CS 1337 Programming Style Guidelines

Programming style is a way of using statements, identifiers, comments, and whitespace to create programs that are easier to read, understand, debug, and maintain.

You are required to follow the style guidelines given below in this course.

General Style Guidelines:

- Provide appropriate comments in your source code. Document the purpose of the program. Explain sections of the program including classes, functions, and complex code segments or calculations.
- Place no more than one statement on a line.
- Use whitespace characters (blank lines, tabs, and spaces) to visually arrange the program in a way that is both pleasing and readable.
  - Put blank lines both above and below comments to separate them from surrounding code.
  - Put blank lines both above and below distinct program components (function definitions, class definitions, etc.)
  - Use a blank line to separate variable definitions from other types of statements.
  - Indent the statements in a block (inside of a set of braces) one level from the surrounding code.
  - Put a space on both the left and right sides of an operator to make the operator stand out visually.
- Use meaningful identifiers. Give your variables, constants, functions, and classes meaningful names.
- Follow naming conventions when choosing the names of variables, constants, functions, and classes. For example, variable and function names should begin with a lowercase letter, class names should begin with a capital letter, and constants should be in all caps.

Decision Style:

- Begin the conditionally executed statement or block on the line below the if or else in a decision structure.
- Indent the conditionally executed statement(s) one level from the key words if or else.
- The else corresponding to an if should be at the same level of indentation as the if in all decision statements except when you are coding a multi-way decision using an if/else if structure.
- Do not use a return or exit to avoid coding the else part of a decision.
Loop Style:

- Begin the body of a while or for loop on the line below the loop header.
- Begin the body of a do-while loop on the line below the key word do.
- Indent the statements in the body of a loop one level from the key word that marks the beginning of the loop.
- Write your loops so that it is clear from the loop continuation expression when the loop iterates and when it terminates
  - Do not use a break to exit a loop.
  - Do not use continue in your loops.
  - Do not use a return or exit to terminate the iterations of a loop.
  - Do not put statements in the body of a for loop that change the value of the variable(s) used to control the iterations of the loop.

Function Style:

- Document your function using comments. Describe the purpose of the function, the parameters, and the return value of the method.
- Give your functions meaningful names.
- Indent the statements in the body of a function one level from the header.
- Define variables locally (inside some method). Send data items to other methods using arguments/parameters.
- Pass by value whenever possible.

Class Style:

- Document your class using comments. Describe the purpose of the class.
- Give your classes meaningful names.
- Indent the statements in the body of a class one level.
- All variables should be private.

This document is not guaranteed to be complete. Make sure you follow the formatting conventions described in the textbook and lectures.