

Style Requirements for CPS 150 Programs

1. Each **file** in your project must begin with a single-line comment giving the filename, followed by a “flowerbox” comment identifying the programmer (your name), the date due, and the purpose of the program. Use the following format:

```
// filename.cpp
```

```
/******  
* Author: [your name] *  
* Assignment: 1 *  
* Class: CPS 150-N1 *  
* Instructor: Dr. Saverio Perugini *  
* Date due: [date the current assignment is due] *  
* *  
* This program accepts a series of numbers from standard input, displays them, *  
* calculates the average and standard deviation, and displays the result. *  
* *  
*****/
```

HINT: You can use notepad or any other text editor to type in a “template” of the above comment header, then copy and paste it into your programs. You will then just need to change the assignment number, date due, and purpose sections as needed.

2. Section 2.5 of PSwC++ discusses conventions to be followed for formatting code. The rules are summarized here:
 - Use meaningful names for variables, function names, etc. This means using names that describe the purpose of what they represent.
 - Be sure that variables are declared to be of the correct data type.
 - Be sure that variables are initialized before the program attempts to use their value. This can be done when the variable is declared or with an assignment statement before the variable is first used.
 - Use enough parentheses in arithmetic expressions to make the order of operations clear.
 - Always include a prompt line in a program whenever the user is expected to enter data from the keyboard and always echo the user’s input, unless stated otherwise.
 - Almost all number constants in a program should be given meaningful names that can be

used in place of the numbers and use the modifier `const` in a declaration of a constant.

- All code within a block should be indented; e.g., indent all code within functions (including `main`), within `if` statements, and within loops. This often means several levels of indenting within a block of code. Follow the pattern used in your textbook. Indenting should increase consistently (e.g., 3 spaces or 1 tab stop) with program level.
- My preference for curly braces `{}` is to always place the opening brace on the same line as the block it opens, and indent all lines of code between each brace. This makes it easy to see where blocks of code, such as loops, begin and end. You also may place each brace on a line by itself. You may use either of these styles, but be consistent throughout your program.
- Insert comments to explain major subsections or any unclear part of a program.
 - Precede single-line comments with `//`
 - Use `/* ... */` for multi-line comments (but do not use multi-line comments where they affect code readability).

Other grading criteria:

Style is just one portion of your assignment grade. To receive full credit for a programming assignment, your program also must:

- execute under all (reasonable and pathological) test conditions
- meet the requirements of the problem
- use an algorithm which is efficient, and that can be followed with reasonable ease