

Loops

10-1-2003

Opening Discussion

- What did we talk about last class?
- Do you have any questions about the assignment?
- A good use of the switch statement:
 - Imagine you want one block of code to run when $x=2, 5, 13,$ or 14 and a different set for $x=1, 4, 7, 10,$ and 11 .
 - Let's do a quick code example.

Motivation

- At that point you actually have the ability to theoretically calculate anything. Recursion is powerful enough. However, in the C language it is not always efficient and it requires too much code for simple repetition.
- The biggest problem with recursion in C is that each call adds to the stack and so doing something MANY times uses a lot of memory.

Repeated Execution, Looping

- To get around this limitation we need a new construct, one that allows us to have a piece of code execute several times without invoking a new function call.
- There are three basic loop structures in C and we will talk about the first one today.
- All loops in C have 3 basic parts: initialization, conditional, and iterator.

The while Loop

- The most basic loop in C is the while loop. Here is the basic syntax of the while loop.

```
while(condition) {  
    statements;  
}
```
- When this loop is encountered, the statements are executed repeatedly as long as the condition is true.
- This leaves out some essential things you need in your head to write this though.

A Better Mental Picture

- Here is the way you should think of a while loop.

```
initializer-code;  
while(condition) {  
    body-statements;  
    iterator;  
}
```
- The initializer sets things up for the loop and the iterator makes sure that you "move to the next case".

A Pre-Check Loop

- The while loop is what we called a pre-check loop. In these types of loops, the condition is checked before the body of the loop is executed. This really only matters the first time through.
- With a loop like this, it is possible that the body will never be executed if the condition is false when the program first gets to the loop.

Tracing Code

- One of the skills that is critical in this class is the ability to track through the code that you write and figure out what it will do without the computer. Not only will I ask you to do so on quizzes and tests, but you need it to debug code and if you can't trace it you will probably find it very hard to even construct code in the first place.

Code

- Now we will write some code that uses a while loop.

Minute Essay

- Write a loop that will read integers from input and adds them all up, then stops when the user inputs a value of 0 and prints the sum.
- Remember to submit your code for assignment #3 today.
