Here is the styling guide to follow or at least the parts important:

7. Functions: Applies to labs 6-10

Functions should:

1. Have a descriptive name for the function and the function parameters.
2. Be short, typically less than 20 lines and no more than 40.
3. Do one task.  For example, you would not want a function which both a) returns the total number of words in a document and b) capitalizes first letters of all the words.  Those should be done by separate functions, making them simpler and more general to use in the future.
4. Be structured to avoid duplication of code.  Thus, you should not have places in your lab where you have duplicated snippets of code many times, which would have been better handled by a function call each time.
5. If you pass in a large data structure by reference (e.g. vector, array), which will not be updated, then pass it in as const.  You should name these formal parameters as with any normal variable and NOT with all CAPS as that type of constant is somewhat different.

With respect to items 2 and 3,  note that one issue here is the complexity of the code. Generally shorter code is simpler to write and easier to understand than longer code. However, code that is a bit longer and simple to understand is better than being shorter but difficult to understand.  Also, do not try to shorten code just by removing the whitespace used to make the code more readable (see the Format section).

Constants and variables are declared at the beginning of their enclosing function (including main), or at the top of the program when the constant is global.  You will not use global variables.  If the constant is local it should just be declared in that function.

6. Function names are CamelCase with uppercase first, or with lower case letters and words separated by underscores.

    DO:

        NumberOfCharacters(string inputString)

        number\_of\_characters(string input\_string)

    DON’T:

        nChars(string s)

        Num\_Characters(string str)

Make sure that your naming of functions and parameters are sufficiently clear that you rarely need a comment header to describe the function and parameters.

8. Objects and Classes:  Applies to labs 8-10

1. Class naming is CamelCase with uppercase first.  Member functions follow the same naming rules as functions.
2. Always create your own default constructor for each class, and create a parameterized constructor(s) when needed (usually).  You may combine these into one constructor function by creating a default constructor with default parameters.
3. Any member function which will not mutate data members should be declared as const.
4. All data members and helper functions should be private.