Lab 8

For Loops

Learning Objectives

There is also a "for" loop in C, which says: initialize a variable to some base value, execute the contents of the loop while some condition is true, and after each run of the loop perform some operation.

After completing this lab you will be able to write repetition statements using for loops in C.

Basic Lab Instructions!

- Talk to your classmates for help.
- You may want to bring your textbook to future labs to look up syntax and examples.
- Stuck? Confused? Have a question? Ask a TA/Lab Engineer for help, or look at the book or past lecture slides.
- Complete as many problems as you can within the allotted time. You don't need to keep working on these exercises after you leave the lab.
- Before you leave today, make sure to check in with one of the Lab Engineers/TAs in the lab to get credit for your work.

Lab Tasks

Creating a New Project

After completing this section you will have created a project for today's lab.

1. Open the solution CS110Labs you created in the lab last week.
2. Add a new project called Lab07 within the solution CS110Labs. Set the project as startup project (right-click the project in Solution Explorer). STOP HERE! start with the lab tasks, and refer to the following steps accordingly.
3. Click File in the top menu bar of Visual Studio, and then select the Add New Item option (Add Existing Item option depending on the lab task).
4. Select C++ File
5. Type in the code listed.
6. Compile (Build Lab07) the program and execute it (Execute without Debugging).

Task #1: Counting with a For Loop

As you saw in Counting with a While Loop, a while loop can be used to to make something happen an exact number of times.

However, this isn't our best choice. while loops are designed to keep going as long as something is true. But if we know in advance how many times we want to do something, Java has a special kind of loop designed just for making a variable change values: the for loop.

Type in the following code (CountingFor.c), and get it to compile. Then answer the questions down below.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int main()
{
  char message[80];

  printf( "Type in a message, and I'll display it five times.\n" );
  printf( "Message: " );
  gets( message );

  for ( int n = 1 ; n <= 5 ; n = n+1 )
  {
    printf( "%d. %s\n", n, message );
  }

  return EXIT_SUCCESS;
}
```

for loops are best when we know in advance how many times we want to do something.

- ❖ Do this ten times.
- ❖ Do this five times.
- ❖ Pick a random number, and do it that many times.
- ❖ Take this list of items, and do it one time for each item in the list.

On the other hand, while loops are best for repeating as long as something is true:

- ❖ Keep going as long as they haven't guessed it.
- ❖ Keep going as long as you haven't got doubles.
- ❖ Keep going as long as they keep typing in a negative number.
- ❖ Keep going as long as they haven't typed in a zero.

What You Should See

```
Type in a message, and I'll display it five times.
Message: Perhaps later.
1. Perhaps later.
2. Perhaps later.
3. Perhaps later.
4. Perhaps later.
5. Perhaps later.
```

What You Should Do on Your Own

Assignments turned in without these things will receive no credit.

- ❖ What does n = n+1 do? Remove it and see what happens. (Then put it back.)
- ❖ What does int n = 1 do? Remove it and see what happens. (Then put it back.)
- ❖ Change the code so that the loop repeats ten times instead of five.
- ❖ See if you can change the for loop so that the message starts at 2 and counts by twos, like so:

```
Type in a message, and I'll display it five times.
Message: Perhaps later.
2. Perhaps later.
4. Perhaps later.
6. Perhaps later.
8. Perhaps later.
10. Perhaps later.
```

Task #2: Eight Times

Write a program EightTimes.c that prints the important phrase "I am what I am." on the screen eight times. Use a for loop to do it.

```
Type in a message, and I'll display it eight times.
Message: I am what I am.
I am what I am.
I am what I am.
I am what I am.
I am what I am.
I am what I am.
I am what I am.
I am what I am.
I am what I am.
```

If you want, you can number the lines of output like so:

```
Type in a message, and I'll display it eight times.
Message: I am what I am.
1. I am what I am.
2. I am what I am.
3. I am what I am.
4. I am what I am.
5. I am what I am.
6. I am what I am.
7. I am what I am.
8. I am what I am.
```

Task #3: Counting Machine Revisited

Write a program CountingMachineRevisited.c that gets three integers from the user. Count from the first number to the second number in increments of the third number. Use a for loop to do it.

```
Count from: 4
Count to  : 13
Count by  : 3

4 7 10 13
```

```
Count from: 5
Count to  : 20
Count by  : 5

5 10 15 20
```

```
Count from: 2
Count to  : 10
Count by  : 1

2 3 4 5 6 7 8 9 10
```

Task #4: Counting By Halves

Write a program CountingByHalves.c that uses a for loop. With the loop, make the variable x go from -10 to 10, counting by 0.5. (This means that x can't be an int.)

