Jim's Coding Commandments (for class labs)

1) **Never** use global variables -- they allow your code to do weird things that you can never figure out. We stopped using them back in the 1960s.

2) **Never** use goto statements -- there is a reason we have structured programming. Read Dyjkstra and learn. (Yes, there are times that they are the best way to solve a problem, but not in any of these labs.)

3) **Never** use while(true) with break statements -- it makes it really hard to figure out where and why you ended the loop (especially when it is 100s of lines of code). Instead you should use while (not ending condition).

4) **Never** use break outside of switch statements. In the working world there are exceptions to this, in this classroom there are not.

5) **Never** use continue. It, like break, is a shortcut, and is too easy to abuse when writing bad code. The real world may accept it when appropriate, in this classroom, there is no appropriate use of it.

6) **Never** use recursion except in functions designed to be recursive (base case with recursive call domain moving toward base case).

7) **Always** use global constants -- if it is a constant, make it global so it can be shared

8) **Always** reuse code -- if you can write some function that solves a problem, keep reusing it -- that is why we have functions. If you have two copies of the same functionality, put it in a function and call it.

9) **Read** the instructions -- that is why they are there. Clients will expect you to implement the programs they request. We expect it too.

10) **Comment** your code -- you might be the poor sucker who has to redo it in two years.

11) **Indent** appropriately -- you might be the poor sucker who has to redo it in two years.

12) **Ask** when you need help – we can't read your minds