CLEAR].

[PRGM] [▸] (from the Program editor only)





1:Input	Lets the user enter a value or display a graph.
2:Prompt	Prompts the user to enter variable values.
3:Disp	Displays text or values on the Home screen.
4:DispGraph	Displays the current graph.

5:DispTable	Displays the current table.
6:Output(	Displays text or values at a specified position.
7:getKey	Checks the keyboard for a keystroke.
8:ClrScreen	Clears the Home screen.
9:ClrTable	Clears the current table.
0:GetCalc(	Gets a variable from another TI-73.
A:Get(	Gets a variable from the CBL 2/CBL or CBR.
B:Send(	Sends a variable to the CBL 2/CBL or CBR.

Input [PRGM] [▸] 1

Input functions in two different ways. You can use it to store to a variable value or to display the current graph.

Storing to a Variable

Input accepts input and stores it to *variable*. When the program is executed, a **?** (question mark) prompt (unless otherwise defined) is displayed. Enter a real number, a list name, or a $\mathbf{Y_n}$ function. Then press [ENTER], which tells the calculator to evaluate the input and store the value to *variable*.

:Input *variable*

To input lists and expressions during program execution, you must use the Text editor to include braces ({}) around the list elements and quotation marks (**" "**) around the expressions and $\mathbf{Y_n}$ functions.

You also can display text of up to 16 characters as a prompt. During program execution, enter a value after the prompt, and then press [ENTER]. The value is stored to *variable*, and the program resumes execution.

:Input "*text***",***variable*

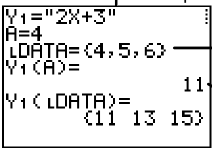
Write a program named **INPUTVAR** that inputs two sets of data and a function, and then solves the function using both of the data sets.

```
PROGRAM:INPUTVAR
:Input "Y1=",Y1
:Input "A=",A
:Input "LDATA=",LDATA
:Disp "Y1(A)=",Y1(A)
:Pause
:Disp "Y1(LDATA)=",Y1(LDATA)
:Pause
```

For $\mathbf{Y_1}$, press [2nd] [VARS] **2 1.**

For **L**, press [2nd] [STAT] [▶] **9.**

Surround $\mathbf{Y_1}$ with quotation marks.



Enclose list in { }.

Press [ENTER] between data displays.

Displaying the Current Graph

Input, with no arguments, displays the current graph. Once the graph screen is shown, you can move the free-moving cursor, which updates **X** and **Y** by a value of .1. The pause indicator is displayed. Press [ENTER] to resume program execution. The Home screen then displays the X- and Y-coordinates.

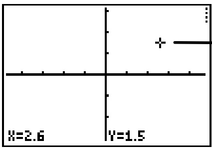
:Input

Write a program named **GRPHINPT** that gets input from the graph screen (the (X,Y) coordinates of the cursor's position) and displays the values on the Home screen.

```
PROGRAM:GRPHINPT
:FnOff
:PlotsOff
:ZDecimal
:Input
:Disp X,Y
:Pause
```

For **FnOff**, press [2nd] [VARS] **2 6.** For **PlotsOff**, select [2nd] [PLOT] **4.**

For **ZDecimal**, press [ZOOM] **8.**



Move cursor to desired location, and press [ENTER].

X=2.6; Y=1.5.

Prompt [PRGM] [▸] 2

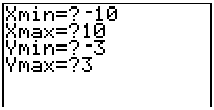
During program execution, **Prompt** displays the specified *variables* followed by **=?**, one at a time on separate lines. During program execution, the user enters a value or expression for each *variable*, and then presses [ENTER]. The values are stored, and the program resumes execution. **Y_n** functions are not valid with **Prompt**.

:Prompt *variableA*[,*variableB*,*variableC*…]

Write a program named **WINDOW** that requests inputs to be stored to **WINDOW** variables.

```
PROGRAM:WINDOW
:Prompt Xmin
:Prompt Xmax
:Prompt Ymin
:Prompt Ymax
```

For **WINDOW** variables, press [2nd] [VARS] **1**.



Disp [PRGM] [▸] 3

Disp displays one or more variable *values* during program execution. To display text, surround the *text* with quotation marks.

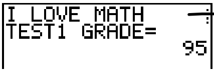
:Disp *valueA*[,*valueB*,*valueC*,**...]**
:Disp "*text***"**[,*valueA*]

Pause after **Disp** halts execution temporarily so that you can examine the screen. To resume execution, press [ENTER]. If a list is too large to display in its entirety, an ellipsis (...) is displayed in the last column, but the list cannot be scrolled.

- If *value* is a variable, the current value stored to the variable is displayed.
- If *value* is an expression, it is evaluated and the result is displayed on the right side of the next line.
- If *value* is text within quotation marks, it is displayed on the left side of the current display line. → is not valid as text.

Write a program named **DISPNOTE** that displays the messages, "**I LOVE MATH**" and "**TEST1 GRADE=95**".

```
PROGRAM:DISPNOTE
:Disp "I LOVE MATH"
:Pause
:Disp "TEST1 GRADE=",95
:Pause
```



Press [ENTER].

DispGraph [PRGM] [▶] 4

DispGraph (display graph) displays the graph of all defined and selected Y_n functions during program execution. If **Pause** is encountered after **DispGraph**, the program halts temporarily so that you can examine the screen. Press [ENTER] to resume execution.

:DispGraph

DispTable [PRGM] [▶] 5

DispTable (display table) displays the table for all defined and selected Y_n functions during program execution. If **Pause** is encountered after **DispTable**, the program halts temporarily so that you can examine the screen. Press [ENTER] to resume execution.

:DispTable

Output([PRGM] [▶] 6

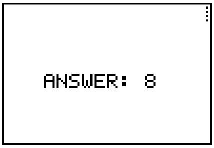
Output(displays *text* or *value* on the Home screen beginning at row (1-8) and column (1-16), overwriting any existing characters. You may want to precede **Output(** with **ClrScreen**.

Expressions are evaluated and values are displayed according to the current mode settings. → is not valid as text.

:Output(*row*,*column*,**"***text***")**
:Output(*row*,*column*,*value***)**

Write a program named **OUTPUT** that writes the contents of *B* to a specific area on the screen.

```
PROGRAM:OUTPUT
:3+5→B
:ClrScreen
:Output(5,4,"ANSWER: ")
:Output(5,12,B)
:Pause
```

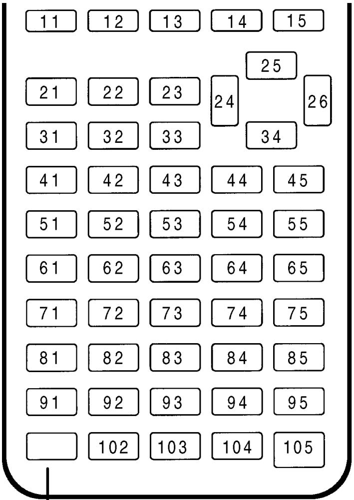


getKey [PRGM] [▸] 7

getKey returns a number corresponding to the last key pressed, according to the following key code diagram. If no key has been pressed, **getKey** returns 0. Use **getKey** inside loops to transfer control, for example, when creating programs that use a key to control the logic flow.

:getKey

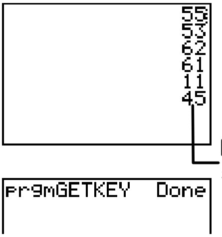
TI-73 Key Code Diagram



[ON] has no number associated with it. It stops program execution.

Write a program named **GETKEY** that displays the key code for the last key pressed, represented as variable *K*. End the program when *K*=45 ([CLEAR]).

```
PROGRAM:GETKEY
:Lbl A
:0→K
:While K=0
:getKey→K
:End
:Disp K
:If K≠45
:Goto A
```



[CLEAR] (45) stops execution.

ClrScreen and ClrTable [PRGM] [▸] 8 and 9

ClrScreen (clear Home screen) clears the Home screen during program execution.

:ClrScreen

ClrTable (clear table) clears the values in the table during program execution.

:ClrTable

GetCalc([PRGM] [▸] 0

GetCalc(gets the contents of *variable* from another TI-73 and stores it to *variable* on the receiving 73. *variable* can be a real number, list element, list name, **Y_n** variable, or picture.

:GetCalc(*variable***)**

You can access **GetCalc(** from the **CATALOG** ([2nd] [CATALOG]) to execute it from the Home screen.

Get(and Send([PRGM] [▸] A and B

Get(gets data from the Calculator-Based Laboratory™ (CBL 2™, CBL™), or Calculator-Based Ranger™ (CBR™) system and stores it to *variable* on the receiving TI-73. *variable* can be a real number, list element, list name, **Y_n** variable, or picture.

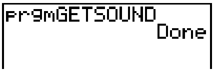
:Get(*variable***)**

Send(sends the contents of *variable* to the CBL 2/CBL or CBR. You cannot use it to send to another TI−73. *variable* can be a real number, list element, list name, **Y_n** variable, or picture. *variable* can be a list of elements.

:Send(*variable***)**

Write a program named **GETSOUND** that gets sound data and time in seconds from a CBL 2/CBL.

```
PROGRAM:GETSOUND
:Send({3,.00025,99,1,0,0,0,0,
  1})
:Get(L1)
:Get(L2)
```



Editing Program Commands

To edit a stored program, select the program name that you want to edit from the PRGM **EDIT** menu. The calculator displays the Program editor and all existing program lines for that program.

The PRGM **EDIT** menu lists in alphabetical order all created programs. From this list, select the program you want to edit. The calculator then displays the Program editor which displays all existing programming commands that make up the selected program.

This menu labels the first 10 items using **1** though **9**, then **0**. All other programs are still included in the list but are not labeled with a number. To select a menu item, press the number associated with it or highlight the item with the cursor keys, and then press ENTER.

PRGM ▶

Inserting, Deleting, and Editing Command Lines

- To insert a new command line anywhere in the program, place the cursor where you want the new characters, press [2nd] [INS], and then press [ENTER]. A colon indicates a new line.
- To insert characters on an existing line, place the cursor where you want the new line, press [2nd] [INS], and then enter the new characters.
- To delete a command line, place the cursor on the line, press [CLEAR] to clear all instructions and expressions on the line, and then press [DEL] to delete the command line, including the colon.
- To move the cursor to the beginning of a command line, press [2nd] [◄]; to move to the end, press [2nd] [►].

Copying and Renaming a Program

You can copy all command lines from one program into a new or existing program.

- To copy into a new program, use the ([PRGM] [►] [►]) menu to create and name the new program. The calculator then automatically displays the Program editor with the program name on the top line.
- To copy into an existing program, use the [PRGM] **EDIT** ([PRGM] [►]) menu and select the existing program name. The calculator then automatically displays the Program editor with the existing program name on the top line.

Then follow these steps:

1. Position the cursor where you want the copy of the program to begin.
2. Press [2nd] [RCL]. **Rcl** is displayed on the bottom line of the Program editor.
3. Press [PRGM] [▸] [▸] to display the [PRGM] **EXEC** menu.
4. Select a name from the menu. **prgm***name* is pasted to the bottom line of the Program editor. You cannot directly enter the subroutine name using the Text editor when using **Rcl**. You must select the name from the [PRGM] **EXEC** menu.)
5. Press [ENTER]. All command lines from the selected program are copied into the new or existing program.

Calling a Program from Another Program

The [PRGM] **EXEC** (execute) menu ([PRGM] [▸] [▸]), accessed only from the Program editor, lets you call any stored program into the current program. The called program then becomes a subroutine in the current program.

The [PRGM] **EXEC** menu lists in alphabetical order all created programs. From this list, select the program that you want to call. The program name is pasted to the cursor location in the Program editor.

This menu labels the first 10 items using **1** though **9**, then **0**. All other programs are still included in the list, but are not labeled with a number. To select a menu item, press the number associated with it or highlight the item with the cursor keys, and then press [ENTER].

You also can enter a program name on a command line by selecting **E:prgm** from the [PRGM] **CTL** menu, and then entering the program name using the Text editor.

[PRGM] [▶] [▶] (from the Program editor only)

When **prgm***name* is encountered during execution, the next command that the program executes is the first command in the subroutine. It returns to the subsequent command in the first program when it encounters either **Return** or the implied **Return** at the end of the second program.

Notes about Calling Programs

- Variables are global.
- *label* used with **Goto** and **Lbl** is local to the program where it is located. *label* in one program is not recognized by another program. You cannot use **Goto** to branch to a *label* in another program.
- **Return** exits a subroutine and returns to the calling program, even if it is encountered within nested loops.

Executing a Program

The [PRGM] **EXEC** (execute) menu lists in alphabetical order all created programs. From this list, select the program that you want to execute. The program name is pasted to the cursor location on the Home screen. Pressing [ENTER] begins executing the program. Pressing [ENTER] after a program is completed returns you to the Home screen.

The [PRGM] **EXEC** menu labels the first 10 items using **1** though **9**, then **0**. All other programs are still included in the list, but are not labeled with a number. To select a menu item, press the number associated with it or highlight the item with the cursor keys, and then press [ENTER].

[PRGM] (except from the Program editor)	

Breaking Out of a Program

To stop program execution, press [ON]. The **ERR:BREAK** menu is displayed.

- To return to the Home screen, select **1:Quit**.
- To go where the interruption occurred, select **2:Goto**.

Debugging a Program

The TI−73 checks for program errors during program execution. It does not check for errors as you enter a program.

If the calculator finds an error during program execution, it stops execution and then displays an error screen.

- To return to the Home screen, press **1:Quit**.
- To go where the error occurred in the program code, select **2:Goto**.

13 Communication Link and the CBL/CBR Application

TI-73 Link Capabilities

The TI−73 comes with a unit-to-unit link cable. With this cable, you can connect to and communicate with another TI−73, a TI−82, a TI−83, the Calculator-Based Laboratory™ (CBL 2™, CBL™), or the Calculator-Based Ranger™ (CBR™). You can communicate with a personal computer using TI™ Connect or TI-GRAPH LINK™ software and a TI-GRAPH LINK cable.

For information about any of these accessories, contact Texas Instruments Customer Support (see Appendix C: Battery/Service and Warranty Information).

To connect the TI−73 to another device using the unit-to-unit cable, use the link port located at the center of the bottom edge of the calculator.

1. Insert either end of the unit-to-unit cable into the TI−73 port very **firmly**.
2. Insert the other end of the cable into the port of the other device.

Linking to Another Calculator

By linking two TI-73's you can transfer all variables and programs to another TI−73 or back up the entire RAM (Random Access Memory) of a TI−73. To transmit from one TI−73 to another, you first must set up one TI−73 to send and the other to receive using the [APPS] **1:Link SEND** and **RECEIVE** menus (see page 255 and 257).

Linking a TI−73 to a TI−82 or TI−83 lets you transfer some types of data between the calculators. Use the [APPS] **1:Link SEND** menu items **9:Vars to TI82** and **0:Vars to TI83** (see page 256).

- You can only transfer numerical list data stored in **L1–L6** (NOT categorical lists) to a TI−82. All fractional elements are converted to decimals.

  If dimension>99 for a TI−73 list that is selected to be sent to a TI−82, the TI−82 truncates the list at the 99th element during transmission.

- You can only transfer numerical list data stored in **L1–L6** or user-named numerical lists to a TI-83 (NOT categorical lists). All fractional elements are converted to decimals.
- From a TI-82 or a TI-83 to a TI-73, you cannot perform a memory backup (but you can send real numbers, real number lists, and picture variables).

Linking to the CBL 2/CBL System or CBR

Connect a CBL 2/CBL or CBR to a TI-73 using one of the unit-to-unit link cables that are included with the calculator, the CBR and the CBL 2/CBL. See the section entitled, "Selecting the CBL/CBR Application" in this chapter.

Linking to a PC or Macintosh™

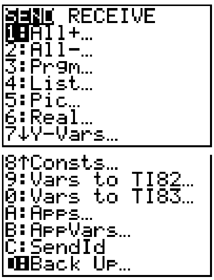
TI-GRAPH LINK™ is an optional application that connects to a TI-73 to enable communications with a personal computer.

The Link SEND Menu APPS 1

You choose the type of data you want to send from the TI-73 to another device from the APPS **1:Link SEND** menu.

To communicate between two calculators, you must set up one calculator to send the data and the other calculator to receive the data. The following section describes how to set up the TI-73 to *send* data. To set up a TI-82 or TI-83, refer to its user manual.

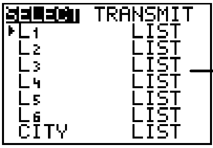
APPS **1**



1:All+…	Displays all RAM items as selected.
2:All–…	Displays all RAM items as deselected.
3:Prgm…	Displays all program names.
4:List…	Displays all list names.
5:Pic…	Displays all picture data types.
6:Real…	Displays all real variables.
7:Y-Vars…	Displays all $\mathbf{Y_n}$ variables.
8:Consts…	Displays all constants.
9:Vars to TI82…	Displays list names $\mathbf{L_1}$-$\mathbf{L_6}$ that are defined as numerical lists, real number variables and picture variables.
0:Vars to TI83…	Displays list names that are defined as numerical lists, real number variables, and picture variables.
A:Apps…	Displays all software applications.
B:AppVars…	Displays all software applications variables.
C:SendId	Sends the Calculator ID number immediately. (You do not need to select **TRANSMIT.**)
D:Back Up…	Selects all RAM for backup to a TI−73.

To select data items to send from the sending unit to another calculator, follow these steps:

1. Press [APPS] to display the **APPLICATIONS** menu.
2. Select **1:Link** to display the **Link SEND** menu.
3. Select the type of data you want to send. The corresponding **SELECT** screen is displayed. Each **SELECT** screen, except the one for **All+**, is displayed initially with no data items selected.
4. Press [▲] and [▼] to move the selection cursor (▶) to an item you want to select or deselect.
5. Press [ENTER] to select or deselect an item. Selected names are marked with a black box (■). To exit a **SELECT** screen without transmitting any items, press [2nd] [QUIT].



SELECT LIST screen includes **L1-L6** and any user-named lists.

6. Repeat steps 4 and 5 to select or deselect additional items.

The Link RECEIVE Menu [APPS] 1 [▶]

You set up the TI-73 to receive data from another device using the [APPS] **1:Link RECEIVE** menu.

To communicate between two calculators, you must set up one calculator to send the data and the other calculator to receive the data. The following section describes how to set up the TI-73 to *receive* data. To set up a TI-82 or TI-83, refer to its user manual.

[APPS] **1** [▶]

1:Receive	Sets unit to receive data transmission.

