Counting Loops

Submitted by Andy Lindsay on Thu, 03/21/2013 - 16:07

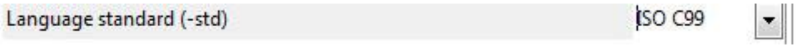
original source: http://learn.parallax.com/propeller-c-start-simple/counting-loops

Lesson edited to work with **Dev-C++** IDE by Jeff La Favre 10/22/15

---

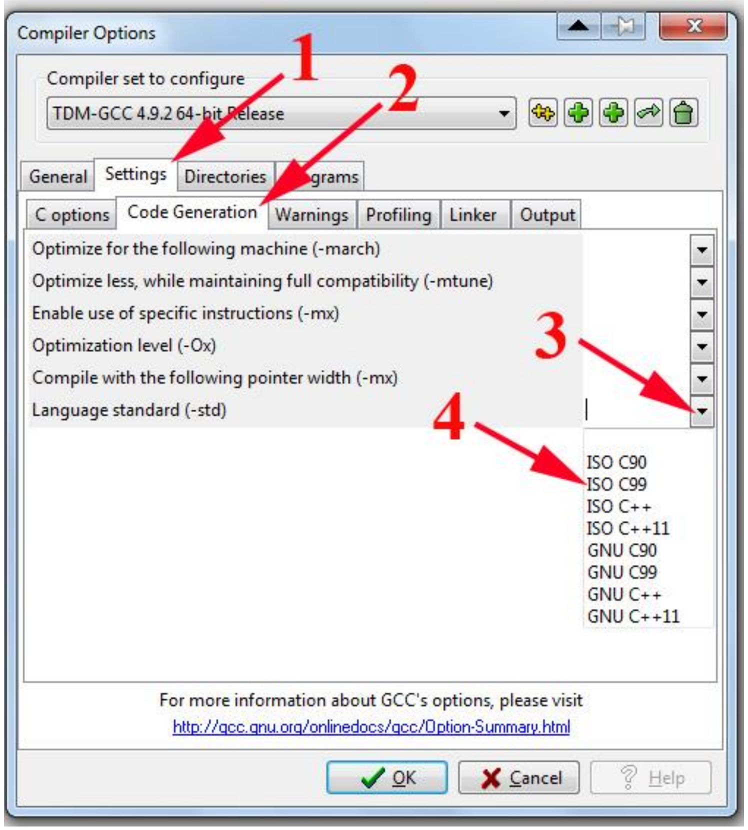
*(Updated 2013-08-08 for SimpleIDE 0.9.4 and its Learn folder's Simple Libraries and Examples) [SimpleIDE is the IDE for use with the robot. This lesson is edited so that we can use **Dev-C++** as the IDE, which does not require a robot – J. La Favre]*

You must change a setting in **Dev-C++** before you can proceed with this lesson. Open the **Tools** menu and select **Compiler Options**. The **Compiler Options** dialog box will open. First, click on the **Settings** tab (step one in illustration to right). Then click the **Code Generation** tab (step 2). Then click the down arrow for **Language standard** (step 3). In the list that opens, click on **ISO C99** (step 4). If you do this correctly, then **ISO C99** will appear next to the drop-down arrow when the list closes like this:



Language standard (-std) ISO C99

Now click the **OK** button and you are finished with the setting change.



Optimize for the following machine (-march)
Optimize less, while maintaining full compatibility (-mtune)
Enable use of specific instructions (-mx)
Optimization level (-Ox)
Compile with the following pointer width (-mx)
Language standard (-std)

For more information about GCC's options, please visit
http://gcc.gnu.org/onlinedocs/gcc/Option-Summary.html

A **for** loop repeats a block of code, but it's more configurable than a **while** loop. A **for** loop is also the preferred tool for counting tasks. **while** loops are better for repeating something "while" a condition is true. In contrast, **for** loops are better for repeating a block a certain number of times. This next program counts to ten with a **for** loop.

- Start **Dev-C++**.
- Open the **File** menu and select **New**. Then select **Source File**.
- Click the mouse in the text window of **Dev-C++** and use the keyboard to enter the following text: **#include <stdio.h>**
- Open the **File** menu and select **Save**, which opens a **Save As** dialog box.
- In the dialog box, open the drop-down labeled **Save as type** and select **c source files(*.c)**. In the **file name** slot enter this name for the file: **count to ten.** At the top of the dialog box there is a **Save in** slot, which determines where the file will be saved. Make sure you know the location where you are saving your file so that you can find it later. Now click the **Save** button to save your program file.
- Copy the text in the box on the next page and paste it into the text window of **Dev-C++** under the first line of text you have already entered. Alternatively, you can enter the text using the keyboard.
- Click the **Save** button to save the code you just pasted or entered with keyboard.
- Run the program by opening the **Execute** menu and selecting **Compile and Run**. If there are no errors in the program, a new program window will open. The program will pause after a line is displayed until you press the Enter key. Then the for loop will run again and print the next line. Keep pressing the Enter key until the program lets you know it is done. Verify that the program display counts up to 10.

```
int main()
{
 for (int n = 1; n <= 10; n++)          // Count to ten
 {
  printf("n = %d\n", n);                // Display name & value of n
  // pause(500);                        //this line in original program has been suppressed with a comment due to lack of support in Dev-C++
   getchar();         //this line is substituted for line above…program will pause here until user presses the Enter key
 }
 printf("All done!");                   // Display all done
}
```

How it Works

The **`for`** statement has three parameters: **`for(start index; condition; increment).`**

- start index – typically a variable that is assigned an initial value where the counting starts
- condition – a condition that has to be true for the loop to repeat
- increment – an operation that should be performed on the **`index`** variable with each loop repetition

In the Count to Ten program, the start index is **`int n = 1`**, a variable declaration that initializes the variable **`n`** to the value `1`. The condition is **`n <= 10`**, which means that the loop will keep repeating while **`n`** is less than or equal to ten. The increment is **`n++`**, which is a shorthand form of **`n = n + 1`**. So with each repetition of the **`for`** loop's code block, the value of **`n`** increases by `1`.

Did You Know?

Increment operators — The `++` operator is called an *increment operator*. Although increment and decrement operators are especially useful in `for` loops, they can be used as shortcuts in many parts of your programs.

```
++    add 1 to a variable
--    subtract 1 from a variable
```

Assignment operators — There are also some useful shortcuts for assigning values to variables. For example, instead of `n = n + 5`, you can use `n += 5`. As you might expect there are, shortcuts for subtraction `-=`, multiplication `*=`, division `/=`, and modulus (remainder) `%=`. These are called the *assignment* forms of the operators.

Try This

Here is a `for` loop that will count to 200 in steps of 5, just like the `while` loop you experimented with earlier.

- Open the **File** menu and select **Save As** to save your program as **count to 200 by 5s**.
- Modify the `for` loop so that it looks like the code below.

```
for (int n = 0; n <= 200; n = n + 5)          // modify the for line to look like this
```

- Verify that it counts to 200 in steps of 5.

Your Turn

- Modify your program so that it counts up to 200 in steps of 5, then down to 0 in steps of 10.