Scoping

9-17-2001

Opening Discussion

- What did we talk about last class?
- Functions can not occur inside other functions in C++.
- Today we will continue our discussion of function calls. We will still deal with the topics of last class, but we will also move on to look at the scoping of variables. From the reading who can tell me what I mean when I talk about the scope of a variable?

A min Function

- Between the two classes only about 50% of the minute essays had correct declarations for a min function. So lets think about it here again. You want a function that returns the smaller of two ints.
- The fact that it returns an int is straightforward. It needs two arguments though because it is the smaller of TWO numbers.

```
int min(int num1,int num2);
```

How Function Calls Work

- In the memory of the machine function calls are implemented with what is called a stack. We'll be using simpler stack data structures later in the semester.
- When a function call is made, the memory that is needed for it to function is allocated at the end of the stack. That block of memory will include space for arguments and local variables.
- When the function exits the stack is "popped" and the memory that was used by that function is freed up.

Scope of a Variable

- When talking about functions and arguments it becomes very important to understand the concept of scope.
- The scope of a variable is the region of code over which that variable exists. Equivalently it is the region over which the variable can be used in the program.
- Note that his concept is only important because they don't exist everywhere just because you declared them.

Blocks and Scoping

- When a variable is declared in a block, the scope of that variable is between the line where it is declared and the end of the block. They are called local variables.
- A variable name is bound to the closest declaration with that name.
- Global variables are declared outside of all blocks and exist anywhere in the program. Using globals is generally considered poor coding practice.

Arguments and Static Variables

- The arguments to a function have a scope through the full function, but not outside. When you are dealing with pass by reference the scope is only for the name.
- The keyword static means many things in C++ (this is not a good thing). In this context when a local variable is declared static it means that it retains its value between calls to the function. This is not normally the case.

Example Program

- Now we will look at an example program of an inventory calculator. It prompts us for the names, prices, and values of different items. It assigns a unique number to each item and tallies up the total inventory value.
- It does involve one loop structure that we haven't learned about yet but don't get bogged down with that.

Minute Essay

- What did we talk about today? Why is it important?
- On Wednesday we will conclude our discussion of chapter 3. I'm not strongly attached to the rest of the material in the chapter. What topics would you like me to spend that time discussing to help you with your understanding of functions and how to write/use them?