To set up the TI-73 to receive data, follow these steps:

1. Press [APPS] to display the **APPLICATIONS** menu.
2. Select **1:Link** and press [▶] to display the **Link RECEIVE** menu.
3. Select **1:Receive**. The message **Waiting...** and the busy indicator are displayed. The receiving unit is ready to receive transmitted items.

   To exit the receive mode without receiving items, press [ON], and then select **1:Quit** from the **Error in Xmit** menu.

   When transmission is complete, the unit is still in the receive mode. Press [2nd] [QUIT] to exit the receive mode.

Transmitting Data Items

To transmit data items from a TI-73, follow these steps:

1. Select items to send on the sending unit. Keep the **SELECT** screen displayed on the sending unit (see page 257).
2. Set the receiving unit to receive mode (see page 257).
3. Press [▶] on the TI-73 to display the **TRANSMIT** menu.

4. Confirm that **Waiting...** is displayed on the receiving unit, which indicates it is set to receive.
5. Select **1:Transmit**. The name and type of each data item are displayed line by line on the sending unit as the item is queued for transmission, and then on the receiving unit as each item is accepted.
6. After all selected items have been transmitted, the message **Done** is displayed on both calculators. Press [▲] and [▼] to scroll through the names.

To stop a transmission, press [ON]. The **Error in Xmit** menu is displayed on both units. To leave the error menu, select **1:Quit**.

During transmission, if the receiving unit does not have sufficient memory to receive an item, the **Memory Full** menu is displayed on the receiving unit.

- To skip this item for the current transmission, select **1:Omit**. Transmission resumes with the next item.
- To cancel the transmission and exit transmission mode, select **2:Quit**.

Repeating a Transmission to an Additional TI-73

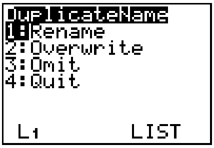
After sending and receiving data between two TI-73s, you can repeat the same transmission without having to reselect data items to send. Use the original sending unit *only* and as many additional TI-73 units as necessary.

Simply repeat the transmission process without selecting or deselecting any new items. **Note**: You cannot repeat the transmission if you selected **All+** or **All-**.

DuplicateName Menu

During transmission, if a variable name is duplicated, the **DuplicateName** menu is displayed on the receiving TI-73.

The duplicate variable name, **L1**, and its type, **LIST**, are displayed.



1:Rename	Prompts you to rename the receiving variable.
2:Overwrite	Overwrites data while receiving the variable.
3:Omit	Skips transmission of sending variable.
4:Quit	Stops transmission.

- When you select **1:Rename**, the **Name=** prompt is displayed, and you can enter another appropriate variable name using the [2nd] [VARS] menu (for example, renaming **Pic1** to **Pic2** where **Pic2** is undefined), or you can enter text using the Text editor ([2nd] [TEXT]) (for example, renaming **L1** to **LABC** where **LABC** is undefined). When renaming lists, do not enter the **L** ([2nd] [STAT] **OPS 9**). The calculator assumes that it is a list name. Press [ENTER] to resume transmission.

  Note: You cannot rename software applications or constants (the **1:Rename** option is excluded from the **DuplicateName** menu).

- When you select **2:Overwrite**, the sending unit's data overwrites the existing data stored on the receiving unit. Transmission resumes.
- When you select **3:Omit**, the sending unit does not send the data in the duplicated variable name. Transmission resumes with the next item.
- When you select **4:Quit**, transmission stops, and the receiving unit exits receive mode.

Transmission Error Conditions

A transmission error (**Error in Xmit**) occurs after one or two seconds if:

- The unit-to-unit cable is not attached to the sending or receiving unit. **Note**: If the cable is attached, push it in firmly and try again.
- The receiving unit is not set to receive transmission.
- You attempt a backup between a TI‑73 and a TI‑82 or TI‑83.
- You attempt a data transfer from a TI‑73 to a TI‑82 with data other than numerical lists, **L1-L6**, or without using menu item **9:Vars to TI82**.

- You attempt a data transfer from a TI−73 to a TI−83 with data other than numerical lists, **L1-L6**, or user-named numerical lists, or without using menu item **0:Vars to TI83**.

Although a transmission error does not occur, these two conditions may prevent successful transmission:

- You try to use **Get(** with a calculator instead of a CBL 2/CBL.
- You try to use **GetCalc(** with a TI−82 or TI−83 instead of a TI−73.

Backing Up Memory

The TI−73 includes two types of memory: RAM (Random Access Memory) and F-ROM (Flash Read Only Memory). RAM includes all lists, programs, variables, and equations. F-ROM includes software applications, such as the **CBL/CBR** application ([APPS] **2**).

To copy (and overwrite) the exact contents of RAM in the sending TI−73 to the memory of the receiving TI−73, follow these steps:

1. Set up the receiving unit in receive mode (see page 257).
2. Then, on the sending unit, select **D:Back Up** from the **Link SEND** menu.
3. Select **1:Transmit** from the **MEMORYBACKUP** menu on the sending unit to begin transmission. Selecting **2:Quit** returns you to the **Link SEND** menu.
4. As a safety check to prevent accidental loss of memory, the message **WARNING–BACKUP** is displayed when the receiving unit receives notice of a backup.

   Select **1:Continue** to begin the backup transmission.

   Select **2:Quit** to prevent the backup and return to the **Link SEND** menu.

When the backup is complete, both the sending calculator and receiving calculator display a **MEMORY BACKUP** confirmation screen. If a transmission error occurs during a backup, the receiving unit's memory is reset.

Upgrading your TI-73 Graph Explorer Software

You can upgrade the software, or operating system, on your TI-73. You do this by transferring this software from a computer to your TI-73 using the TI Connect™ or TI-GRAPH LINK™ software and a TI-GRAPH LINK cable.

Graph Explorer Software Upgrades

You can upgrade two different types of software. These are stored in F-ROM. Therefore, this software is unaffected if you select [2nd] [MEM] **7:Reset 1:All RAM**. These include:

- New versions which enhance the existing software (released free of charge).
- Feature upgrades which modify or add functionality to existing software (available for purchase).

  If you want to download these feature upgrades, which must be purchased from the TI web site, you must provide the unique ID number that identifies your TI-73. To find the ID number, press [2nd] [MEM] **1:About**.

Where to Get Upgrades

For up-to-date information about available upgrades and how to install them, check the TI web site at **education.ti.com** or contact Texas Instruments as described in Appendix C: Battery/Service and Warranty Information.

How to Install Upgrades

To install new Graph Explorer Software, including free or purchased upgrades and applications, you need your TI-73, a computer, TI Connect or TI-GRAPH LINK software, and a TI-GRAPH LINK cable. Extensive directions for installing upgrades are provided on the web site at **education.ti.com**.

1. Transfer the software from the web site to your computer.
2. Transfer the software from the computer to your unit.

Backing Up Your Unit before an Installation

When you install new operating system software, the installation process:

- Deletes all user-defined data items located in RAM.
- Resets all system variables and modes to their original factory settings. This is equivalent to using the **MEMORY RESET** menu to reset all memory.

To retain any existing data items, do either of the following before installing the upgrade:

- Transmit the data items to another TI−73 as described on page 258.
- Use the TI Connect™ or TI-GRAPH LINK™ software and a TI-GRAPH LINK cable to send the data items to a computer.

The APPLICATIONS Menu [APPS]

For the TI−73, you can buy additional software applications, which allow you to customize further your calculator's functionality. The calculator reserves four spaces (placeholders) within ROM memory specifically for applications. The TI−73 comes with the **CBL/CBR** application already listed on the **APPLICATIONS** menu ([APPS] **2**).

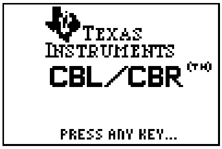
Steps for Running the CBL/CBR Application

Follow these basic steps when using the CBL/CBR application. You may not have to do all of them each time.

Select the CBL/CBR application.

[APPS] **2** [ENTER]

[ENTER]

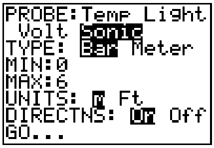


Specify the data collection method.

1, **2**, or **3**

Select options, as applicable.

Highlight options or enter value and press [ENTER].



Collect the data.
Follow directions, if applicable.

Select **Go…** or **START NOW.**

Stop the data collection, if necessary. Repeat these steps or exit the **APPLICATIONS** menu.

[ON] and (TRIGGER) or [ON/HALT]

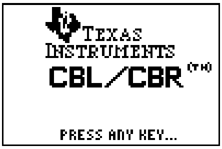
Selecting the CBL/CBR Application

You access the **CBL/CBR** application by pressing [APPS]. In order to use a **CBL/CBR** application, you need a CBL 2/CBL or CBR (as applicable), a TI−73, and a unit-to-unit link cable.

[APPS]

Select **2:CBL/CBR** to set up the TI−73 to use either of the applications. An informational screen first appears. Press any key to continue to the next menu.

[APPS] **2**

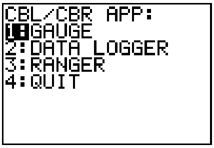


Specifying the Data Collection Method

With a CBL 2/CBL or CBR, you can collect data in one of three ways: **GAUGE** (bar or meter), **DATA LOGGER** (a Temp-Time, Light-Time, Volt-Time, or Sonic-Time graph), or **RANGER**, which runs the **RANGER** program, the built-in CBR data collection program.

CBL 2/CBL and CBR differ in that CBL 2/CBL allows you to collect data using one of four different probes: Temperature, Light, Volt, or Sonic. CBR collects data using only the Sonic probe. You can find more information on CBL 2/CBL and CBR in their user manuals.

[APPS] **2** [ENTER]



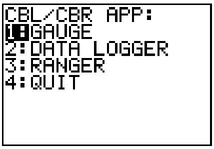
1:GAUGE	Represents results as either a bar or meter. Compatible with CBL 2/CBL or CBR.
2:DATA LOGGER	Represents results as a Temp-Time, Light-Time, Volt-Time, or Sonic-Time graph. Compatible with CBL 2/CBL or CBR.
3:RANGER	Sets up and runs the **RANGER** program and represents results as a Distance-Time, Velocity-Time, or Acceleration-Time graph. Compatible with CBR only.
4:QUIT	Quits the CBL/CBR application.

Specifiying Data Collection Options

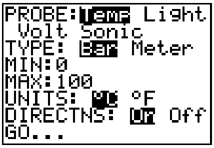
After you select a data collection method, a screen showing the options for that method is displayed. The method you choose, as well as the data collection options you choose for that method, determine whether you use the CBR or the CBL 2/CBL. Refer to the charts in the following sections to find the options for the application you are using.

GAUGE

[APPS] **2** [ENTER]



1



The **GAUGE** data collection method lets you choose one of four different probes: **Temp**, **Light**, **Volt**, or **Sonic**. You can use the CBL 2/CBL with all probes; you can use the CBR only with the **Sonic** probe.

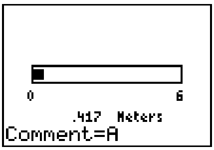
When you select a **PROBE** option, all other options change accordingly. Use ▶ and ◀ to move between the **PROBE** options. To select a probe, highlight the one you want with the cursor keys, and then press [ENTER].

GAUGE Options (Defaults)				
	Temp	**Light**	**Volt**	**Sonic**
TYPE:	**Bar** or **Meter**			
MIN:	**0**	**0**	**-10**	**0**
MAX:	**100**	**1**	**10**	**6**
UNITS:	**°C** or **°F**	**mW/cm²**	**Volt**	**m** or **Ft**
DIRECTNS:	**On** or **Off**			

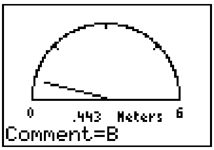
TYPE

The **GAUGE** data collection results are represented according to **TYPE**: **Bar** or **Meter**. Highlight the one you want with the cursor keys, and then press [ENTER].

Bar



Meter



MIN and MAX

MIN and **MAX** refer to the minimum and maximum **UNIT** values for the specified **PROBE**. Defaults are listed in the table on page 267. See the CBL 2/CBL and CBR guidebook for specific **MIN/MAX** ranges. Enter values using the number keys.

UNITS

The results are displayed according to the **UNITS** specified. To specify a unit measurement (**Temp** or **Sonic** probes only), highlight the one you want using the cursor keys, and then press [ENTER].

DIRECTNS (Directions)

If **DIRECTNS=On**, the calculator displays step-by-step directions on the screen, which help you set up and run the data collection. To select **On** or **Off**, highlight the one you want with the cursor keys, and then press [ENTER].

With the **Sonic** data collection probe, if **DIRECTNS=On**, the calculator displays a menu screen before starting the application asking you to select **1:CBL** or **2:CBR.** This ensures that you get the appropriate directions. Press **1** to specify **CBL** or **2** to specify **CBR**.

Data Collection Comments and Results

To label a specific data point, press [ENTER] to pause the data collection. You see a **Comment=** prompt. Enter up to a six-character comment using the Text editor ([2nd] [TEXT]) or number keys. The calculator automatically converts the comment labels and the corresponding results into list elements using the following list names (you cannot rename these lists):

Probe	**Comment Labels (*X*) Stored to:**	**Data Results (*Y*) Stored to:**
Temp	**ʟTCMNT**	**ʟTEMP**
Light	**ʟLCMNT**	**ʟLIGHT**
Volt	**ʟVCMNT**	**ʟVOLT**
Sonic	**ʟDCMNT**	**ʟDIST**

To see all elements in one of these lists, you can insert these lists into the List editor just as you would any other list. Access list names from the [2nd] [STAT] **Ls** menu.

CAUTION: These lists are only temporary placeholders for comment labels and data results for any particular probe. Therefore, every time you collect data and enter comments for one of the four probes, the two lists pertaining to that probe are overwritten with comment labels and data results from the most recently collected data.

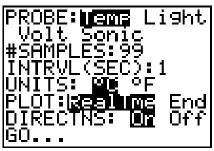
If you want to save comment labels and data results from more than one data collection, copy all list elements that you want to save to a list with a different name.

Also, the **DATA LOGGER** data collection method stores data results to the same list names, overwriting previously-collected data results, even those collected using the **GAUGE** data collection method.

DATA LOGGER

[APPS] **2** [ENTER]

2



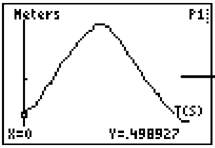
The **DATA LOGGER** data collection method lets you choose one of four different probes: **Temp**, **Light**, **Volt**, or **Sonic**. You can use the CBL 2/CBL with all probes; you can use the CBR only with the **Sonic** probe.

When you select a **PROBE** option, all other options change accordingly. Use [▶] and [◀] to move between the **PROBE** options. To select a probe, highlight the one you want with the cursor keys, and then press [ENTER].

DATA LOGGER Options (Defaults)				
	Temp	**Light**	**Volt**	**Sonic**
#SAMPLES:	**99**	**99**	**99**	**50**
INTRVL (SEC):	**1**	**1**	**1**	**1**
UNITS:	**°C** or **°F**	**mW/cm²**	**Volt**	**m** or **Ft**
PLOT:	**RealTme** or **End**			
DIRECTNS:	**On** or **Off**			
Ymin ([WINDOW]):	**0**			
Ymax ([WINDOW]):	**6**			

The **DATA LOGGER** data collection results are represented as a Temp-Time, Light-Time, Volt-Time, or Distance-Time graph.

Probe-Time Graph



A Distance-Time graph in meters (**Sonic** probe).

#SAMPLES

#SAMPLES refers to how many data samples are collected and then graphed. For example, if **#SAMPLES=99**, data collection stops after the 99th sample is collected. Enter values using the number keys.

INTRVL (SEC)

INTRVL (SEC) specifies the interval in seconds between each data sample that is collected. For example, if you want to collect 99 samples and **INTRVL=1**, it takes 99 seconds to finish data collection. Enter values using the number keys. See the CBR or CBL 2/CBL guidebook for more information about interval limits.

UNITS

The results are displayed according to the **UNITS** specified. To specify a unit measurement (**Temp** or **Sonic** only), highlight the one you want using the cursor keys, and then press [ENTER].

PLOT

You can specify whether you want the calculator to collect realtime (**RealTme**) samples, which means that the calculator graphs data points immediately as they are being collected, or you can wait and show the graph only after all data points have been collected (**End**). Highlight the option you want with the cursor keys, and then press [ENTER].

Ymin and Ymax

To specify **Ymin** and **Ymax** values for the final graph, press [WINDOW] to view the **PLOT WINDOW** screen. Use [▲] and [▼] to move between options. Enter **Ymin** and **Ymax** using the number keys. Press [2nd] [QUIT] to return to the **DATA LOGGER** options screen.

DIRECTNS (Directions)

If **DIRECTNS=On**, the calculator displays step-by-step directions on the screen, which help you set up and run the data collection. To select **On** or **Off**, highlight the one you want with the cursor keys, and then press [ENTER].

With the **Sonic** data collection probe, if **DIRECTNS=On**, the calculator displays a menu screen before starting the application asking you to select **1:CBL** or **2:CBR.** This ensures that you get the appropriate directions. Press **1** to specify **CBL** or **2** to specify **CBR**.

Data Collection Results

The calculator automatically converts all collected data points into list elements using the following list names (you cannot rename the lists):

Probe	Time Values (*X*) stored to:	Data Results (*Y*) Stored to:
Temp	ʟTTEMP	ʟTEMP
Light	ʟTLGHT	ʟLIGHT
Volt	ʟTVOLT	ʟVOLT
Sonic	ʟTDIST	ʟDIST

To see all elements in one of these lists, you can insert these lists into the List editor just as you would any other list. Access list names from the [2nd] [STAT] **Ls** menu.

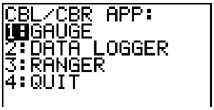
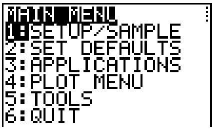
CAUTION: These lists are only temporary placeholders for data results for any particular probe. Therefore, every time you collect data for one of the four probes, the list pertaining to that probe is overwritten with data results from the most recently collected data.

If you want to save data results from more than one data collection, copy all list elements that you want to save to a list with a different name.

Also, the **GAUGE** data collection method stores data results to the same list names, overwriting previously-collected data results, even those collected using the **DATA LOGGER** data collection method.

RANGER

Selecting the **RANGER** data collection method runs the CBR **RANGER** program, a customized program especially for the TI-73 which makes it compatible with the CBR.

[APPS] **2** [ENTER] **3**	
[ENTER]	
	

For detailed information about the **RANGER** program as well as option explanations, see the Getting Started with CBR™ guidebook.