```
x
------
-10.0
-9.5
-9.0
-8.5
-8.0
...
9.0
9.5
10.0
```

Task #5: Xs and Ys

Write another program XsAndYs.c that uses a for loop. With the loop, make the variable x go from -10 to 10, counting by 0.5. (This means that x can't be an int.)

Inside the body of the loop, make another variable y become the current value of x squared. Then display the current values of both x and y.

```
 x       y
-----------------
-10.0   100.00
-9.5    90.25
-9.0    81.00
-8.5    72.25
-8.0    64.00
...
9.0     81.00
9.5     90.25
10.0    100.00
```

Task #6: Noticing Even Numbers

Write a program NoticingEvenNumbers.c that uses a for loop to display all the numbers from 1 to 20, marking those which are even (divisible by two). It should use modulus by 2: if the remainder is zero, it's divisible by 2.

```
1
2 <
3
4 <
5
```

```
6 <
7
8 <
9
10 <
11
12 <
13
14 <
15
16 <
17
18 <
19
20 <
```

Task #7: Fizz Buzz

Write a program FizzBuzz.c that prints the numbers from 1 to 100. But for multiples of three print "Fizz" instead of the number and for the multiples of five print "Buzz". For numbers which are multiples of both three and five print "FizzBuzz".

```
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
14
FizzBuzz
16
17
Fizz
19
Buzz
  ...
97
98
Fizz
Buzz
```

Task #8: Letter at a Time

Did you know that using a loop, you can examine a String one letter at a time? Open and examine the code in LetterAtATime.c.

What You Should See

```
What is your message? Are we having fun yet

Your message is 21 characters long.
The first character is at position 0 and is 'A'.
The last character is at position 20 and is 't'.

Here are all the characters, one at a time:
```

```
 0 - 'A'
 1 - 'r'
 2 - 'e'
 3 - ' '
 4 - 'w'
 5 - 'e'
 6 - ' '
 7 - 'h'
 8 - 'a'
 9 - 'v'
 10 - 'i'
 11 - 'n'
 12 - 'g'
 13 - ' '
 14 - 'f'
 15 - 'u'
 16 - 'n'
 17 - ' '
 18 - 'y'
 19 - 'e'
 20 - 't'

Your message contains the letter 'a' 2 times.
```

What You Should Do on Your Own

Assignments turned in without these things will receive half credit or less.

- The for loop is defined so that it repeats as long as i < strlen(message). Try changing it to <=. What happens? Answer in a comment, then change it back.
- If a string variable contains the value "box", what is its length? What is the position of the last character (the 'x')?
- So, why does the for loop repeat as long as i < strlen(message) instead of i <= strlen(message)?
- Currently the code prints out the number of 'a's in the message. Change it so that it prints out the number of vowels (a A e E i I o O u U).

Task #9: Adding Values with a For Loop

Write a program AddingValuesForLoop.c that gets an integer from the user. Add up all the numbers from 1 to that number, and display the total. Use a for loop to do it.

You have done something like this before.

```
Number: 5

1 2 3 4 5
The sum is 15.

Number: 8
1 2 3 4 5 6 7 8
The sum is 36.
```

Task #10: Baby Blackjack

Write a program BabyBlackjack.c that allows a human user to play a single hand of "blackjack" against a dealer.

- ❖ Pick two values from 1-10 for the player. These are the player's "cards".
- ❖ Pick two more values from 1-10 for the dealer.
- ❖ Whoever has the highest total is the winner.

```
Baby Blackjack!

You drew 6 and 5.
Your total is 11.

The dealer has 7 and 3.
Dealer's total is 10.

YOU WIN!
```

Hand in

Hand in the word file for this lab at the appropriate location on the blackboard system at LMS. You should hand in a single file named Lab_8_<your reg. No. XXX without angle brackets>.doc(x) that contains the following.

1. All completed C source files representing the work accomplished for this lab: CountingFor.c; EightTimes.c; CountingMachineRevisited.c; CountingByHalves.c; XsAndYs.c; NoticingEvenNumbers.c; FizzBuzz.c; LetterAtATime.c; AddingValuesForLoop.c; and BabyBlackjack.c.
2. A paragraph at the end that includes a) a brief explanation of the lab, and b) any comments, or suggestions.

To Receive Credit

1. By showing up on time for lab, working on the lab solution, and staying to the end of the class period, only then you can receive full credit for the lab assignment.
2. Comment your program heavily. Intelligent comments and a clean, readable formatting of your code account for 20% of your grade.
3. In-class lab time is not intended as free time for working on your program assignments. Only if you have completely solved the lab assignment, including all challenges, and have had your work checked off for completeness by your TA/Lab Engineer should you begin the program assignment.