Programming Style and Errors

9-12-2001

Opening Discussion

- What did we talk about last class?
- Do you have any analysis type questions on the homework problem?
- vi and Linux are not on the tests, but you do need to know them to complete the assignments.
- endv vs. \n
- #include statements (to .h or not)
- Exponents in C++ use pow function.

FTP for Linux

- The general way to move files to and from a Linux machine is with a program called FTP. FTP comes with every Unix and Windows OS. You can also get nice graphical versions of FTP for Windows.
- "open machine-name" opens a connection
- "close" closes a connection
- "cd" to change directory on remote machine
- "lcd" changes your local directory
- "get" and "put" move a specified file, "mget" and "mput" allow wildcards.
- "quit" to exit
**Type Casting**

- One topic that we didn’t cover on Monday was that of type casting. C++ does not like it when you try to assign a double value to an integer variable. If you really want to do this you have to “cast” the double to an integer type.

  ```cpp
  int i;
  double x=2.5;
  i=2*x;   // Generates warning
  i=(int)2*x; // No warning.
  ```

**Solving a Simple Problem**

- We are going to go through the solution of creating a program to calculate grades for this class...almost.
- We will start off by doing the analysis of the problem. What elements are involved in your grade for this class and how?
- Now we can do the design. Is there something we will have to leave out because we don’t know enough about C++ to do it yet? Does this alter our analysis?

**Style in the Code**

- Page 69 in the text has a list of recommended things you can should do to make your code both readable and understandable. I expect you to follow all of them.
- When we talk about loop and if-else structures we will talk more about how to arrange curly braces and indent the code for better readability.
Errors

- Syntax errors - These are the easy errors because the compiler finds them for you and tells you about them. The way it tells you can be cryptic at times though.
- Runtime errors - This is where the code compiles properly but when it executes it crashes. Something like divide by zero.
- Logic errors - These are the real difficult ones to locate. Often originate in design.

---

Minute Essay

- What did we talk about today? Do you now feel confident to write a simple program in C++ (such as the one you have been assigned)?
- Read chapter 3 for Friday. You should certainly get through 3.2 to help you understand what will be discussed.