Note: If you execute the **RANGER** data collection method, the program name, **RANGER**, appears in the [PRGM] **EXEC** menu. You can't edit the program, but you can execute it from this menu, just as you would another program. If you delete **RANGER** from the [PRGM] **EXEC** menu ([2nd] [MEM] **4:Delete 6:Prgm**), you can no longer access **RANGER** from this menu; you must select [APPS] **2:CBL/CBR 3:RANGER**.

The **RANGER** data collection method only uses the **Sonic** probe.

Collecting the Data

After you specify all of the options for your data collection method, select the **Go** option from the **GAUGE** or **DATA LOGGER** options screen. If you are using the **RANGER** data collection method, select **1:SETUP/SAMPLE** from the **MAIN MENU** menu, and then **START NOW**.

- If **DIRECTNS=Off**, **GAUGE** and **DATA LOGGER** data collection begin immediately.
- If **DIRECTNS=On**, the calculator displays step-by-step directions.

  If **PROBE=Sonic**, the calculator first displays a menu screen asking you to select **1:CBL** or **2:CBR.** This ensures that you get the appropriate directions. Press **1** to specify **CBL** or **2** to specify **CBR**.
- If you select **START NOW** from the **MAIN MENU** of the **RANGER** data collection method, the calculator displays one directions screen. Press [ENTER] to begin data collection.

Stopping Data Collection

To stop the **GAUGE** data collection method, press [CLEAR] on the TI-73.

The **DATA LOGGER** and **RANGER** data collection methods stop after the specified number of samples have been collected. To stop them before this happens:

1. Press [ON] on the TI-73.
2. Press (TRIGGER) on the CBR, (START/STOP) on the CBL 2, or [ON/HALT] on the CBL.

To exit from the **GAUGE** or **DATA LOGGER** option menus without beginning data collection, press [2nd] [QUIT].

To exit from the **RANGER** option menu without beginning data collection, select **MAIN MENU**. Select **6:QUIT** to return to the **CBL/CBR APP** menu.

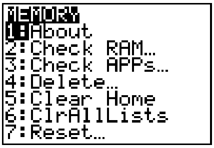
Press **4:QUIT** from the **CBL/CBR APP** menu to return to the TI-73 Home screen.

14 Memory Management

The [2nd] [MEM] MEMORY Menu

At any time, you can check available memory or manage existing memory by selecting items from the [2nd] [MEM] **MEMORY** menu.

[2nd] [MEM]

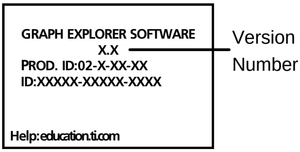


1:About	Displays information about the calculator.
2:Check RAM	Reports memory availability and variable usage.
3:Check APPs	Reports availability of application spaces.
4:Delete	Displays the **DELETE FROM** menu.
5:Clear Home	Clears the Home screen.
6:ClrAllLists	Clears all lists in memory.
7:Reset	Displays the **RESET** menu, which allows you to reset all RAM memory or all defaults.

About [2nd] [MEM] 1

About displays information about your TI-73. To leave the **About** screen and return to the Home screen, press either [2nd] [QUIT] or [CLEAR].

[2nd] [MEM] **1**



Check RAM [2nd] [MEM] 2

Check RAM displays the **MEM FREE** screen. The top line reports the total amount of available memory. The remaining lines report the amount of memory each variable type is using. You can check this screen to see whether you need to delete variables from memory to make room for new data.

To leave the **MEM FREE** screen, press either [2nd] [QUIT] or [CLEAR].

[2nd] [MEM] **2**

```
MEM FREE 24580
 Real        60
 List       673
 Y-Vars     123
 Prgm        49
 Pic          0
```

Check APPs [2nd] [MEM] 3

Check Apps displays the **Spaces Free** screen, which displays application memory available on the calculator. Four spaces in the calculator's memory are reserved for applications. The TI-73 comes with the CBL/CBR application already loaded.

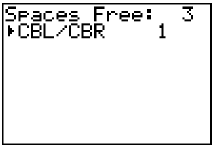
Since applications take up no RAM memory, selecting [2nd] [MEM] **4:Delete 1:All** doesn't delete any applications. Instead, delete an application using [2nd] [MEM] **4:Delete 8: Apps**.

The **Spaces Free** screen displays how many spaces are free as well as the names of all loaded applications and the spaces each one occupies. (Any application can occupy anywhere from one to four spaces depending upon its size.)

To leave the **Spaces Free** screen and return to the Home screen, press either [2nd] [QUIT] or [CLEAR].

For more information on running applications on the TI-73, see Chapter 13: Communication Link and the CBL/CBR Application.

[2nd] [MEM] **3**

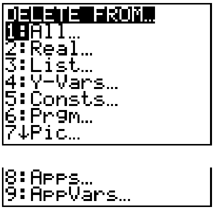


Delete [2nd] [MEM] 4

To increase available RAM memory or application space, you can delete the contents of any type of system variable. You also can delete applications or the application variable, **AppVars**. **Delete** displays a menu of types of variables from which you can select. Selecting a type displays a **DELETE:***type* screen of specific variables to delete.

To leave any **DELETE:***type* screen without deleting anything, press [2nd] [QUIT], which displays the Home screen. Some system variables, such as the last-answer variable **Ans** and the statistical variable **RegEQ** are not listed and cannot be deleted.

[2nd] [MEM] **4**



To delete from the **DELETE FROM** screen:

1. Press [2nd] [MEM] **4:Delete** to display the **DELETE FROM** secondary menu.
2. Select the data type of the variable you want to delete, or select **1:All** for a list of all variables of all types. The **DELETE:** *type* menu is displayed, listing each specific variable of the type you selected and the number of bytes each variable is using.
3. Press [▲] and [▼] to move the selection cursor (▶) next to the variable you want to delete, and then press [ENTER]. The variable is deleted from memory. Repeat, as necessary.

List [2nd] [MEM] 4 3

In addition to deleting lists from the **DELETE:List** menu, you also can delete **IDList**. The **IDList** stores any additional IDs that have been collected from other calculators (using the [APPS] **1:Link 1:Receive** feature). Therefore, at any time you can delete **IDList**, just as you would delete any other variable.

The [2nd] [MEM] **2:Check RAM MEM FREE** screen adds all statistical list and **IDList** memory bytes together and displays the total after the **List**.

Apps and AppVars [2nd] [MEM] 4 8 and 9

Apps allows you to delete individual applications that have been stored on the calculator. Individual applications are not deleted when you select [2nd] [MEM] **4:Delete 1:All** because they are stored in ROM memory (as opposed to RAM).

AppVars is a variable holder used to store variables created by independent applications, but which are not recognized by the TI-73. For example, if you create a matrix with an application and save it to the calculator's memory, the calculator stores it in **AppVars** since matrices are not recognized by the TI-73.

In addition, you cannot edit or change variables in **AppVars** unless you do so through the application which created them.

For more information on running applications, see Chapter 13: Link Communication and the CBL/CBR Application.

Clear Home [2nd] [MEM] 5

Clear Home not only clears the Home screen (like **ClrScreen**) but also clears all previous entries stored in [2nd] [ENTRY] (unlike **ClrScreen**). In addition, all previous entries displayed on the Home screen are erased. To cancel **Clear Home** without clearing, press [CLEAR].

Note: **Clear Home** is different from the **ClrScreen** programming command found under the [PRGM] **I/O** menu.

You can execute **Clear Home** from either the Home screen or the Program editor. If you select **Clear Home** from the Program editor, it is inserted at the cursor location. The Home screen and all entries are cleared when the program is executed.

Clear Home takes no additional arguments.

To clear the Home screen and all entries:

1. Press [2nd] [QUIT] to display the Home screen.
2. Press [2nd] [MEM] **5** to paste the instruction to the Home screen.
3. Press [ENTER] to execute the instruction.

ClrAllLists [2nd] [MEM] 6

ClrAllLists sets to 0 the dimension of each list in memory. To cancel **ClrAllLists**, press [CLEAR]. **ClrAllLists** does not delete list names from memory, from the [2nd] [STAT] **Ls** menu, or from the List editor.

You can execute **ClrAllLists** from either the Home screen or the Program editor. If you select **ClrAllLists** from within the Program editor, it is inserted at the cursor location. The lists are cleared when the program is executed.

ClrAlllLists takes no additional arguments.

To clear all elements from all lists:

1. Press [2nd] [QUIT] to display the Home screen.
2. Press [2nd] [MEM] **6** to paste the instruction to the Home screen.
3. Press [ENTER] to execute the instruction.

Reset [2nd] [MEM] 7

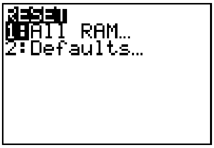
The **RESET** secondary menu gives you the options of resetting all RAM memory (including default settings) or only resetting the default settings while preserving other data stored in memory, such as programs and $\mathbf{Y_n}$ functions. To leave without resetting and to return to the Home screen, press either [2nd] [QUIT] or [CLEAR].

Resetting All Memory [2nd] [MEM] 7 1

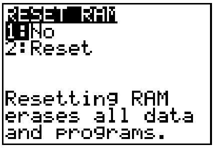
Resetting all RAM memory on the TI-73 restores the memory to the factory settings. It deletes all non-system variables and all programs. It resets all system variables to the default settings.

Before you reset *ALL* memory, consider deleting only selected data using [2nd] [MEM] **4:Delete**.

[2nd] [MEM] **7**



[2nd] [MEM] **7** **1**



From the **RESET RAM** screen:

- Select **1:No** to cancel memory reset and return to the Home screen.
- Select **2:Reset** to erase from memory all data and programs. All factory defaults are restored. **Mem cleared** is displayed on the Home screen.

When you clear memory, the contrast sometimes changes. If the screen is faded or blank, adjust the contrast. Press [2nd] [▲] to increase the contrast or [2nd] [▼] to decrease the contrast.

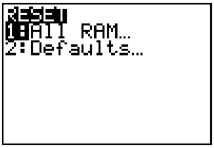
Resetting Defaults [2nd] [MEM] 7 2

When you reset defaults on the TI-73, all defaults are restored to the factory settings. Stored data and programs are not changed.

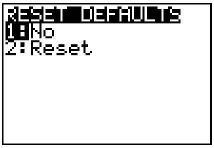
Some examples of the TI-73 defaults that are restored by resetting the defaults are:

- Mode settings ([MODE]).
- $\mathbf{Y_n}$ functions that are deselected ([Y=]).
- **WINDOW** variables ([WINDOW]).
- Stat plots that are deselected ([2nd] [PLOT]).
- **WINDOW** format settings ([2nd] [FORMAT]).
- **rand** seed value ([MATH] **PRB 1:rand**).

[2nd] [MEM] **7**



[2nd] [MEM] **7** **2**



From the **RESET DEFAULTS** screen:

- Select **1:No** to cancel defaults reset and return to the Home screen.
- Select **2:Reset** to reset all defaults. Default settings are restored. **Defaults set** is displayed on the Home screen.

A Function and Instruction Reference

All the operations in this section are included in the **CATALOG** ([2nd] [CATALOG]). Non-alphabetic operations (such as +, !, and >) are listed at the end of the **CATALOG**.

You always can use the **CATALOG** to select an operation and insert it next to the cursor on the Home screen or to a command line in the Program editor. You also can use the specific keystrokes, menus, or screens listed here below the function or instruction's name.

† Indicates menus or screens that insert the operation's name only if you are in the Program editor. In most cases (like mode or window format settings), you can use these menus or screens from the Home screen to perform the operation interactively; the name is not inserted at the cursor.

‡ Indicates menus or screens that are valid only from the Program editor's main menu. From the Home screen, you cannot use these menus or screens to select an operation.

[ ] Indicate optional arguments. If you specify an optional argument, do not enter the brackets.

A␣b/c

† [MODE]

Selects the **A␣b/c** Display Format mode setting. Displays results as mixed numbers, if applicable.

$\frac{4}{5}+\frac{8}{5}$ [ENTER] $2\frac{2}{5}$

►Ab/c↔d/e

[A b/c ↔ d/e]

Converts a simple fraction to a mixed number or a mixed number to a simple fraction.

$3\frac{1}{3}$ ►Ab/c↔d/e [ENTER] $\frac{10}{3}$

$\frac{10}{3}$ ►Ab/c↔d/e [ENTER] $3\frac{1}{3}$

abs(*value***)**

MATH **NUM**

Returns the absolute value of a real number, expression, or each element of a list.

abs(⁻35) ENTER 35

conditionA **and** *conditionB*

2nd [TEXT]

Logic (boolean) operator; returns **1** if both *conditionA* and *conditionB* are true (non-zero). Returns **0** if either *conditionA* or *conditionB* is false (zero). *conditionA* and *conditionB* can be real numbers, expressions, or lists.

If both *conditions* are lists, they must have the same number of elements. If one *condition* is a list and the other a non-list, the non-list is compared with each element of the list, and a list is returned.

Test operations are frequently used in programs.

```
PROGRAM:AND
:1→A
:2→B
:A>0 and B<0
```

Ans

2nd [ANS]

Returns the last answer calculated.

1.7*4.2 ENTER 7.14

147/ Ans ENTER 20.58823529

augment(*list1*,*list2***)**

2nd [STAT] **OPS**

Combines the elements of two lists, *list1* and *list2*, to create a new list.

augment({1,⁻3,2},{5,4}) ENTER {1 ⁻3 2 5 4}

Autosimp

† MODE

Selects the **Autosimp** Simplification mode setting. Automatically simplifies fractional results.

$\frac{1}{9}+\frac{5}{9}$ ENTER $\frac{2}{3}$

AxesOff
AxesOn

† 2nd [FORMAT]

WINDOW format settings; turns off or on the graph axes.

b/c

† MODE

Selects the **b/c** Display Format mode setting. Displays results as simple fractions, if applicable.

$\frac{3}{4}+\frac{2}{4}$ ENTER $\frac{5}{4}$

BarPlot

See **Plot1**: Bar Graph

BoxPlot
See **Plot1**: Box Plot

Circle(X,Y,*radius***)**

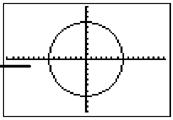
[DRAW]

Draws a circle with center (X,Y) and *radius*, a real number.

ClrDraw[ENTER] Done

Circle(0,0,7)[ENTER]

WINDOW values are set with **ZSquare**.



Clear Home

[2nd] [MEM]

Clears the Home screen (like **ClrScreen**), and also clears all entries stored in [2nd] [ENTRY] and erases all entries on the History screen.

ClrAllLists

[2nd] [MEM]

Sets the dimension of all lists in memory to 0.

ClrDraw

[DRAW]

Clears all drawn elements from the graph screen.

ClrList *list1*[,*list2*,*list3*,...]

[2nd] [STAT] **O P S**

Clears all items in at least one specified list.

ClrList L1,ʟLIST[ENTER]

ClrScreen

‡[PRGM] **I / O**

Programming command; clears the Home screen during program execution.

ClrTable

†[PRGM] **I / O** –or– [2nd] [CATALOG]

Clears the values in the table during program execution if **Indpnt:Ask** is set.

coin(*tosses***)**

[MATH] **P R B**

Returns a random list of 0s and 1s that represents heads and tails for one or more coin *tosses*. *tosses* is a positive whole number.

coin(5)[ENTER] {1 1 0 1 0}

CoordOff
CoordOn

† [2nd] [FORMAT]

WINDOW format settings; turns off or on cursor coordinates so that they are not displayed at the bottom of the graph.

---

cos(*value***)**

[2nd] [TRIG] **TRIG**

Returns the cosine of a real number, expression, or each element of a list. Results are determined by Angle mode setting (**Degree** or **Radian**).

In Degree mode:
cos(45) [ENTER] .7071067812
cos ({0,60,90}) [ENTER]
{1 .5 0}

In Radian mode:
cos(π/2) [ENTER] 0
cos ({0,π/2,π}) [ENTER]
{1 0 ⁻1}

---

cos⁻¹(*value***)**

[2nd] [TRIG] **TRIG**

Returns the arccosine of a real number, expression, or each element of a list. ⁻1≤*value* ≤1. Results are determined by Angle mode setting (**Degree** or **Radian**).

In Degree mode:
cos⁻¹(1) [ENTER] 0
cos⁻¹ ({1,0}) [ENTER]
{0 90}

In Radian mode:
cos⁻¹(.5) [ENTER] 1.047197551

cos⁻¹ ({0,.5}) [ENTER]
{1.570796327 1. ...}

Degree

† [MODE]

Selects the **Degree** Angle mode setting. Interprets angles as degrees.

In Degree mode:
sin(90) [ENTER] 1
sin(π/2) [ENTER] .0274121336

DelVar *variable*

† [PRGM] **CTL** –or– [2nd] [CATALOG]

Deletes the contents of *variable* from memory. You cannot delete a program or a system variable.

```
PROGRAM:DELVAR
:{1,2}→L1
:Disp L1²
:Pause
:DelVar L1
:Disp L1
:Pause
```

ERROR 14: UNDEFINED

DependAsk

† [2nd] [TBLSET]

Selects the **Depend: Ask TABLE SETUP** format setting. The user must highlight a dependent variable (*Y*) space with the cursor, and then press [ENTER] to view the value.

DependAuto

† [2nd] [TBLSET]

Selects the **Depend: Auto TABLE SETUP** format setting. Table automatically displays dependent variable (*Y*) values.

DiagnosticOff
DiagnosticOn

[2nd] [CATALOG]

Settings which tell the calculator not to display (**DiagnosticOff**) or to display (**DiagnosticOn**) r and r^2 (coefficient of determination) with **LinReg** and **ExpReg** regression model ([2nd] [STAT] **CALC**) results or R^2 for QuadReg regression model results.

dice(*rolls*[,#*dice*]**)**

[MATH] **PRB**

Returns a random list of numbers (between 1 and 6) that represent dice rolls. **dice(** takes one optional argument, #*dice*, a positive whole number>1. If #*dice* is specified, each list element is the total sum of one roll's results.

dice(5)[ENTER] {5 1 3 6 2}

dice(5,2)[ENTER] {11 5 7 2 10}

dim(*list***)**
newDimension#[STO▸]**dim(***list***)**
dimension#[STO▸]**dim(***newList***)**

[2nd] [STAT] **OPS**

Returns the dimension (number of elements) of a defined list, changes the dimension of an existing list, or creates a new list with a specified number of elements. New elements are set to 0.

