Functions

9-7-2006

Opening Discussion

- What did we talk about last class?
- Do you have any questions about the reading?
- Minute essay from last class.
  - int num1=5,num2=3;
  - num1+=7;
  - num2=(num1++);
  - num1%=5;

Modulus

- C includes some operators you might not be familiar with. One of these is the modulus operator. It works on integers and returns the remainder after division. For example, num%2 would be 1 for odd numbers and 0 for even numbers.
- Modulus is a remarkably useful operator when doing math on computers.
- Let's put some code into our program so that we can enter our average speed and get the time it will take to make the trip in hours, minutes, and seconds.

Bitwise Operators

- There are some operators that appear in the front cover of the book that aren't covered in chapter 3, but that relate do our discussion of binary numbers and binary arithmetic.
- C has certain bitwise operators that allow you to play with the bits in numbers.
  - ~ : bitwise not or ones complement
  - << : left shift
  - >> : right shift
  - & : bitwise and
  - ^ : bitwise xor
  - | : bitwise or
- A common use of these is packing colors into an ARGB int.

Type Casting and Type Conversion

- One type of expression that might seem odd to you is the type casting operator. You use this when you want to force the type of an expression to be something.
- An example is when you want to divide two values and you have them stored in ints, but you want the fractional value.
- C also does implicit type conversions when you operate on two expressions with different types.

Problem Decomposition

- One of our majors goals in programming is to take large problems and break them into smaller pieces that can be solved more easily.
- A standard method of computer science is top-down problem decomposition. Here we take the top level problem and break it into pieces. Often we write a function and simply specify which functions it will call. We repeat this in those functions until we get to a level where we can solve the problem simply.
- It is also possible to do some problem solving bottom-up. This typically requires more knowledge of what the solution will look like.

Function Writing Syntax

- Functions in C have the syntax of a return type, followed by a name, followed by a list of parameters in parentheses.
- Functions must be declared before they are used. You can decide if you define them before or after they are used. A declaration ends with a semicolon. A definition has a compound statement in curly braces.
- Let's take our gas program and break it into functions. We will exaggerate things here and build more small functions than we might normally just to demonstrate the concepts.

Standard Functions

- The C standard libraries have a lot of different functions in them that are part of different include files.
- Most math functions are in the math library, math.h. Your book lists those and also talks about random numbers.
- Remember that you can use man to get information about standard C functions.

Scope

- The scope of something is a programming language is the section of code over which it can be used.
- C has rather simple scoping rules. There is local scope (inside the closest curly braces) and global scope (through the full program).
- This gets a bit more complex when multiple files are involved, but not too much.
- Limiting scope is a good thing because it reduces the number of lines you have to look for certain types of bugs on. This is because you can't screw up something you aren't allowed to touch.

Minute Essay

- Write a function that accepts integers for red, green, and blue and returns an integer that is those values packed into a single int.
- Remember to turn in assignment #1 by midnight tonight.
- Have a good weekend. I'll be posting the description of assignment #2 soon.