

## C201

### **RULES FOR PROGRAMMING ASSIGNMENTS**

- 1) Programs/Homeworks are due at the beginning of class on the assigned due dates, put them on front desk when you come in. If you cannot make it to class have a friend turn it in for you.
- 2) Homework turned in one class period early will receive an extra three points. Homework turned in one class period late will have three points deducted, and those turned in two class periods late will have six points deducted. **No homework accepted after this.**
- 3) Incomplete homework will be accepted, but get as much done as possible along with a functional written algorithm and an explanation of the program's "current state" and a suggested test strategy.
- 4) You may work with someone else on the **first assignment only**, and make sure you put persons name on your homework as well. Here you will be able to turn in essentially the same program, each must turn in their own. The rest of the homeworks you must do the program by yourself, but you may still work on the algorithms with someone else, but still put persons name on your homework. Remember different programs will come from the same algorithm. **If two or more people turn in essentially the same program then each program will only be worth 50% of total grade.**
- 5) Make sure all programs are commented correctly, with enough comments for clarity of operation, see example programs that will follow. **DON'T COMMENT THE OBVIOUS!!**
- 6) Programs will be graded as follows:
  - A. Comments and Program Style ( 25% )

Each program should have comments which introduce the main program and each function. Comments should also be used alongside lines of code that might be hard to follow. Program Style includes using good variable names and proper formatting.
  - B. Testing and Program Output ( 25% )

Hand in copy of program output with different cases that show the program is working correctly. In testing the logic of a program, quantity is not a substitute for quality. Program output should always contain a brief message explaining what the results are, if needed use hand written messages, no messages needed for obvious output.
  - C. Program Design ( 25% )

This includes things such as the design of main function and other functions, and the proper use of these functions.
  - D. Algorithms ( 25% )

A working program will get most of the points, but unnecessarily long programs with poor algorithms will not receive full credit.