{1,2,3}→L1 [ENTER] {1 2 3}
dim(L1) [ENTER] 3

5→dim(L1) [ENTER] 5
L1 [ENTER] {1 2 3 0 0}

4→dim(ʟNEW) [ENTER] 4
ʟNEW [ENTER] {0 0 0 0}

Disp [*valueA*,*valueB*,…]

‡[PRGM] **I/O**

Programming command (display); displays one or more *values*, as specified in an argument. To display text, surround the *value* with quotation marks. To see the output, follow **Disp** with a **Pause** instruction.

```
PROGRAM:DISP
:10→X
:Disp X
:Disp X³+3X–6
:Pause
```

```
PROGRAM:DISPTEXT
:Disp "MATH IS FUN!"
:Pause
```

DispGraph

‡[PRGM] **I/O**

Programming command (display graph); displays the graph for all defined and selected Y_n functions during program execution.

```
PROGRAM:GRAPH
:"2X+5"→Y₁
:DispGraph
```

DispTable ‡[PRGM] **I/O** Programming command (display table); displays the table for all defined and selected $\mathbf{Y_n}$ functions during program execution.	PROGRAM:TABLE :"2X+5"→Y_1 :IndpntAuto :DependAuto :DispTable
angle▶**DMS** [2nd] [TRIG] **ANGLE** Converts an angle to DMS (degrees° minutes' seconds") notation. Results are determined by the Angle mode setting (**Radian** or **Degree**).	In Degree or Radian mode: 50°▶DMS [ENTER] 50°0'0" In Radian mode: 50 ▶DMS [ENTER] 2864°47'20.312"
:**DS<(***variable*,*value***)** :*command1* (if answer ≥ *value*) :*command2* ‡[PRGM] **CTL** Programming command (decrement and skip if less than); subtracts 1 from *variable*. If the answer is < *value*, then *command1* is skipped; if the answer is ≥ *value*, then *command1* is executed. *command2* is always executed.	PROGRAM:DS :9→A :Lbl S :Disp A :DS<(A,5) :Goto S :Disp "A IS NOW <5" :Pause
e^(*x***)** [MATH] **LOG** Raises e to a power of *x*, where *x* is a real number, an expression that results in an real number, or a list of real numbers. e equals 2.71828182846.	e^(2.5) [ENTER] 12.18249396
value **E** *exponent* [2nd] [EE] Enters a number in scientific notation. The display of the result depends upon the Numeric mode setting (**Normal** or **Sci**). *value* can be a real number or list.	In Normal Numeric mode: 12.3456789**E**5 [ENTER] 1234567.89 (1.78/2.34)**E**2 [ENTER] 76.06837607 {6.34,854.6}**E**3 [ENTER] {6340 854600}
Else See **If**:**Then**:**Else**:**End**	
End ‡[PRGM] **CTL** Programming command; you must include an **End** instruction at the end of each **For(**, **While**, or **Repeat** loop. Also, you must enter an **End** instruction at the end of each **If–Then** group and each **If–Then–Else** group.	

ExpReg [*XList*,*YList*,*freq*,Y_n]

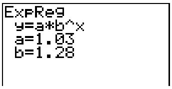
[2nd] [STAT] **CALC**

Fits the equation ($y=ab^x$) to *XList* and *YList* with frequency list, *freq*, and stores the regression equation to Y_n. *XList*, *YList*, and *freq* (if specified) must have the same number of elements.

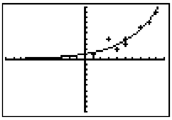
freq is the frequency of occurrence for each corresponding data point in *XList*. If *freq* is omitted, all values are used once.

Defaults for *XList* and *YList* are **L1** and **L2**.

{1,3,4,5,5,7,8,9}→L3 [ENTER] Done
{1,4,2,3,4,6,7,9}→L4 [ENTER] Done
Set Decimal mode to 2.
ExpReg L3,L4,Y_1 [ENTER]



Select ZStandard.



ExprOff
ExprOn

† [2nd] [FORMAT]

WINDOW format settings; turns off or on the expression display in the top left corner while tracing a graph.

►F↔D

[F◂▸D]

Converts a fraction to its decimal equivalent or changes a decimal to its fractional equivalent, if possible.

$\frac{3}{4}$ ►F↔D [ENTER] .75

.75►F↔D [ENTER] $\frac{3}{4}$

Fill(*number*,*list***)**

[2nd] [CATALOG]

Replaces each element in existing *list* with specified real number, *number*.

{3,4,5}→L1 [ENTER] {3 4 5}
Fill(8,L1) [ENTER] Done
L1 [ENTER] {8 8 8}

Fix #*ofplaces*

† [MODE]

Sets fixed Decimal mode setting for # of decimal places. #*ofplaces* must be an integer between 0 and 9. It can be an expression which equals an appropriate integer.

Fix 3 [ENTER] Done
π [ENTER] 3.142

Float

† [MODE]

Selects the **Float** Decimal Notation mode setting. Displays a decimal with a maximum of 10 digits, including the sign and decimal point.

Float [ENTER] Done
π [ENTER] 3.141592654

FnOff [**1,2,3,4**] **FnOn** [**1,2,3,4**] [2nd] [VARS] **2:Y-Vars** Turns off (deselects) or on (selects) all $\mathbf{Y_n}$ functions or specified $\mathbf{Y_n}$ functions ($\mathbf{Y_1}$, $\mathbf{Y_2}$, $\mathbf{Y_3}$, or $\mathbf{Y_4}$).	FnOff 1,3 [ENTER] Done FnOn 2 [ENTER] Done
:**For(***variable*,*begin*,*end*,[*increment*]**)** :*block* (while *variable* ≤ *end*) :**End** :*command* ‡ [PRGM] **CTL** Programming command; executes commands in *block* through *end*, increasing *variable* from *begin* by *increment* until *variable*>*end*.	PROGRAM:FOR :For(A,0,8,2) :Disp A^2 :Pause :End
fPart(*value***)** [MATH] **NUM** Returns the fractional part of a real number, expression, or each element in a list.	fPart(23.45)[ENTER] .45 fPart(⁻17.26∗8) [ENTER] -.08 fPart({1.2,3.4,5.6}) [ENTER] {.2 .4 .6} fPart($1\frac{1}{2}$) $\frac{1}{2}$
gcd(*valueA*,*valueB***)** [MATH] **MATH** Returns the greatest common divisor (the largest number that can divide into the two *values* evenly), of two positive whole numbers or lists of positive whole numbers.	gcd(27,36) [ENTER] 9 $\frac{27}{36}$▸Simp 9 [ENTER] $\frac{3}{4}$
Get(*variable***)** † [PRGM] **I/O** –or– [2nd] [CATALOG] Gets data from a CBR or CBL 2/CBL System and stores it in *variable*.	PROGRAM:GETSOUND :Send ({3,.00025,99,1,0,0,0,0, 13}) :Get(L1) :Get(L2)
GetCalc(*variable***)** † [PRGM] **I/O** **–or–** [2nd] [CATALOG] Gets contents of *variable* from another TI-73 and stores it to *variable* on the receiving TI-73.	PROGRAM:GETCALC :GetCalc(L1) :GetCalc(Y_1) :GetCalc(Pic1)
getKey ‡ [PRGM] **I/O** Programming command; returns the key code for the current keystroke. See Chapter 12: Programming for the Key Code Diagram displayed with the **GetKey** explanation.	PROGRAM:GETKEY :Lbl A :0→K :While K=0 :getKey→K :End :Disp K :If K≠45 :Goto A

Goto *label*

‡[PRGM] **CTL**

Programming command; transfers program control to the *label* specified by preceding *label* instruction.

PROGRAM:GOTO
:Lbl 99
:Input A
:If A≥100
:Stop
:Disp A^2
:Pause
:Goto 99

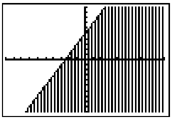
GraphStyle(Y_n,*type***)**

†[PRGM] **CTL** –or– [2nd] [CATALOG]

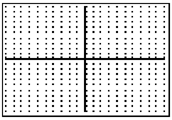
Defines one of seven graphstyle *types* for Y_n. Y_n=1, 2, 3, or 4 (for $\mathbf{Y_1}$, $\mathbf{Y_2}$, $\mathbf{Y_3}$, $\mathbf{Y_4}$). The *type* icons described below are located to the left of $\mathbf{Y_n}$ in the Y= editor.

1 = (line)
2 = (thick)
3 = (above)
4 = (below)
5 = (path)
6 = (animate)
7 = (dot)

PROGRAM:STYLE
:"2X+5"→Y_1
:GraphStyle(1,4)
:ZStandard



GridOff
GridOn



†[2nd] [FORMAT]

WINDOW format settings; turn off or on grid lines that correspond with **Xscl** and **Yscl** while graphing.

Histogram

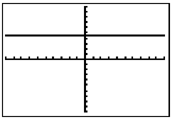
See **Plot1**: Histogram

Horizontal *y*

[DRAW] **DRAW**

Draws a horizontal line on the current graph at Y=*y*. *y* can be an expression but not a list.

Horiz 4.5[ENTER]



:**If** *condition*
:*command1* (if true)
:*command2*

‡[PRGM] **CTL**

Programming command; if *condition* is true (non-zero), then *command1* is executed. If *condition* is false (zero), then *command1* is skipped.

PROGRAM:IF
:0→A
:Lbl Z
:A+1→A
:Disp "A IS",A
:Pause
:If A≥2
:Stop
:Goto Z

If–Then

:**If** *condition*
:**Then**
:*block* (if true)
:**End**

‡[PRGM] **CTL**

Programming commands; if *condition* is true (non-zero), then *block* is executed. If *condition* is false (zero), then *block* is skipped.

```
PROGRAM:THEN
:1→X:10→Y
:If X<10
:Then
:2X+3→X
:2Y–3→Y
:End
:Disp (X,Y)
:Pause
```

If–Then–Else

:**If** *condition*
:**Then**
:*block1* (if true)
:**Else**
:*block2* (if false)
:**End**

‡[PRGM] **CTL**

Programming commands; if *condition* is true (non-zero), then *block1* is executed. If *condition* is false (zero), then *block2* is executed.

```
PROGRAM:ELSE
:Input "X=",X
:If X<0
:Then
:X²→Y
:Else
:X→Y
:End
:Disp X,Y
:Pause
```

IndpntAsk

†[2nd] [TBLSET]

Selects the **Indpnt: Ask TABLE SETUP** format setting. Table asks the user for independent variable (*X*) values.

IndpntAuto

†[2nd] [TBLSET]

Selects the **Indpnt: Auto TABLE SETUP** format setting. Table automatically displays independent variable (*X*) values.

Function	Example
Input **Input** [*variable*] **Input** ["*text*",*variable*] ‡ [PRGM] **I/O** Programming command; **Input**, with no arguments, displays the current graph. Otherwise, **Input** accepts input and stores it to *variable* (prompted by a **?**, unless otherwise defined). *text* designates a specific text prompt (≤16 characters), if desired, and must be enclosed in quotation marks.	PROGRAM:INPUTVAR :Input "Y_1=",Y_1 :Input "A=",A :Input "ʟDATA=",ʟDATA :Disp "Y_1(A)=",Y_1(A) :Pause :Disp :"Y_1(ʟDATA)=",Y_1(ʟDATA) :Pause :PROGRAM:GRPHINPT :FnOff :PlotsOff :ZStandard :Input :Line (0,0,8,8) :Pause
int(*value***)** [2nd] [CATALOG] Returns the largest integer ≤ *value*, where *value* can be a real number, expression, or list. For a negative non-integer, **int** returns the integer that is one less than the integer part of the number. To return the exact integer part, use **iPart** instead.	int (23.45)[ENTER] 23 int (⁻23.45) [ENTER] ⁻24
posintegerA **Int/***posintegerB* [2nd] [INT÷] Divides two positive integers and displays the quotient and the remainder, **r**.	9 Int/ 2 [ENTER] 4r1
iPart(*value***)** [MATH] **NUM** Returns the integer part of a real number, expression, or each element of a list.	iPart (23.45)[ENTER] 23 iPart (⁻17.26*8) [ENTER] ⁻138 iPart ({1.2,3.4,5.6})[ENTER] {1 3 5} iPart($1\frac{1}{2}$) 1
:**IS>(***variable*,*value***)** :*command1* (if *answer* is ≤ *value*) :*command2* ‡ [PRGM] **CTL** Programming command (increment and skip if greater than); adds *variable* by 1. If the answer is > *value*, then *command1* is skipped; if the answer is ≤ *value*, then *command1* is executed. *command2* is always executed.	:PROGRAM:IS :0→A :Lbl S :Disp A :IS>(A,5) :Goto S :Disp "A IS NOW >5" :Pause

ʟ*listname*

[2nd] [STAT] **OPS**

List signifier; precedes all user-created names when displayed outside of the List editor.

{1,2,3}→ʟABC [ENTER]	{1 2 3}
ʟABC [ENTER]	{1 2 3}

LabelOff
LabelOn

† [2nd] [FORMAT]

WINDOW format settings; turns off or on axes labels.

Lbl *label*

‡ [PRGM] **CTL**

Programming command; gives a name (*label*) to a particular location in a program. *label* can be one or two text characters.

```
PROGRAM:LBL
:Lbl 99
:Input A
:If A≥100
:Stop
:Disp A²
:Pause
:Goto 99
```

lcm(*valueA*,*valueB***)**

[MATH] **MATH**

Returns the least common multiple (the smallest number that the two *values* can divide into evenly) of two positive whole numbers or lists of positive whole numbers.

If both arguments are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is paired with each element of the list, and a list is returned.

lcm(10,6) [ENTER]	30

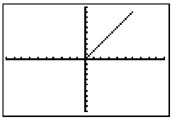
Line(X_1,Y_1,X_2,Y_2[,**0**]**)**

[DRAW] **DRAW**

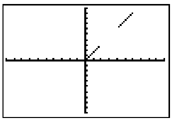
Draws a line from point (X_1,Y_1) to (X_2,Y_2).

Including the argument, **0**, after the X and Y coordinates erases a line from (X_1,Y_1) to (X_2,Y_2).

Select ZStandard and return to the Home screen.
Line(0,0,6,9) [ENTER]



Return to the Home screen.
Line(2,3,4,6,0) [ENTER]



LinReg(ax+b) [*XList*,*YList*,*freq*,Y_n]

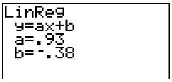
[2nd] [STAT] **CALC**

Fits the linear equation (y=ax+b) to *XList* and *YList* with frequency list, *freq*, and stores the regression equation to Y_n. *XList*, *YList*, and *freq* (if specified) must have the same number of elements.

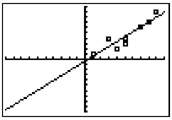
freq is the frequency of occurrence for each corresponding data point in *XList*. If *freq* is omitted, all values are used once.

Defaults for *XList* and *YList* are **L1** and **L2**.

Decimal mode set to 2:
{1,3,4,5,5,7,8,9}→L3 [ENTER] Done
{1,4,2,3,4,6,7,9}→L4 [ENTER] Done
LinReg(ax+b) L3,L4,Y_1 [ENTER]



Select ZStandard.



ΔList(*list***)**

[2nd] [STAT] **OPS**

Returns a list of the differences between consecutive elements in a list.

{4.5,4.6,6,7.5}→L2 [ENTER] {4.5 4.6 6 7.5}

ΔList(L2) [ENTER] {.1 1.4 1.5}

ln(*value***)**
ln(*list***)**

[MATH] **LOG**

Returns the natural logarithm of a positive real number, an expression that results in a positive real number, or a list of positive real numbers.

ln(2) [ENTER] .6931471806

ln(36.4/3) [ENTER] 2.495956486

log(*value***)**
log(*list***)**

[MATH] **LOG**

Returns the base 10 logarithm of a positive real number, an expression (that results in a positive real number), or a list of positive real numbers.

log(2) [ENTER] .3010299957

log(36.4/3) [ENTER] 1.083980129

Mansimp

† [MODE]

Selects the **Mansimp** Simplification mode setting. Requires user to simplify fractional results manually using the [SIMP] key.

$\frac{1}{4}+\frac{1}{4}$ [ENTER] $\downarrow\frac{2}{4}$

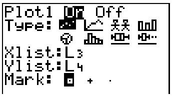
$\frac{2}{4}$▶Simp [ENTER] Fac=2 $\frac{1}{2}$

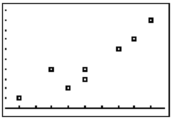
Manual-Fit [Y_n]

[2nd] [STAT] **CALC**

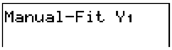
Allows you to fit manually a line to plotted data. The regression equation is stored to Y_n, if specified.

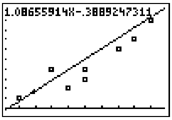
{1,3,4,5,5,7,8,9}→L3 [ENTER] Done
{1,4,2,3,4,6,7,9}→L4 [ENTER] Done
Set up Plot1 as a scatter plot and graph using ZStat:





Return to the Home screen and select Manual-Fit.





Select beginning and ending points of line by moving the cursor and then pressing [ENTER].

max(*valueA*,*valueB***)**

[MATH] **NUM** – or – [2nd] [STAT] **MATH**

Returns the larger of two *values* or the largest element in one *list*. *value* can be a real number, expression or a list.

If both *values* are lists, they must have the same number of elements. If one *value* is a list and the other a non-list, the non-list is paired with each element of the list, and a list is returned.

max(2.3,1.4)[ENTER] 2.3

max({1,3,6})[ENTER] 6

max({1,10},{2,9})[ENTER] {2 10}

max$\left(\frac{2}{3},\frac{3}{4}\right)$ $\frac{3}{4}$

mean(*list*[,*freq*]**)**

[2nd] [STAT] **MATH**

Returns the mean (mathematical average) of *list*. If a second list, *freq*, is specified, it is interpreted as the frequency of the elements in the first list. *list* and *freq* must have the same number of elements.

mean({1,2,3,4})[ENTER] 2.5

mean({1,2,3,4},{4,5,4,6})[ENTER]
2.631578947

median(*list*[,*freq*]**)**

[2nd] [STAT] **MATH**

Returns the median (the middle element) of *list*. If a second list, *freq*, is specified, it is interpreted as the frequency of the elements in the first list. *list* and *freq* must have the same number of elements.

median({1,2,3,4})[ENTER] 2.5

median({1,2,6},{4,5,4})[ENTER] 2

Med-Med [*XList*,*YList*,*freq*,Y_n]

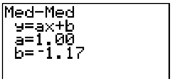
[2nd] [STAT] **CALC**

Fits a median-median model equation, **y=ax+b**, to *XList* and *YList* with frequency list, *freq*, and stores the regression equation to Y_n. *XList*, *YList*, and *freq* (if specified) must have the same number of elements.

