Unit 1: Program Basics | **Skill Builder 3: Programs and Functions**

In this third lesson for Unit 1, you will learn the basic difference between a program and a function.

Objectives:

- Write a program and a function that appear to do the same thing
- Explore the differences between a program and a function

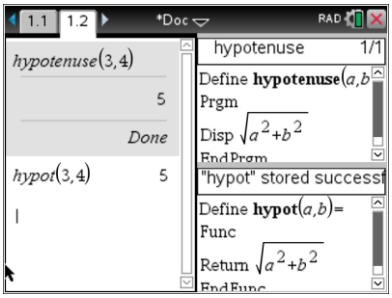
Teacher Tip: This lesson explains the difference between a TI-Nspire™ CX program and user-defined function. This distinction is not unique to the TI-Nspire. In fact, some programming languages rely solely on functions.

The use of local and global variables is discussed in Unit 2.

While a function can use *Disp* statements, this limits the usefulness of the function. For example, it cannot be used for graphing. The use of *Return* in a function allows a function value to be used by other operations.

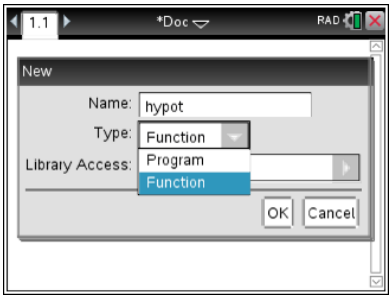
What is a function?

The purpose of a function is to represent, or return a value. In the image to the right, the program **hypotenuse** and the function **hypot** perform the same task. Note the change from **Disp** in the program to **Return** in the function. The function 'represents' a value which can be used by other operations, even graphing.



Creating a Function

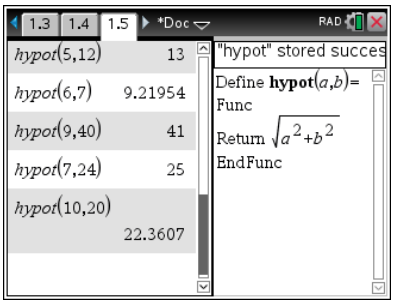
1. In a Calculator app, select **menu > Functions & Programs > Program Editor > New…**.



2. Enter **hypot** for the name, and change the Type to **Function**. Select **OK** or press **enter** to create the new program.

You will see a split page with the Calculator app on the left and the Function Editor on the right. This makes it easy to edit and test the function on the same page.

3. Add the arguments **a,b** inside the parentheses. Inside the function, add the **Return** statement by selecting **menu > Transfers > Return**. Complete the statement by adding the square root of a^2+b^2.
   Return $\sqrt{(a^2 + b^2)}$



4. 'Check Syntax & Store' the program by selecting **menu > Check Syntax & Store > Check Syntax & Store** (or use the shortcut **ctrl+B**).

5. In the Calculator app, test the function. Use the example **hypot(3,4).** Also try some expressions using this function such as:
   2*hypot(5,12) hypot(4,5) + hypot(6,7) (hypot(7,24))^2

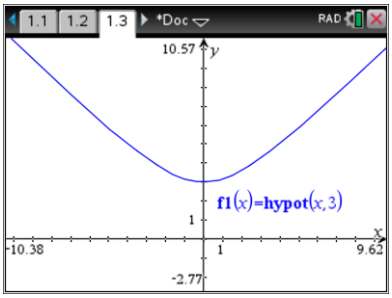
Functions are similar to programs but are not the same. Functions can have many statements and look like other programs in the Program Editor. Functions are different from programs because their purpose is to return a value. The value can be a number, list, string, matrix, or any other built-in data type. Programs are limited in scope (where they can be used). Functions are more versatile because they can be used anywhere a built-in function is used. Programs can only be run (executed) from a Calculator app or inside a Math Box in a Notes app. Functions represent a value that can be used as part of a larger expression.

On a TI-Nspire CX CAS, a function will return an algebraic expression when undefined variables are used as arguments. It could also return a symbolic expression such as √(13) for hypot(2,3).
On a non-CAS TI-Nspire CX, the use of undefined variables produces an error message. For a symbolic expression such as √(13), the function returns an approximate numeric value.

$hypot(x,y)$

$\sqrt{x^2+y^2}$

Graphing the hypot(x,b) Function
User-defined functions have the advantage of being available just like any built-in function.



After you've written and stored the **hypot(a,b)** function above, add a Graphs app, enter f1(x)=**hypot(x,3)**, and press **enter**.

What shape is this function?

Teacher Tip: What shape is the function? **Hyperbola**

Functions have special limitations. Since their sole purpose is to return a value, they are not allowed to impact variables that are not local to the function.
Programs can and do affect other variables and care should be taken when using variables in a program. Arguments (or parameters) are special because they are only used by the program and are not created in the current problem.

The use of local and global variables is discussed in Unit 2.