

# Foundations of Computer Programming

## COSC 181-0001

### Programming Assignment 1

**Due Date: Sept 3, 2008 (before class)**

---

#### Instructions

The instructions below give important information about how the program should be submitted and written:

- Complete the programming assignment as given in the instructions below.
- All programs will be graded using the DevC++ compiler.
- Email your assignment using your UVA Wise mail account to dar5p@mcs.uvawise.edu before class on the due date.
- Your C++ program (the .cpp file) should be sent as a file attached to the email message.
- Your program file should be named using your last name and first name separated by underscores. So “John Doe” should name his program file “Doe\_John.cpp.” This will make grading your programs much easier, so you will lose points if you don’t get this right. And you will not receive any points if I cannot determine who submitted the program.
- Your name and “Programming Assignment #1” (or whichever program number this is) should appear on the “Subject:” line of your email message. Points might be deducted if this is not the case.
- Make sure to “Cc:” yourself on this mail message, then verify that what you sent is indeed a valid C++ file, and that it is the one you intended to send.
- Late programs will not be accepted.
- If you send more than one email, only the last one you send will be graded.
- No group work is permitted.
- Once you sit down at a computer to work on an assignment, you should be doing your own work. (Review the syllabus if necessary).

In addition, the following instructions give important information about how the program should be formatted:

- You should have identifying header-block comments (name, student number, course and section number, program number) along with comments describing the purpose of the program at the top of your source program.

- Each program should also have several internal comments to explain particular sections of code that otherwise might be somewhat confusing to whoever is reading and grading your program.
- If your program's indentation (or lack of it) makes your program too difficult to grade, points will be deducted.
- Similarly, if your choice of variable names makes your program too difficult to grade, points will be deducted.
- You must not change the order of the input, add extra input values, or otherwise modify the input stream in any manner.
- You must not change the output, or add any extra output to the program you turn in. While using many extra "cout" statements is an excellent way to debug a program, be sure that all the extra output statements are either removed or commented out before you turn in your program. Extra output can make your program much more difficult to grade, and anytime this happens points might be deducted.

## Programming Assignment

Write a program that prompts the user for 20 lower-case characters, one at a time. For instance, the screen should look like:

```
Enter Character 1:  
b  
Enter Character 2:  
u  
Enter Character 3:
```

so forth and so on all the way to the twentieth character.

After the twentieth character has been entered, the program should print out how many total vowels (a,e,i,o, or u) have been entered as well as how many of each vowel was entered. For instance, the final program output might read:

```
Total number of vowels: 6  
Total number of 'a's = 1  
Total number of 'e's = 2  
Total number of 'i's = 1  
Total number of 'o's = 1  
Total number of 'u's = 1
```

Please note, your program will be tested with multiple inputs. You should ensure that your program works with any legitimate input and that no bugs are present.