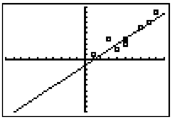
freq is the frequency of occurrence for each corresponding data point in *XList*. If *freq* is omitted, all values are used once.

Defaults for *XList* and *YList* are **L1** and **L2**.

Decimal mode set to 2:
{1,3,4,5,5,7,8,9}→L3 [ENTER] Done
{1,4,2,3,4,6,7,9}→L4 [ENTER] Done
Med-Med L3,L4,Y_1 [ENTER]



[ZOOM] 6



Menu("*title*","*item1*",*label1*[,"*item2*",*label2*…]**)**

‡[PRGM] **CTL**

Programming command; generates a menu of up to seven *items* during program execution. When you select a menu item, the calculator branches to the *label* corresponding with that *item*.

:PROGRAM:FRIENDS
:Menu("FRIENDS","JULIE",A,
"XIAODAN",B,"LETICIA",C,
"ROBERTO",D,"DOUGLAS",E,
"ANSIK",F,"DETER",G)

min(*valueA*,*valueB***)**
min(*list***)**

[MATH] **NUM** – or – [2nd] [STAT] **MATH**

min((minimum) returns the smaller of two *values* or the smallest element in one *list*. *value* can be a real number, expression or a list.

If both *values* are lists, they must have the same number of elements. If one *value* is a list and the other a non-list, the non-list is paired with each element of the list, and a list is returned.

min(3,⁻5) [ENTER] ⁻5
min(⁻5.2, ⁻5.3) [ENTER] ⁻5.3
min(5,2+2)[ENTER] 4

min($\frac{2}{3}$,$\frac{3}{4}$) $\frac{2}{3}$

ModBoxPlot

See **Plot1**: Modified Box Plot

mode(*list*[,*freq*]**)**

[2nd] [STAT] **MATH**

Returns the mode (element which occurs most frequently) of *list*. If a second list, *freq*, is specified, it is interpreted as the frequency of the elements in the first list. *list* and *freq* must have the same number of elements.

mode({1,2,4,3,1,8})[ENTER] {1}

MultiConst

† [2nd] [SET]

Selects the **Multiple** mode (affects the **Set Constant** editor). Allows the user to access all defined constants (as opposed to only one).

items **nCr** *number*

[MATH] **PRB**

Returns the number of combinations of n *items* taken r *number* at a time. The order in which you select the items DOES NOT matter. *items* and *number* can be non-negative integers or lists.

If both arguments are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is paired with each element in the list, and list of combinations is returned.

5 nCr 2 [ENTER] 10

5 nCr {2,4,6,8} [ENTER] {10 5 0 0}

Normal

† [MODE]

Selects the **Normal** Decimal Notation mode setting; Displays results with digits to the left and right of the decimal (as opposed to scientific notation).

123E⁻2 [ENTER] 1.23

items **nPr** *number*

[MATH] **PRB**

Returns the number of permutations of n *items* taken r *number* at a time. The order in which you select the items DOES matter. *items* and *number* can be nonnegative integers or lists.

If both arguments are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is paired with each element in the list, and list of permutations is returned.

5 nPr 2 [ENTER] 20

5 nPr {2,4,6,8} [ENTER] {20 120 0 0}

conditionA **or** *conditionB*

[2nd] [TEXT]

Logic (boolean) operator; returns **1** if either *conditionA* or *conditionB* is true (non-zero). Returns **0** if both *conditionA* or *conditionB* are false (zero). *conditionA* and *conditionB* can be real numbers, expressions, or lists

If both *conditions* are lists, they must have the same number of elements. If one *condition* is a list and the other a non-list, the non-list is compared with each element of the list, and a list is returned.

Test operations are frequently used in programs.

```
PROGRAM:OR
:1→A
:2→B
:A>0 or B<0
```

Output(*row*,*column*,"*text*"**)**
Output(*row*,*column*,*value***)**

‡[PRGM] **I/O**

Programming command; displays *text* or *value* at the beginning of specified *row* and *column*. You must surround *text* with quotation marks ([2nd] [TEXT]).

```
PROGRAM:OUTPUT
:3+5→B
:ClrScreen
:Output(5,4,"ANSWER")
:Output(5,12,B)
:Pause
```

Pause [*value*]

‡[PRGM] **CTL**

Programming command; suspends program execution until you press [ENTER] or displays *value* and suspends program execution until you press [ENTER].

```
PROGRAM:PAUSE
:10→X
:"X+2"→Y1
:ZStandard
:Pause
```

PictoPlot

See **Plot1**: Pictograph

PiePlot

See **Plot1**: Pie Chart

Plot1(*type*,*argument1*,*argument2*,…**)**
Plot2(*type*,*argument1*,*argument2*,…**)**
Plot3(*type*,*argument1*,*argument2*,…**)**

†[2nd] [PLOT] **PLOTS**

Selects and defines Statistical Plot 1,2, or 3 (**Plot1, Plot2, Plot3**), according to one of eight stat plot *types*. All *types* and corresponding *arguments* are listed next. Select *type* from the [2nd] [PLOT] **TYPE** menu.

(*continued*)

Scatter Plot
xyLine Plot

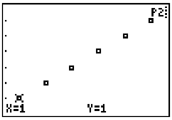
Plot*n*(**Scatter,***Xlist*,*Ylist*[,*mark*])
Plot*n*(**xyLine,***Xlist*,*Ylist*[,*mark*])

The optional *mark* (□, **+**, or **•**) specifies the character used to plot the points. If omitted, default *mark* is box. Access *mark* from [PRGM] [2nd] [PLOT] **MARK** or [2nd] [CATALOG].

{1,2,3,4,5,6}→L1 [ENTER]
{1 2 3 4 5 6}

{1,2,3,4,5,6}→L2 [ENTER]
{1 2 3 4 5 6}

PROGRAM:SCATTER
:PlotsOff
:Plot2(Scatter,L1,L2)
:ZStat
:Trace



Pictograph

Plot*n*(**PictoPlot,***CategList*,*DataList*,*scale*,*orientation*,*typeIcon*)

orientation=**0** (vertical) or **1** (horizontal).

typeIcon choices: **PersonIcon**; **TreeIcon**; **DollarIcon**; **FaceIcon**; **PieIcon**; **DiamondIcon**; **StarIcon**. Access *typeIcons* from [PRGM] [2nd] [PLOT] **MARK** or [2nd] [CATALOG].

Bar Graph

Plot*n***(BarPlot**,*CategList*, *orientation*, *DataList1*[,*DataList2*,*DataList3*]**)**

orientation=**0** (vertical) or **1** (horizontal). Specify between 1 and 4 *DataLists*.

Pie Chart

Plot*n***(PiePlot**,*CategList*,*DataList*,*type***)**

type=**0** (Number Pie Chart) or **1** (Percent Pie Chart).

Histogram
Box Plot
Modified Box Plot

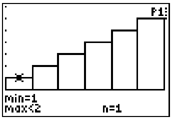
Plot*n***(Histogram,***Xlist*[,*freq*]**)**
Plot*n***(BoxPlot,***Xlist*[,*freq*]**)**
Plot*n***(ModBoxPlot,***Xlist*[,*freq*,*mark*]**)**

freq=**1** (default) or a list name. The optional *mark* (□;**+**;**•**) specifies the character used to plot the points. If omitted, default *mark* is box. Access *mark* from [PRGM] [2nd] [PLOT] **MARK** or [2nd] [CATALOG].

{1,2,3,4,5,6}→L1 [ENTER]
{1 2 3 4 5 6}

{1,2,3,4,5,6}→L2 [ENTER]
{1 2 3 4 5 6}

PROGRAM:HISTOGRM
:PlotsOff
:Plot1(Histogram,L1,L2)
:ZStat
:Trace



PlotsOff [**1**,**2**,**3**]
PlotsOn [**1**,**2**,**3**]

[2nd] [PLOT]

Turns off (deselects) or on (selects) all stat plots if no arguments are specified, or turns off or on specified stat plots using 1, 2, or 3, (for **Plot1**, **Plot2**, or **Plot3**).

PlotsOff 1,3[ENTER] Done

prgm*name*

‡[PRGM] **CTRL**

Programming command; calls **prgm***name* as a subroutine in an existing program. *name* can be a program not yet created.

PROGRAM:VOLUME
:Input "DIAMETER=",D
:Input "HEIGHT=",H
:prgmAREA
:A∗H→V
:Disp "VOLUME=",V
:Pause

PROGRAM:AREA
:D/2→R
:π∗R²→A
:Return

Prompt *variableA*[,*variableB*,…]

‡[PRGM] **I/O**

Programming command; displays specified *variable* followed by **=?**. During program execution, at each prompt, the user enters a value or expression for each *variable*, and then presses [ENTER]. $\mathbf{Y_n}$ functions are not valid with **Prompt**.

PROGRAM:PROMPT
:Prompt Xmin
:Prompt Xmax
:Prompt Ymin
:Prompt Ymax

The calculator adjusts **WINDOW** variable values according to user's input.

Pt-Change(*X*,*Y***)**

[DRAW] **POINTS**

Changes a point's status (on or off) at (*X*,*Y*).

Pt-Change(⁻6,2) [ENTER]

Pt-Off(*X*,*Y*[,*mark*]**)**
Pt-On(*X*,*Y*[,*mark*]**)**

[DRAW] **POINTS**

Erases or draws a point at (*X*,*Y*) using *mark*, (**1** = **•**; **2** = □; **3** = **+**). If *mark* is omitted, the default *mark* is box. If you specified *mark* to turn on a point with **Pt-On(**, you must specify the same *mark* when turning it off.

Pt-Off(3,5,2) [ENTER]
Pt-On(3,5,2) [ENTER]

Pxl-Change(*row*,*column***)**

[DRAW] **POINTS**

Changes a pixel's status (on or off) at (*row*, *column*); 0≤*row*≤62, and 0≤*column*≤94.

PxlChange(10,75) [ENTER]

Pxl-Off(*row*,*column***)**
Pxl-On(*row*,*column***)**

[DRAW] **POINTS**

Erases or draws a pixel at (*row*, *column*); 0≤*row*≤62, and 0≤*column*≤94.

Pxl-Off(10,75) [ENTER]
Pxl-On(10,75) [ENTER]

pxl-Test(*row*,*column***)**

[DRAW] **POINTS**

Returns **1** if pixel at (*row*, *column*) is on; returns **0** if it is off; 0≤*row*≤62, and 0≤*column*≤94.

Pxl-On(10,75)[ENTER] Done
pxl-Test(10,75)[ENTER] 1

QuadReg [*XList*,*YList*,*freq*,Y_n]

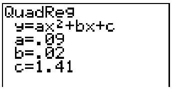
[2nd] [STAT] **CALC**

Fits the second-degree polynomial (y=ax^2+bx+c) to *XList* and *YList* with frequency list, *freq*, and stores the regression equation to Y_n. *XList*, *YList*, and *freq* (if specified) must have the same number of elements.

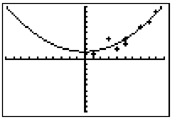
freq is the frequency of occurrence for each corresponding data point in *XList*. If *freq* is omitted, all values are used once.

Defaults for *XList* and *YList* are **L1** and **L2**.

Decimal mode set to 2:
{1,3,4,5,5,7,8,9}→L3 [ENTER] Done
{1,4,2,3,4,6,7,9}→L4 [ENTER] Done
QuadReg L3,L4,Y_1 [ENTER]



Select ZStandard.



Radian

† [MODE]

Sets the **Radian** Angle mode setting. Interprets angles as radians.

In Radian mode:
sin(90)[ENTER] .8939966636
sin(π/2) [ENTER] 1

---

rand
seed [STO▸]**rand**

[MATH] **PRB**

Generates a random number between **0** and **1**. By storing an integer seed value (default=0) to **rand**, you can control a random number sequence.

0→rand[ENTER] 0
rand[ENTER] .9435974025
rand[ENTER] .908318861

1→rand[ENTER] 1
rand[ENTER] .7455607728
rand[ENTER] .8559005971

---

randInt(*lower*,*upper*[,*#ofIntegers*]**)**

[MATH] **PRB**

Generates a random integer between *lower* and *upper* (both integers) boundaries. To generate more than one random integer, specify *#ofIntegers*, a positive whole number>0.

(Results may vary.)

randInt(1,10)[ENTER] 3
randInt(1,10,3)[ENTER]
{3 5 7}

RecallPic *number*

[DRAW] **STO**

Displays the current graph and superimposes **Pic***number* on it. *number* can be **1 (Pic1), 2 (Pic2),** or **3 (Pic3).**

Line(0,0,6,6)[ENTER]
StorePic 2[ENTER] Done

RecallPic 2[ENTER]
(Pic2 displayed)

remainder(*dividend*,*divisor***)**
remainder(*list*,*divisor***)**
remainder(*dividend*,*list***)**
remainder(*list*,*list***)**

[MATH] **NUM**

Returns the remainder resulting from the division of two positive whole numbers, *dividend* and *divisor*, each of which can be a list.

If both arguments are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is paired with each element of the list, and a list is returned.

remainder(10,4)[ENTER] 2

{5,5,5,5,5}→L1 [ENTER]
{5 5 5 5 5}
{1,2,3,4,5}→L2 [ENTER]
{1 2 3 4 5}
remainder(L1,L2) [ENTER]
{0 1 2 1 0}

:**Repeat** *condition*
:*block*
:**End**
:*command*

‡[PRGM] **CTL**

Programming command; executes *block* until *condition* is true.

```
PROGRAM:REPEAT
:0→I:0→J
:Repeat I≥6
: I+1→I:J+1→J
:Disp "J=",J
:Pause
:End
```

Return

‡[PRGM] **CTL**

Programming command; returns to the calling program.

```
PROGRAM:AREA
:D/2→R
:π*R²→A
:Return
```

```
PROGRAM:RETURN
:Input "DIAMETER=",D
:Input "HEIGHT=",H
:prgmAREA
:A*H→V
:Disp "VOLUME=",V
```

round(*value*[,#*decimal_places*]**)**

[MATH] **NUM**

Returns a number, expression, or each element in a list rounded to 10 digits or #*decimal_places* (≤9), if specified.

In Float mode:
round(π,4) [ENTER] 3.1416

round(π) [ENTER]
3.141592654

Scatter

See **Plot1**: Scatter Plot

Sci

†[MODE]

Selects the **Sci** Numeric Notation mode setting. Displays results in scientific notation.

123 [ENTER] 1.23E2

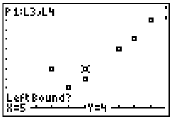
Select(*XList*,*YList***)**

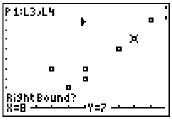
[2nd] [STAT] **OPS**

Selects one or more specific data points from a Scatter or xyLine stat plot, and updates the lists in memory as specified by *XList* and *YList*.

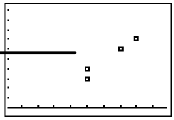
{1,3,4,5,5,7,8,9}→L3 [ENTER] Done
{1,4,2,3,4,6,7,9}→L4 [ENTER] Done
Select L5,L6 [ENTER]

Select bounds by moving [▶] and [◀], and then pressing [ENTER].





X values for selected points are stored in **L5**; *Y* values for selected points are stored in **L6**.



Send(*variable***)**

‡ [PRGM] **I/O**

Programming command; sends contents of *variable* to the CBL 2/CBL System.

PROGRAM:SEND
:{1,2,3,4,5}→L1
:Send(L1)

---

seq(*expression*,*variable*,*begin*,*end* [,*increment*]**)**

[2nd] [STAT] **OPS**

Returns a list that fulfills the requirements of the five arguments, which you specify. *increment* is 1 if not specified.

seq(X^2,X,1,8,2) [ENTER]
{1 9 25 49}

SetConst(*expression*,C_n**)**

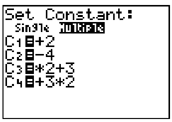
† [2nd] [SET] –or– [2nd] [CATALOG]

Programming command equivalent of the **Set Constant** editor. *expression* defines the constant to be recalled, and C_n is **1**, **2**, **3**, or **4** (for **C_1**, **C_2**, **C_3**, or **C_4**).

PROGRAM:SETCONST
:MultiConst
:SetConst(+2,1)
:SetConst(−4,2)
:SetConst(∗2+3,3)
:SetConst(+3∗2,4)

prgmSETCONST [ENTER] Done

All constants are defined in the **Set Constant** editor.



SetMenu("*title*","*item1*",*variable1*[,"*item2*",*variable2*...]**)**

‡[PRGM] **CTL**

Sets up a menu with *title* (1≤characters≤16) and of up to seven *items* (1≤characters≤10). During program execution, the user inputs (and edits, as necessary) numerical values, called *variables*, to each *item*.

PROGRAM:SETMENU
:SetMenu("MATHGRADES",
"TEST1",A,"TEST2",B,"TEST3",
C,"TEST4",D,"TEST5",E

SetUpEditor [*list1*,*list2*,*list3*...]

[2nd] [CATALOG]

Removes all list names from the List editor, and then sets it up to display *lists* in the specified order, starting with column **1**. If no lists are specified, the calculator sets up **L1**-**L6** in order and includes one blank list to the right of **L6**.

{1,2,3,4}→L1 [ENTER] {1 2 3 4}
{5,6,7,8}→L2 [ENTER] {5 6 7 8}
SetUpEditor L1,L2 [ENTER] Done

Press [LIST] to view List editor.

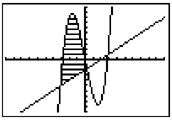
Shade(*lower*,*upper*[,*left*,*right*,*pattern*,*res*]**)**

[DRAW] **DRAW**

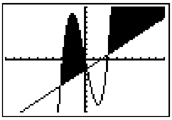
Draws both functions, *lower* and *upper*, shading above *lower* and below *upper*. You can limit shading by defining up to four optional arguments. Specify *left* and *right X* boundaries, *pattern*, which can equal 1-4 (descriptions shown below), and *res*, which equals 1-8 (1=highest resolution; 8=lowest resolution).

Pattern:
1 = vertical (default)
2 = horizontal
3 = diagonal upper left to lower right
4 = diagonal lower left to upper right

Shade(X–2,X^3–8X,$^{-}$5,1,2,3)
[ENTER]



ClrDrw[ENTER] Done
Shade(X^3–8X,X–2) [ENTER]



▶Simp [*simplification_factor*]

[SIMP]

In **Mansimp** Simplification mode, **Simp** simplifies a fraction by its lowest common factor (default) or by *simplification_factor*.

In Mansimp mode:

$\frac{24}{36}$▶Simp[ENTER] Fac=2↓$\frac{12}{18}$

$\frac{24}{36}$▶Simp 12[ENTER] $\frac{2}{3}$

---

sin(*value***)**

[2nd] [TRIG] **TRIG**

Returns the sine of a real number, expression, or each element of a list. Results are determined by the Angle mode setting (**Degree** or **Radian**).

In Degree mode:
sin(30)[ENTER] .5
sin({0,30,90})[ENTER]
{0 .5 1}

In Radian mode:
sin(π/2) [ENTER] 1
sin({0,π/2,π}) [ENTER]
{0 1 0}

sin⁻¹(*value*)

[2nd] [TRIG] **TRIG**

Returns the arcsine of a real number, expression, or each element of a list. ⁻1≤*value* ≤1. Results are determined by the Angle mode setting (**Degree** or **Radian**).

In Degree mode:
sin⁻¹(1) [ENTER] 90
sin⁻¹({1,.5,0}) [ENTER]
{90 30 60}

In Radian mode:
sin⁻¹(1) [ENTER] 1.570796327
sin⁻¹({1,.5,0}) [ENTER]
{.5235987756 0 1.570796327}

SingleConst

† [2nd] [SET]

Selects the **Single** mode (affects the **Set Constant** editor). Allows the user to access only one defined constant at a time.

SortA(*list*)
SortA(*indpntlist*,*dependlist1*,*dependlist2*,...)

[2nd] [STAT] **OPS**

Sorts *list* elements from lowest to highest value (ascending order) and categorical lists alphabetically.

When using dependent lists, *dependlist*, the calculator sorts *indpndtlist* first, and then sorts all *dependlists* by placing their elements in the same order as their corresponding elements in the independent list.

{5,8,⁻4,0, ⁻6}→L1 [ENTER]
{5 8 ⁻4 0 ⁻6}
SortA (L1) [ENTER] Done
L1 [ENTER] {⁻6 ⁻4 0 5 8}

{"E","A","Z"}→L2 [ENTER]
{"E" "A" "Z"}
SortA (L2) [ENTER] Done
L2 [ENTER] {"A" "E" "Z"}

SortD(*list*)
SortD(*indpntlist*,*dependlist1*,*dependlist2*,...)

[2nd] [STAT] **OPS**

Sorts *list* elements from highest to lowest value (descending order) and categorical lists in reverse alphabetical order.

When using dependent lists, *dependlist*, the calculator sorts *indpndtlist* first, and then sorts all *dependlists* by placing their elements in the same order as their corresponding elements in the independent list.

{5,8,⁻4,0d,⁻6}→L1 [ENTER]
{5 8 ⁻4 0 ⁻6}
SortD (L1) [ENTER] Done
L1 [ENTER] {8 5 0 ⁻4 ⁻6}

{"E","A","Z"}→L2 [ENTER]
{"E" "A" "Z"}
SortD(L2) [ENTER] Done
L2 [ENTER] {"Z" "E" "A"}

stdDev(*list*[,*freq*,*type*])

[2nd] [STAT] **MATH**

Returns the standard deviation of *list*. If a second list, *freq*, is specified, it is interpreted as the frequency of the elements in the first list. *list* and *freq* must have the same number of elements.

type=**0** (population standard deviation) or **1** (sample population deviation). If *type* is not specified, the calculator returns sample population deviation.

{1,2,8,10,11,21}→L1 [ENTER]
{1 2 8 10 11 21}

stdDev(L1) [ENTER]
7.250287351

Stop ‡ [PRGM] **CTL** Programming command; ends program execution and returns to Home screen.	PROGRAM:STOP :Input "T=",T :If T≥20 :Then :Disp "T≥20" :Pause :Else :Stop
StorePic *number* [DRAW] **STO** Stores the current graph display in one of three picture variables. *number* is **1**, **2**, or **3** (for variable **Pic1, Pic2, or Pic3**).	Line(0,0,6,6) [ENTER] StorePic 2 [ENTER] Done
sum(*list*[,start,end]) [2nd] [STAT] **MATH** Returns the sum of all elements in *list*. Specify the additional optional arguments to return the sum of the range of elements between *start* and *end*.	sum({1,2,4,8}) [ENTER] 15 sum({1,2,4,8},2,4) [ENTER] 14 sum({1,2,4,8},3) [ENTER] 12
tan(*value***)** [2nd] [TRIG] **TRIG** Returns the tangent of a real number, expression, or all elements in a list. Results are determined by the Angle mode setting (**Degree** or **Radian**).	In Degree mode: tan(0) [ENTER] 0 tan({0,1,30}) [ENTER] {0 .0174550649 .5773502692} In Radian mode: tan(π/4) [ENTER] 1 tan({π/2,π/4,0}) [ENTER] {1 0 0}
tan^{-1}(*value***)** [2nd] [TRIG] **TRIG** Returns the arctangent of a real number, expression, or each element in a list. Since tan=sin/cos, tan^{-1} is undefined when cos=0. Results are determined by the Angle mode setting (**Degree** or **Radian**).	In Degree mode: tan^{-1}(1) [ENTER] 45 tan^{-1}({.5,1,0)) [ENTER] {26.56505118 45 0} In Radian mode: tan^{-1}(.5) [ENTER] .463647609 tan^{-1}({.5,1,0)) [ENTER] {.463647609 .7853981634 0}

Text(*row*,*column*,["]*text*["]**)**

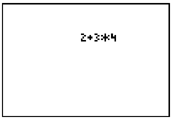
DRAW **DRAW**

Draws *text* (functions, variables, or text instructions) on the Graph screen when a graph is displayed.

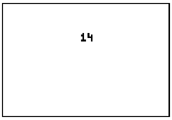
0 ≤ *row* ≤ 57, and 0 ≤ *column* ≤ 94.

If *text* is surrounded by quotation marks, the text characters are displayed. If the quotation marks are omitted, the TI-73 calculates and displays the result (up to 10 characters).

Select AxesOff.
Text(15,45,"2+3*4") ENTER



Return to the Home screen
ClrDraw ENTER Done
Text(15,45,2+3*4) ENTER



Then

See **If-Then-End**

Trace

† TRACE

Selects the TRACE mode when displaying a graph.

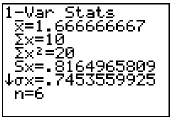
PROGRAM:TRACE
:"X²"→Y_1
:DispGraph
:Trace

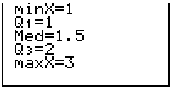
1-Var Stats [*XList*,*freq*]

2nd [STAT] **CALC**

Analyzes and returns data for one list, *XList*, with one measured variable (*X*). The frequency list, *freq*, is the frequency of occurrence for each corresponding data point in *XList*. Default *XList* is **L1**.

{1,2,3}→L2 ENTER {1 2 3}
{3,2,1}→ʟFREQ ENTER {3 2 1}
1-Var Stats L2,ʟFREQ ENTER





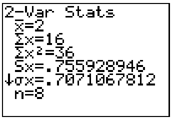
2-Var Stats [*XList*,*YList*,*freq*]

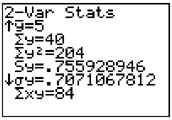
[2nd] [STAT] **CALC**

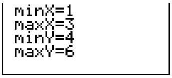
Analyzes and returns data for two lists, *XList* and *YList*, with two measured variables, *X*, the independent variable, and *Y*, the dependent variable.

The frequency list, *freq*, is the frequency of occurrence for each corresponding data point in *XList* and *YList*. Defaults for *XList* and *YList* are **L1** and **L2**.

{1,2,3}→L2 [ENTER]	{1 2 3}
{4,5,6}→L3 [ENTER]	{4 5 6}
{2,4,2}→ʟFREQ [ENTER]	{2 4 2}
2-Var Stats L2,L3,ʟFREQ [ENTER]	





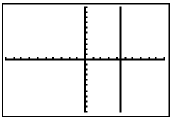


Vertical x

[DRAW] **DRAW**

Draws a vertical line on the current graph at X=x. x can be an expression but not a list.

Vertical 4.5[ENTER]



:**While** *condition*
:*block* (while *condition* is true)
:**End**
:*command*

‡[PRGM] **CTL**

Programming command; tests *condition* before commands in loop are executed. **While** *block* of commands WHILE *condition* is true.

```
PROGRAM:WHILE
:0→I:0→J
:While I<6
:I+1→I:J+1→J
:Disp "J=",J
:Pause
:End
```

xyLine

See **Plot1**: xyLine Plot

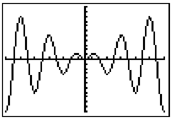
ZBox

ZOOM **ZOOM**

Displays a graph, lets you (interactively) draw a box that defines a new viewing **WINDOW**, and then updates the **WINDOW**.

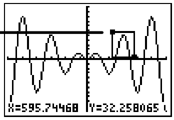
Define Y_1=Xsin(X).
Set the following WINDOW values:
Xmin=$^{-}$1000, Ymin=$^{-}$1000,
Xmax=1000, Ymax=1000,
Xscl=90, Yscl=90

Graph Y_1.

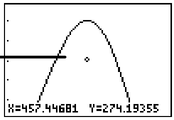


Select ZBox.

Move the cursor, and press ENTER to select upper left and bottom right corners of the box.



Selected portion (the box) is automatically displayed.



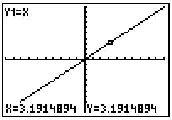
ZDecimal

ZOOM **ZOOM**

Adjusts the viewing **WINDOW** so that **ΔX=0.1** and **ΔY=0.1**, and displays the graph screen with the origin centered on the screen.

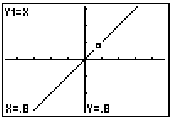
Define Y_1=X, graph using ZStandard, and trace the graph

X and Y values increment by .212466.



Select ZDecimal and trace the graph.

X and Y values now increment by .1.

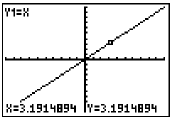


ZInteger

[ZOOM] **ZOOM**

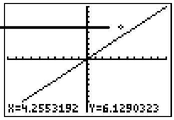
Lets you select a new center point, and then sets **ΔX=1**, **ΔY=1**, **Xscl=10**,**Yscl=10**. Replots the graph immediately.

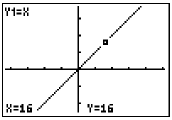
Define Y_1=X, graph using ZStandard, and trace the graph



Select ZInteger, choose a new center point, and trace the graph.

Move the cursor, and press [ENTER] to select new center point.





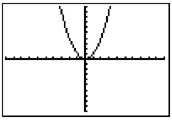
X and Y values now increment by 1.

Zoom In

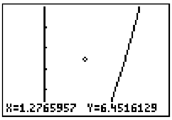
[ZOOM] **ZOOM**

Lets you select a new center point, if desired, and then magnifies the part of the graph that surrounds the cursor location.

Define Y_1=X^2 and graph using ZStandard.



Select Zoom In. Move the cursor to the upper right section of the graph. Press [ENTER] to select a new center point and magnify the upper right side of the graphed function.



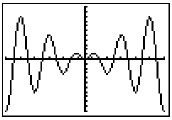
Zoom Out

[ZOOM] **ZOOM**

Displays a greater portion of the graph, centered on the cursor location. Move the cursor keys, and press [ENTER] to select new center point.

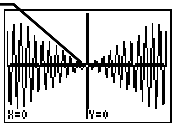
Define Y_1=Xcos(X) and graph using ZStandard.

Set the following WINDOW values:
Xmin=$^{-}$1000, Ymin=$^{-}$1000,
Xmax=1000, Ymax=1000,
Xscl=90, Yscl=90



Zoom Out from the origin.

Select **Zoom Out**, and then press [ENTER] (since cursor automatically starts from the origin).

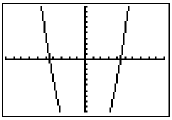


ZoomFit

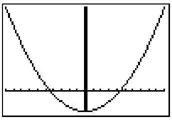
[ZOOM] **ZOOM**

Recalculates **Ymin** and **Ymax** to include the minimum and maximum y values, between **Xmin** and **Xmax**, of the selected functions and replots the functions.

Define Y_1=X^2−20 graph using using standard WINDOW value ([ZOOM] 6).



Adjust the graph with ZoomFit.



ZoomStat

[ZOOM] **ZOOM**

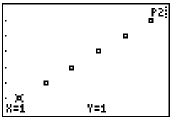
Redefines the viewing **WINDOW** so that all statistical data points are displayed.

ZoomStat also selects an appropriate scale, if one exists, for a Pictograph plot.

{1,2,3,4,5,6}→L1 [ENTER]
{1 2 3 4 5 6}

{1,2,3,4,5,6}→L2 [ENTER]
{1 2 3 4 5 6}

Graph and trace a Scatter stat plot using L1 and L2 ([2nd] [PLOT]) and ZoomStat.



ZPrevious

[ZOOM] **MEMORY**

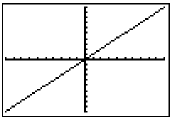
Replots the graph using the **WINDOW** variable values of the graph that was displayed before you executed the last **ZOOM** instruction.

ZQuadrant1

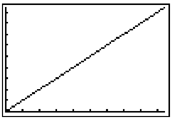
[ZOOM] **ZOOM**

Replots the graph using **WINDOW** variable values for Quadrant I (**Xmin=0**, **Xmax=9.4**, **Xscl=1, Ymin=0**, **Ymax=9.4**, **Yscl=1**).

Define Y_1=X
using ZStandard ([ZOOM] 6).



Select ZQuadrant1.



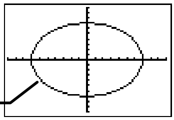
ZSquare

[ZOOM] **ZOOM**

Adjusts the **X** or **Y** **WINDOW** settings so that each pixel represents an equal width and height in the coordinate system and updates the viewing **WINDOW**.

Select ZStandard. Return to the Home screen.

Circle(0,0,7)[ENTER]

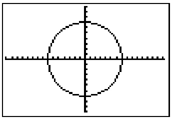


Circle is oval shaped instead of perfectly round.

(Continued)

Select ZSquare. Return to the Home screen.

Circle(0,0,7) [ENTER] — Or, press [2nd] [ENTRY] [ENTER].

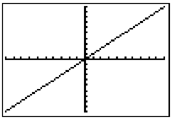


ZStandard

[ZOOM] **ZOOM**

Replots the functions immediately, setting the **WINDOW** variables to the default values (**Xmin=⁻10**, **Xmax=10**, **Xscl=1**, **Ymin=⁻10**, **Ymax=10**, **Yscl=1**).

Define Y_1=X
Select ZStandard.

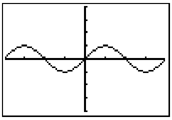


ZTrig

[ZOOM] **ZOOM**

Replots the functions immediately, updating the **WINDOW** variables that are often appropriate for graphing trig functions.

Define Y_1=sin(X) ([Y=]).
Select ZTrig.



---

value!

[MATH] **PRB**

Returns the factorial of *value*. *value* can be an integer or list of integers between 0 and 69.

5! [ENTER]	120

---

angle°

[2nd] [TRIG] **ANGLE**

Designates *angle* as degrees, regardless of the current Angle mode setting or DMS notation.

In Radian mode:

50° [ENTER]	.872664626
50°▸DMS [ENTER]	50°0'0"

In Degree mode:

50° [ENTER]	50
50°▸DMS [ENTER]	50°0'0"

angle^r [2nd] [TRIG] **ANGLE** Specifies an angle as radians, regardless of the current Angle mode setting.	In Radian mode: 50^r [ENTER] 50 50▶DMS [ENTER] 2864°47'20.312" In Degree mode: 50^r [ENTER] 2864.788976 50^r ▶DMS [ENTER] 2864°47'20.312"
$x\,^x\!\sqrt{value}$ [MATH] **MATH** Calculates the x^{th} root of *value*, which is equivalent to n where n^x=*value*. *value* can be a real number, expression, or list.	4 $^x\!\sqrt{256}$ [ENTER] 4
n^3 [MATH] **MATH** Calculates the cube of n, which is equivalent to $n \times n \times n$ of any real number, expression, or each element in a list.	2^3 [ENTER] 8
$^3\!\sqrt{}$(*value*) [MATH] **MATH** Calculates the cube root of *value*, which is equivalent to n where n^3=*value*. *value* can be a real number, expression, or list.	$^3\!\sqrt{(8)}$ [ENTER] 2
real_number% [%] Changes *real_number* to percent. Results display according to the Decimal mode setting.	In Float mode: ⁻30.6% [ENTER] ⁻.306 20 % ∗ 30 [ENTER] 6
conditionA = *conditionB* (equal) *conditionA* ≠ *conditionB* (not equal) *conditionA* < *conditionB* (less than) *conditionA* > *conditionB* (greater than) *conditionA* ≤ *conditionB* (less than or equal to) *conditionA* ≥ *conditionA* (greater than or equal to) [2nd] [TEXT] Relational operators; return **1** if the conditional statement is true. Return **0** if the conditional statement is false. *conditionA* and *conditionB* can be real numbers, expressions, or lists. If both *conditions* are lists, they must have the same number of elements. If one *condition* is a list and the other a non-list, the non-list is compared with each element of the list, and a list is returned.	In Degree mode: sin(30)=cos(60) [ENTER] 1 sin(30)≠cos(90) [ENTER] 1 sin(30)<cos(90) [ENTER] 0 sin(30)>cos(90) [ENTER] 1 sin(30)≤cos(60) [ENTER] 1 sin(30)≥cos(90) [ENTER] 1

□
+
•
See **Plot1**: xyLine, Scatter, and Modified Box Plot: *mark*

PersonIcon
TreeIcon
DollarIcon
FaceIcon
PieIcon
DiamondIcon
StarIcon
See **Plot1**: Pictograph: *typeIcons*

---

value^{-1}

[2nd] [x^{-1}]

Returns the inverse, x^{-1}, of *value*, which is the equivalent of its reciprocal, $1/x$, of a real number, expression, or each element in a list.

In b/c mode:

$\frac{2}{3}^{-1}$ [ENTER]	$\frac{3}{2}$

---

value^{2}

[x^2]

Finds the square of *value*. *value* can be a real number, expression, or list.

{1,2,3}→L1 [ENTER]	{1 2 3}
L1^2 [ENTER]	{1 4 9}

---

value^*power*

[^]

Raises *value* to any power. *value* and power can be real numbers, expressions, or lists. *power* is limited by mathematical rules.

4^4 [ENTER]	256

⁻

[(−)]

Negates a number, expression, or each element in a list. **Note**: This is different from the subtraction key ([−]).

⁻14-68 [ENTER]	⁻82
⁻(4^2) [ENTER]	⁻16
(⁻4^2) [ENTER]	16
⁻{1,2,3} [ENTER]	{⁻1 ⁻2 ⁻3}

10^(*x*)
10^(*list*)

[MATH] LOG

Raises 10 to the power of x, where x is an integer or a list of integers. If x≤⁻4 and ≥10^{10}, then the result is displayed in scientific notation.

10^(4) [ENTER]	1000
10^(⁻4) [ENTER]	1E⁻4
10^({1,2,3}) [ENTER]	
	{10 100 1000

√(*value*) [2nd] [√] Calculates the square root of value, which can be a positive real number, an expression that results in a positive real number, or a list of positive numbers.	√(16) [ENTER] 4
valueA∗*valueB* *valueA*/*valueB* *valueA*+*valueB* *valueA*−*valueB* [×] [÷] [+] [−] Returns the product ([x]), quotient ([÷]), sum ([+]) or difference ([−]) of valueA and valueB, which can be real numbers, expressions, or lists. If both values are lists, they must have the same number of elements. If one argument is a list and the other a non-list, the non-list is paired with each element of the list, and a list is returned.	{1,4,8}→L1 [ENTER] {1 4 8} 4∗L1 [ENTER] {4 16 32} {2,4,8}/{2,2,2} [ENTER] {1 2 4} ⁻456−123 [ENTER] ⁻579 In Autosimp mode: $\frac{2}{3}*\frac{3}{4}$ [ENTER] $\frac{1}{2}$ In A⌞b/c mode: $4+\frac{1}{2}$ [ENTER] $4\frac{1}{2}$
{ [2nd] [TEXT] Signifies the beginning of a list.	{"A","B","C"}→L3 {"A" "B" "C"}
([(] Designates a 1st priority calculation or implies multiplication.	4(3) [ENTER] 12 (4+4)6÷8 [ENTER] 6 4+4(6÷8) [ENTER] 7
} [2nd] [TEXT] Signifies the end of a list.	{"A","B","C"}→L3 {"A" "B" "C"}
) [)] Designates a 1st priority calculation, implies multiplication, or completes functions and instructions.	4(3) [ENTER] 12 (4+4)6÷8 [ENTER] 6 4+4(6÷8) [ENTER] 7 log(10) [ENTER] 1
, [,] Separates list elements when entering them outside of the List editor, and separates function/programming command arguments.	{"A","B","C"}→L3 {"A" "B" "C"} Circle(0,0,7) [ENTER]
' [2nd] [TRIG] **ANGLE** Specifies the minutes in DMS angle notation.	In Radian mode: 50°0'0" [ENTER] .872664626

" [2nd] [TEXT] [2nd] [TRIG] **ANGLE** Surrounds categorical list elements and list formulas that are attached to a list name. Surrounds text displayed on the Graph display using the **Text(** command (from the Home screen or in a Program). In a programming command, they surround text to be displayed with **Disp,** text which designates an **Input** prompt, and functions that are assigned to a **Y$_n$** variable. Specifies seconds in DMS angle notation.	{"A","B","C"}→L3 [ENTER] {"A" "B" "C"} PROGRAM:TEXT :AxesOff :Text(15,45,"TEXT") :DispGraph PROGRAM:FUNCTION :"2X+5"→Y1 :ZStandard PROGRAM:INPUT :Input "NEW LIST=",ʟNEW :Disp "ʟNEW=",ʟNEW :Pause In Radian mode: 50°0'0" [ENTER] .872664626
: [2nd] [CATALOG] Precedes all programming commands (automatically displayed by the calculator in the Program editor). Separates two programming commands listed on one line or two entries on the Home screen.	PROGRAM:GREETING :Disp "HI, TERESA":Pause
π [2nd] [π] Represents the value for the constant, π, in calculations. The calculator uses π=3.1415926535898.	In Float mode: 2π [ENTER] 6.283185307
? [2nd] [CATALOG] Displays a question mark, which acts like a text character.	PROGRAM:QUESTION :Disp "WHAT TIME IS IT?" :Pause

B Reference Information

The TI-73 Menu Map

The TI-73 menu map begins at the top-left corner of the keyboard and follows the keyboard layout from left to right. Default values and settings are shown.

[Y=]

Plot1 Plot2 Plot3

\Y1=

\Y2=

\Y3=

\Y4=

[2nd] [PLOT]

STAT PLOTS
1:Plot1…Off
L1 L2 □
2:Plot2…Off
L1 L2 □
3:Plot3…Off
L1 L2 □
4:PlotsOff
5:PlotsOn

[2nd] [PLOT] (in Program editor)

PLOTS	TYPE	MARK
1:Plot1(	1:Scatter	1:□
2:Plot2(	2:xyLine	2:+
3:Plot3(	3:PictoPlot	3:•
4:PlotsOff	4:BarPlot	4:PersonIcon
5:PlotsOn	5:PiePlot	5:TreeIcon
	6:Histogram	6:DollarIcon
	7:BoxPlot	7:FaceIcon
	8:ModBoxPlot	8:PieIcon
		9:DiamondIcon
		0:StarIcon

[WINDOW]

WINDOW
Xmin=-10
Xmax=10
ΔX=.2127659574…
Xscl=1
Ymin=-10
Ymax=10
Yscl=1

2nd [TBLSET]

TABLE SETUP
TblStart=0
ΔTbl=1
Indpnt:Auto Ask
Depend:Auto Ask

2nd [TBLSET] (in Program editor)

TABLE SETUP
Indpnt:Auto Ask
Depend:Auto Ask

ZOOM

ZOOM
1:ZBox
2:Zoom In
3:Zoom Out
4:ZQuadrant1
5:ZSquare
6:ZStandard
7:ZoomStat
8:ZDecimal
9:ZoomFit
0:ZInteger
A:ZTrig

MEMORY

1:ZPrevious

2:SetFactors…

ZOOM FACTORS
XFact=4
YFact=4

2nd [FORMAT]

CoordOn CoordOff
GridOff GridOn
AxesOn AxesOff
LabelOff LabelOn
ExprOn ExprOff

MODE

Normal Sci
Float 0123456789
Degree Radian
A␣b/c b/c
Autosimp Mansimp

MATH

MATH	NUM	PRB	LOG
1:lcm(	1:abs(	1:rand	1:log(
2:gcd(	2:round(	2:randInt(	2:10^(
3:^{3}	3:iPart(	3:nPr	3:ln(
4: $\sqrt[3]{}$(	4:fPart(	4:nCr	4:e^(
5: $\sqrt[x]{}$	5:min(	5:!	
6:Solver…	6:max(	6:coin(	
	7:remainder(	7:dice(	

DRAW

DRAW	POINTS	STO
1:ClrDraw	1:Pt-On(	1:StorePic
2:Line(	2:Pt-Off(	2:RecallPic
3:Horizontal	3:Pt-Change(	
4:Vertical	4:Pxl-On(	
5:Shade(	5:Pxl-Off(	
6:Circle(	6:Pxl-Change(	
7:Text(	7:pxl-Test(	
8:Pen		

2nd [TRIG]

TRIG	ANGLE
1:sin(	1:°
2:sin^{-1}(	2:'
3:cos(	3:"
4:cos^{-1}(	4:r
5:tan(	5:▶DMS
6:tan^{-1}(	

2nd [STAT]

Ls	OPS	MATH	CALC
1:L_1	1:SortA(	1:min(	1:1-Var Stats
2:L_2	2:SortD(	2:max(	2:2-Var Stats
3:L_3	3:ClrList	3:mean(	3:Manual-Fit
4:L_4	4:dim(	4:median(	4:Med-Med
5:L_5	5:ΔList(	5:mode(	5:LinReg(ax+b)
6:L_6	6:Select(	6:stdDev(	6:QuadReg
7:*name1*	7:seq(	7:sum(	7:ExpReg
8:*name2*	8:augment(		
…	9:L		

PRGM

EXEC	EDIT	NEW
1:*name1*	1:*name1*	1:Create New
2:*name2*	2:*name2*	
…	…	

PRGM (in Program editor)

CTL	I/O	EXEC
1:If	1:Input	1:*name1*
2:Then	2:Prompt	2:*name2*
3:Else	3:Disp	…
4:For(	4:DispGraph	
5:While	5:DispTable	
6:Repeat	6:Output(	
7:End	7:getKey	
8:Pause	8:ClrScreen	
9:Lbl	9:ClrTable	
0:Goto	0:GetCalc(	
A:IS>(	A:Get(	
B:DS<(	B:Send(	
C:Menu(		
D:SetMenu(		
E:prgm		
F:Return		
G:Stop		
H:DelVar		
I:GraphStyle(		

[2nd] [CATALOG]

CATALOG
A␣b/c
▸Ab/c⇔d/e
abs(
…
sin(
sin⁻1(
SingleConst
SortA(
…
π
?

[APPS]

APPLICATIONS

1:Link

SEND
1:All+…
2:All−…
3:Prgm…
4:List…
5:Pic…
6:Real…
7:Y-Vars…
8:Consts…
9:Vars to TI82…
0:Vars to TI83…
A:Apps…
B.AppVars…
C:SendId
D:Back Up…

RECEIVE
1:Receive

2:CBL/CBR

1:GAUGE
2:DATA LOGGER
3:CBR
4:QUIT

[2nd] [VARS]

VARS
1:Window…
2:Y-Vars…
3:Statistics…
4:Picture…
5:Table…
6:Factor

[2nd] [VARS] **1:Window**

WINDOW
1:Xmin
2:Xmax
3:Xscl
4:Ymin
5:Ymax
6:Yscl
7:Xres
8:ΔX
9:ΔY
0:XFact
A:YFact

[2nd] [VARS] **2:Y-Vars**

FUNCTION
1:Y_1
2:Y_2
3:Y_3
4:Y_4
5:FnOn
6:FnOff

[2nd] [VARS] **3:Statistics**

XY	Σ	EQ	PTS
1:n	1:Σx	1:RegEQ	1:x1
2:$\bar{x}$	2:Σx^2	2:a	2:y1
3:Sx	3:Σy	3:b	3:x2
4:σx	4:Σy^2	4:c	4:y2
5:$\bar{y}$	5:Σxy	5:r	5:x3
6:Sy		6:r^2	6:y3
7:σy		7:R^2	7:Q_1
8:minX			8:Med
9:maxX			9:Q_3
0:minY			
A:maxY			

[2nd] [VARS] **4:Picture**

1:Pic1 (Empty)
2:Pic2 (Empty)
3:Pic3 (Empty)

[2nd] [VARS] **5:Table**

TABLE
1:TblStart
2:ΔTbl

[2nd] [CONVERT]

CONVERSIONS
1:Length…
2:Area…
3:Volume…
4:Time…
5:Temp…
6:Mass/Weight…
7:Speed…

[2nd] [CONVERT] **1:Length**

LENGTH
1:mm
2:cm
3:m
4:inch
5:ft
6:yard
7:km
8:mile

[2nd] [CONVERT] **2:Area**

AREA
1:ft^2
2:m^2
3:mi^2
4:km^2
5:acre
6:in^2
7:cm^2
8:yd^2
9:ha

[2nd] [CONVERT] **3:Volume**

VOLUME
1:liter
2:gal
3:qt
4:pt
5:oz
6:cm^3
7:in^3
8:ft^3
9:m^3
0:galUK
A:ozUK

[2nd] [CONVERT] **5:Temp**

TEMP
1:degC
2:degF
3:degK

[2nd] [CONVERT] **6:Mass/Weight…**

MASS/WT.
1:g
2:kg
3:lb
4:ton
5:mton

[2nd] [CONVERT] **7:Speed…**

SPEED
1:ft/s
2:m/s
3:mi/hr
4:km/hr
5:knot

[2nd] [SET]

Set Constant:
Single Multiple
C_1=
C_2=
C_3=
C_4=

[2nd] [SET] (in Program editor)

SET CONSTANTS
1:SetConst(
2:SingleConst
3:MultiConst

[2nd] [MEM]

MEMORY
1:About
2:Check RAM…
3:Check APPs…
4:Delete…
5:Clear Home
6:ClrAllLists
7:Reset…

[2nd] [MEM] **2:Check RAM**

MEM FREE 25002	
Real	**15**
List	**54**
Y-Vars	**32**
Consts	**32**
Prgm	**15**
Pic	**0**

[2nd] [MEM] **3:Check APPs**

SPACES FREE	**3**
CBL/CBR	**1**

[2nd] [CONVERT] **4:Delete**

DELETE FROM…
1:All…
2:Real…
3:List…
4:Y-Vars…
5:Consts…
6:Prgm…
7:Pic…
8:Apps…
9:AppVars…

[2nd] [CONVERT] **7:Reset**

RESET

1:All RAM
RESET RAM

1:No

2:Reset

Resetting RAM erases all data and programs.

2:Defaults
RESET DEFAULTS

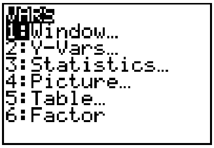
1:No

2:Reset

The VARS Menu [2nd] [VARS]

Access system variables through the **VARS** menu ([2nd] [VARS]). You can enter the names of functions and system variables in an expression or store values to them directly. For more information about storing values to a variable, see Chapter 1: Operating the TI-73.

All **VARS** menu items, except **6:Factor**, display secondary menus. For specific information about the individual menu items, see their respective chapter in this manual. When you select a variable from a menu, it is inserted at the cursor location.

[2nd] [VARS]	

1:Window	Accesses **WINDOW** screen ([WINDOW]) variables (Chapter 9: Function Graphing).
2:Y-Vars	Accesses Y= editor ([Y=]) variables (Chapter 9: Function Graphing).
3:Statistics	Accesses **1-Var Stats** and **2-Var Stats** ([2nd] [STAT] **CALC**) variables (Chapter 7: Statistical Analyses).
4:Picture	Accesses picture ([DRAW] **STO**) variables (Chapter 10: Draw).
5:Table	Accesses **TABLE SETUP** ([2nd] [TBLSET]) variables (Chapter 8: Tables).
6:Factor	Returns the simplification factor of a fraction after you simplify it using [SIMP] (Chapter 3: Fractions).

Equation Operating System (EOS ™)

The Equation Operating System (EOS) defines the order in which functions and expressions are entered and evaluated on the TI−73. Within a priority level, EOS evaluates functions from left to right and in the following order.

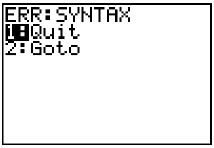
1	Calculations within parentheses.
2	Single-argument functions that precede the argument, such as **√(**, **sin(**, or **log(**. Multi-argument functions, such as **min(2,3)**, are evaluated as they are encountered.
3	Functions that are entered after the argument, such as **!**, **°**, **r**, and conversions.
4	Powers and roots, such as 2^5 or $\sqrt[5]{32}$.
5	Permutations (**nPr**) and combinations (**nCr**).
6	Multiplication, implied multiplication, and division.
7	Addition and subtraction.
8	Relational functions, such as **>** or ≤.
9	Logic operator **and**.
10	Logic operator **or**.

In Case of Difficulty

If	Suggested Action
You cannot see anything on the display.	Press [2nd] [▲] to darken or [2nd] [▼] to lighten the display contrast.
The **LOW BATTERY** message is displayed on the Home screen.	Replace the batteries as described in Appendix C: Battery/Service and Warranty Information.
A checkerboard cursor (▦) is displayed.	Either you have entered the maximum number of characters in a prompt or memory is full. If memory is full, press [2nd] [MEM] **4:Delete**, and then delete some items from memory (See Chapter 13: Memory Management).
The busy indicator (⁝) is displayed in the top right corner.	A calculation, graph, or program has been paused; the TI-73 is waiting for input. Press [ENTER] to continue, or press [ON] to break.
An error message is displayed.	Refer to the section in this chapter entitled "Error Messages." Press [ENTER] to clear.
The TI-73 does not appear to be working properly.	Press [2nd] [QUIT] as many times as needed to exit any menu and to return to the Home screen. – or – Be sure that the batteries are installed properly and that they are fresh.
The difficulty persists.	Refer to Appendix C: Battery/Service and Warranty Information for information on how to contact Customer Support to discuss the problem or to obtain service.

Correcting an Error

When the TI-73 detects an error, it returns an error message as a menu title, such as **ERR:SYNTAX** or **ERR:DIM MISMATCH**.



To correct an error, follow these steps:

1. Note the error type (**ERR**:*error type*).
2. Select **2:Goto**, if it is available. The previous screen is displayed with the cursor at or near the error location.
3. If you select **1:Quit** (or press [2nd] [QUIT] or [CLEAR]), the Home screen is displayed.
4. Determine the cause of the error. If you cannot recognize the error, use the Error Messages table below which describes error messages in detail.
5. Correct the expression.

   If a syntax error occurs in the contents of a $\mathbf{Y_n}$ function during program execution, selecting **2:Goto** returns you to the Y= editor, not to the program.

Error Messages

When the TI-73 detects an error, it displays **ERR:TYPE** and an error menu. This table contains each error type, possible causes, and suggestions for correction.

The TI-73 detects errors while performing the following tasks:

- Evaluating an expression
- Executing an instruction
- Plotting a graph or stat plot
- Storing a value

Error Type	Possible Causes and Suggested Remedies
ARGUMENT	A function or instruction does not have the correct number of arguments. See Appendix A and the appropriate chapter.
BAD GUESS	You specified a *guess* in the Equation Solver that is not between the lower and upper bounds. Your *guess* and several points around it are undefined. Examine a graph of the function. If the equation has a solution, change the bounds and/or initial *guess*.
BOUND	With **Select(**, you defined Left Bound>Right Bound. In the Equation Solver, you entered lower≥upper.
BREAK	You pressed the ON key to break execution of a program, to halt a **DRAW** instruction, or to stop evaluation of an expression.
DATA TYPE	You entered a value or variable that is the wrong data type. • For a function (including implied multiplication) or an instruction, you entered an argument that is an invalid data type, such as a real number where a list is required. • In an editor, you entered a type that is not allowed. • You attempted to store to an incorrect data type, such as a real number to a list.
DIM MISMATCH	You attempted to perform an operation that references more than one list, but the lists do not have the same dimension (number of elements).

Error Type	Possible Causes and Suggested Remedies
DIVIDE BY 0	You attempted to divide by 0. This error is not returned during graphing. The TI−73 allows for undefined values on a graph. You attempted a linear regression with a vertical line.
DOMAIN	You specified an argument to a function or instruction outside the valid range, such as using a negative frequency in box plots. This error is not returned during graphing because the TI−73 allows for undefined values on a graph. See Chapter 6: Statistical Plots or Chapter 9: Function Graphing. In a Pictograph, an element in **Data List** is too large so that the maximum scale (99999) can't make all icons fit in one screen. You attempted an exponential regression with a **⁻Y**.
Duplicate Name	A variable you attempted to transmit cannot be transmitted because a variable with that name already exists in the receiving unit.
Error in Xmit	The TI−73 was unable to transmit an item. Check to see that the cable is firmly connected to both units and that the receiving unit is in receive mode. You pressed [ON] to break during transmission. You attempted to perform a backup from a TI-83 to a TI−73. You attempted to transfer data (other than **L1**-**L6**) from a TI−73 to a TI-83 without using the **Lists to TI83** command. You attempted to use **Get(** with another calculator.

Error Type	Possible Causes and Suggested Remedies
ILLEGAL NEST	You attempted to use an invalid function in an argument to a function, such as **seq(** within expression for **seq(**. Can occur when combinations of nesting of function evaluation exceeds five levels.
INCREMENT	The increment in **seq(** is 0 or has the wrong sign. This error is not returned during graphing. The TI−73 allows for undefined values on a graph. The increment in a **For(** loop is 0.
INVALID	You attempted to reference a variable or use a function where it is not valid. For example, **Yn** cannot reference **Y**, **Xmin**, **ΔX**, or **TblStart**. Defining and graphing a **Yn** equation using the variable **Ans**. You attempted to use **Select(** without having selected (turned on) at least one xyLine or Scatter plot.
INVALID DIM	You specified dimensions for an argument that are not appropriate for the operation. You specified a list dimension as something other than an integer between 1 and 999.
ITERATIONS	The Equation Solver has exceeded the maximum number of permitted iterations. Examine a graph of the function. If the equation has a solution, change the bounds, the initial guess, or both.
LABEL	The label in the **Goto** instruction is not defined with a **Lbl** instruction in the program.

Error Type	Possible Causes and Suggested Remedies
MEMORY	Memory is insufficient to perform the instruction or function. You must delete items from memory (Chapter 13: Memory Management) before executing the instruction or function. Recursive problems return this error; for example, graphing the equation **Y1=Y1**. Branching out of an **If/Then**, **For(**, **While**, or **Repeat** loop with a **Goto** also can return this error because the **End** statement that terminates the loop is never reached.
MemoryFull	You are unable to transmit an item because the receiving unit's available memory is insufficient. You may skip the item or exit receive mode. During a memory backup, the receiving unit's available memory is insufficient to receive all items in the sending unit's memory. A message indicates the number of bytes that the sending unit must delete to do the memory backup. Delete items and try again.
MODE	You attempt to simplify a fraction with [SIMP] while in **Autosimp** Simplification mode.
NO SIGN CHANGE	The Equation Solver did not detect a sign change.
OVERFLOW	You attempted to enter, or you have calculated, a number that is beyond the range of the calculator. This error is not returned during graphing. The TI−73 allows for undefined values on a graph.
RESERVED	You attempted to use a system variable inappropriately. See Chapter 1: Operating the TI−73.
SCALE	The Pictograph scale is invalid. Scale must be an integer between 1 and 99,999.

Error Type	Possible Causes and Suggested Remedies
SINGULARITY	*expression* in the Equation Solver contains a singularity (a point at which the function is not defined). Examine a graph of the function. If the equation has a solution, change the bounds or the initial *guess* or both.
STAT	You attempted a stat calculation with lists that are not appropriate. • Statistical analyses must have at least two data points. • **Med-Med** must have at least three data points in each partition. • When you use a frequency list, its elements must be ≥0. • (**Xmax-Xmin**)/**Xscl** must be ≤47 for a Histogram.
STAT PLOT	You attempted to display a graph when a stat plot that uses an undefined list is turned on.
SYNTAX	The command contains a syntax error. Look for misplaced functions, arguments, parentheses, or commas. See the appropriate chapter.
UNDEFINED	You referenced a variable that is not currently defined. For example, you referenced a stat variable when there is no current calculation because a list has been edited, or you referenced a variable when the variable is not valid for the current calculation, such as **c** after **Med-Med**.
VALIDATION	Electrical interference caused a link to fail or this calculator is not authorized to run the application.

Error Type	Possible Causes and Suggested Remedies
WINDOW RANGE	A problem exists with the **WINDOW** variables. • You defined **Xmax≤Xmin** or **Ymax≤Ymin**. • **WINDOW** variables are too small or too large to graph correctly. You may have attempted to zoom in or zoom out to a point that exceeds the TI−73's numerical range.
ZOOM	A point or a line, instead of a box, is defined in **ZBox**. A **ZOOM** operation returned a math error.

C Battery/ Service and Warranty Information

Battery Information

The TI-73 uses four AAA alkaline batteries and has a user-replaceable backup lithium battery (CR1616 or CR1620).

When to Replace Batteries

When the battery voltage level drops below a usable level, the TI-73 displays the following message when you turn on the unit.

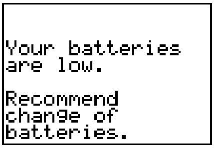
```
Your batteries
are low.

Recommend
change of
batteries.
```

Battery Information

The TI-73 uses four AAA alkaline batteries and has a user-replaceable backup lithium battery (CR1616 or CR1620).

When to Replace Batteries

When the battery voltage level drops below a usable level, the TI-73 displays the following message when you turn on the unit.



Generally, the calculator continues to operate for one week after the low-battery message is first displayed. After this period, the TI-73 will turn off automatically and the unit will not operate. Batteries must be replaced. All memory is retained.

Note: The operating period following the first low-battery message could be longer if you use the calculator infrequently or shorter if you use the calculator frequently.

Replace the lithium battery every three to four years.

The calculator does not let you install new software or application programming if the batteries are too low.

Effects of Replacing the Batteries

Do not remove both types of batteries (AAA and lithium auxiliary) at the same time. Do not allow the batteries to lose power completely. If you follow these guidelines and the steps for replacing batteries on the next page, then you can replace either type of battery without losing any information in memory.

Replacing the Batteries

1. Turn off the calculator. Replace the slide cover over the keyboard to avoid inadvertently turning on the calculator. Turn the back of the calculator toward you.

2. Hold the calculator upright, push downward on the latch on the top of the battery cover with your finger, and then pull the cover toward you.

   Note: To avoid loss of information stored in memory, you must turn off the calculator. Do not remove the AAA batteries and the lithium battery simultaneously.

3. Replace all four AAA alkaline batteries simultaneously. Or, replace the lithium battery.

   - To replace the AAA alkaline batteries, remove all four discharged AAA batteries and install new ones according to the polarity (+ and -) diagram in the battery compartment.

   - To replace the lithium battery, remove the screw from the lithium-battery cover, and then remove the cover. Install the new battery, + side up. Replace the cover and secure it with the screw. Use a CR1616 or CR1620 (or equivalent) lithium battery.

4. Replace the battery compartment cover. Turn the calculator on and adjust the display contrast ([2nd] [▲] or [2nd] [▼]), as necessary.

Battery Precautions

Take these precautions when replacing batteries:

- Do not mix new and used batteries. Do not mix brands or type within brands of batteries.
- Do not mix rechargeable and non-rechargeable batteries.
- Install batteries according to polarity (+ and -) diagrams.
- Do not place non-rechargeable batteries in a battery recharger.
- Do not incinerate batteries.

Texas Instruments (TI) Support and Service

For General Information

Home Page:	education.ti.com
KnowledgeBase and E-mail Inquiries:	education.ti.com/support
Phone:	1.800.TI.CARES (1.800.842. 2737) For U.S., Canada, Mexico, Puerto Rico, and Virgin Islands only
International Information:	education.ti.com/support (Click the International Information link.)

For Technical Support

KnowledgeBase and Support by E-mail:	education.ti.com/support
Phone (not toll-free):	1.972.917.8324

For Product (hardware) Service

Customers in the U.S., Canada, Mexico, Puerto Rico and Virgin Islands: Always contact TI Customer Support before returning a product for service.

All other customers: Refer to the leaflet enclosed with this product (hardware) or contact your local TI retailer/distributor.

Customers in the U.S. and Canada Only

One-Year Limited Warranty for Commercial Electronic Product

This Texas Instruments electronic product warranty extends only to the original purchaser and user of the product.

Warranty Duration. This Texas Instruments electronic product is warranted to the original purchaser for a period of one (1) year from the original purchase date.

Warranty Coverage. This Texas Instruments electronic product is warranted against defective materials and construction. **THIS WARRANTY IS VOID IF THE PRODUCT HAS BEEN DAMAGED BY ACCIDENT OR UNREASONABLE USE, NEGLECT, IMPROPER SERVICE, OR OTHER CAUSES NOT ARISING OUT OF DEFECTS IN MATERIALS OR CONSTRUCTION.**

Warranty Disclaimers. ANY IMPLIED WARRANTIES ARISING OUT OF THIS SALE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE LIMITED IN DURATION TO THE ABOVE ONE-YEAR PERIOD. TEXAS INSTRUMENTS SHALL NOT BE LIABLE FOR LOSS OF USE OF THE PRODUCT OR OTHER INCIDENTAL OR CONSEQUENTIAL COSTS, EXPENSES, OR DAMAGES INCURRED BY THE CONSUMER OR ANY OTHER USER.

Some states/provinces do not allow the exclusion or limitation of implied warranties or consequential damages, so the above limitations or exclusions may not apply to you.

Legal Remedies. This warranty gives you specific legal rights, and you may also have other rights that vary from state to state or province to province.

Warranty Performance. During the above one (1) year warranty period, your defective product will be either repaired or replaced with a reconditioned model of an equivalent quality (at TI's option) when the product is returned, postage prepaid, to Texas Instruments Service Facility. The warranty of the repaired or replacement unit will continue for the warranty of the original unit or six (6) months, whichever is longer. Other than the postage requirement, no charge will be made for such repair and/or replacement. TI strongly recommends that you insure the product for value prior to mailing.

Software. Software is licensed, not sold. TI and its licensors do not warrant that the software will be free from errors or meet your specific requirements. **All software is provided "AS IS."**

Copyright. The software and any documentation supplied with this product are protected by copyright.

Australia & New Zealand Customers only

One-Year Limited Warranty for Commercial Electronic Product

This Texas Instruments electronic product warranty extends only to the original purchaser and user of the product.

Warranty Duration. This Texas Instruments electronic product is warranted to the original purchaser for a period of one (1) year from the original purchase date.

Warranty Coverage. This Texas Instruments electronic product is warranted against defective materials and construction. This warranty is void if the product has been damaged by accident or unreasonable use, neglect, improper service, or other causes not arising out of defects in materials or construction.

Warranty Disclaimers. Any implied warranties arising out of this sale, including but not limited to the implied warranties of merchantability and fitness for a particular purpose, are limited in duration to the above one-year period. Texas Instruments shall not be liable for loss of use of the product or other incidental or consequential costs, expenses, or damages incurred by the consumer or any other user.

Some jurisdictions do not allow the exclusion or limitation of implied warranties or consequential damages, so the above limitations or exclusions may not apply to you.

Legal Remedies. This warranty gives you specific legal rights, and you may also have other rights that vary from jurisdiction to jurisdiction.

Warranty Performance. During the above one (1) year warranty period, your defective product will be either repaired or replaced with a new or reconditioned model of an equivalent quality (at TI’s option) when the product is returned to the original point of purchase. The repaired or replacement unit will continue for the warranty of the original unit or six (6) months, whichever is longer. Other than your cost to return the product, no charge will be made for such repair and/or replacement. TI strongly recommends that you insure the product for value if you mail it.

Software. Software is licensed, not sold. TI and its licensors do not warrant that the software will be free from errors or meet your specific requirements. All software is provided “AS IS.”

Copyright. The software and any documentation supplied with this product are protected by copyright.

All Customers outside the U.S. and Canada

For information about the length and terms of the warranty, refer to your package and/or to the warranty statement enclosed with this product, or contact your local Texas Instruments retailer/distributor.

Index

–C–

–C– (continued)

–D–

–D– (continued)

–D– (continued)

–E–

–G– (continued)

–H–

–I–

–I– (continued)

–L–

–U– (continued)

–V–

–W–

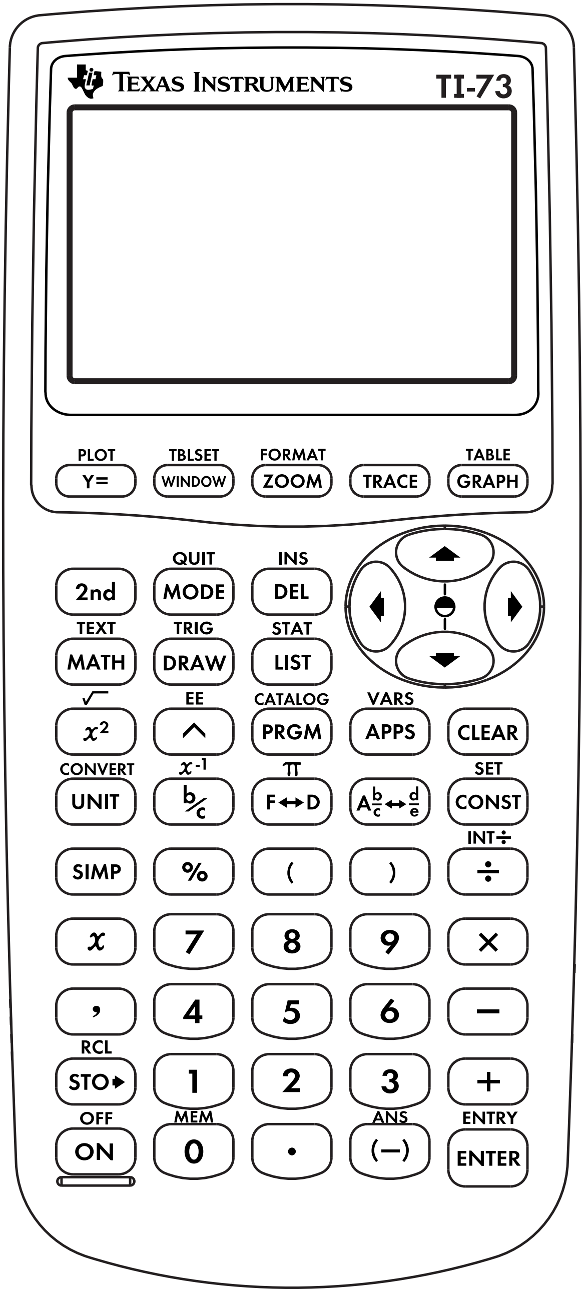
–X–

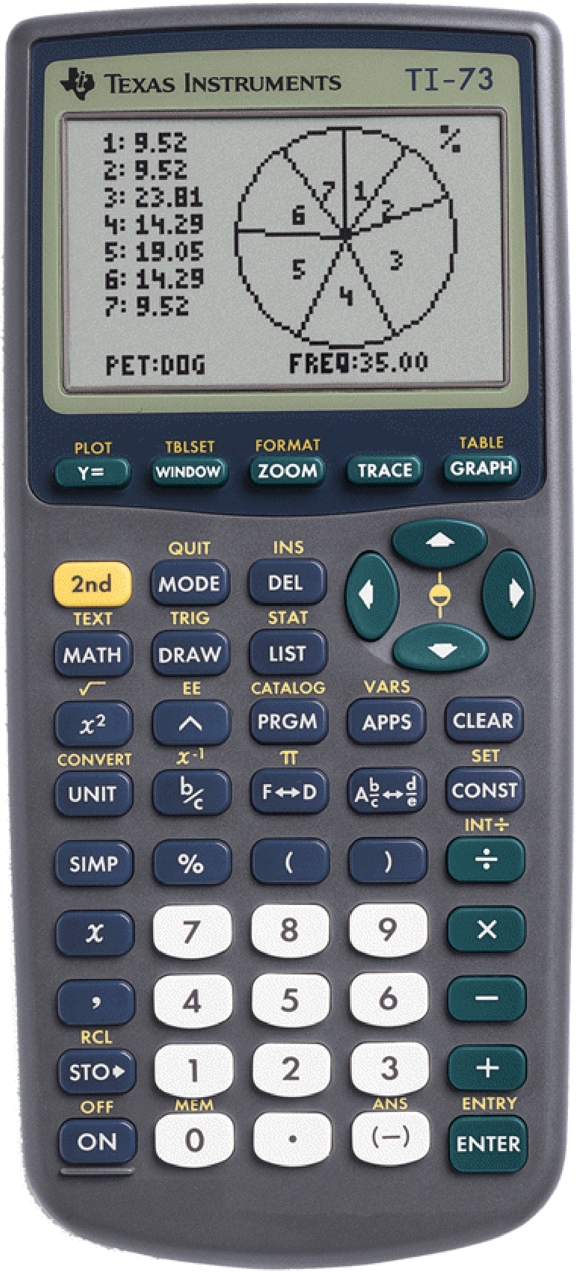
–X– (continued)

–Y–

–Z–

–Z–(continued)


TEXAS INSTRUMENTS
TI-73
PLOT TBLSET FORMAT TABLE
Y= WINDOW ZOOM TRACE GRAPH
QUIT INS
2nd MODE DEL
TEXT TRIG STAT
MATH DRAW LIST
EE CATALOG VARS
PRGM APPS CLEAR
CONVERT π SET
UNIT F↔D CONST
INT÷
SIMP % ( ) ÷
7 8 9 ×
, 4 5 6 −
RCL
STO▶ 1 2 3 +
OFF MEM ANS ENTRY
ON 0 . (−) ENTER


TEXAS INSTRUMENTS
TI-73
1: 9.52
2: 9.52
3: 23.81
4: 14.29
5: 19.05
6: 14.29
7: 9.52
%
PET:DOG
FREQ:35.00
PLOT
Y=
TBLSET
WINDOW
FORMAT
ZOOM
TRACE
TABLE
GRAPH
QUIT
MODE
INS
DEL
2nd
TEXT
MATH
TRIG
DRAW
STAT
LIST
EE
CATALOG
PRGM
VARS
APPS
CLEAR
CONVERT
UNIT
SET
CONST
F↔D
INT÷
SIMP
%
RCL
STO▸
OFF
ON
MEM
ANS
ENTRY
ENTER