CS&E 1222
Lab 5 – While Loop Lab Assignment – 20 points

- ✓ The *lab* must be accomplished solely by you:
  - ➢ DO NOT look at anyone's code other than your own, including code from another's student in your section or another section of the course, or any third party source, e.g. the Internet
  - ➢ DO NOT share or copy anyone else's code for any graded assignment
  - ➢ DO NOT work in pairs or groups
- ✓ All cases of academic misconduct will be reported to the *Committee On Academic Misconduct* (COAM).

Setting up the Programming Environment

Effective commenting and tabbing will affect your grade. The "style" of your program should follow the style of the sample programs in the lecture notes. Also see the example code from Lab #1. Your program should have the file name, your name, creation and last modification dates and a brief description of the program in the comments. ***In addition, read the document on "Commenting" found in the Content tab on Carmen under "Tutorials"***.

1. At the Linux command line type `mkdir lab5`. This will create a new directory named **`lab5`**. Work out of this directory. In order to do that, type `cd lab5`. This changes the current working directory to the directory **`lab5`**.
2. If you have created the directory **`lab5`**, then just type `cd lab5`.
3. Copy the files **`scrabble_solution.exe`** and **`triangle_solution.exe`** in the directory **`/class/cse1222/9643/lab5`** by typing

```
cp /class/cse1222/9643/lab5/scrabble_solution.exe .
cp /class/cse1222/9643/lab5/triangle_solution.exe .
```

Be sure to include ***9643*** (this is your course section indicator) and the period, ".".

Programming Assignment

You will write two programs using ***while loops*** (not *for* loops!) in this lab. You will receive no credit if you use *for* loops.

Write a program **`scrabble.cpp`** that reads in text one character at a time and counts the number of occurrences of the following letters: 'a', 'g', 'm', 'f', 'k', and 'j' in the text, and determines a total score for the text using a point based scheme. Both lower case and upper case letters should be counted together. When a period, '.', or exclamation mark, '!', appears in the input text, the program prints how many of each letter above only, the total score of the text (see point assignments below), and halts. Each letter being counted is assigned a score (similar to the

board game ***Scrabble***) as follows: 'a' (1 point), 'g' (2 points), 'm' (3 points), 'f' (4 points), 'k' (5 points), and 'j' (8 points). All other characters have a score of 0 points. You will compute the total score of the input text by summing the number of occurrences of a letter above multiplied by its score. For example, if the input text is: ***she made a familiar attempt Just right.***, then the counts are 5'a', 1 'g', 3 'm', 1 'f', 0 'k', and 1 'j'. The total score for the phrase is $5 \times 1 + 1 \times 2 + 3 \times 3 + 1 \times 4 + 0 \times 5 + 1 \times 8 = 28$.

Write a program **`triangle.cpp`** which reads in the number of rows and prints *n* rows of digits. For example, if n = 5:

```
12345
1234
123
12
1
```

where the *i*'th row is 1234 . . . *k*. If k is 10 or more digits, then the digit after 9 should start again from 0. For instance, if the number of rows is 15, the output should be:

```
123456789012345
12345678901234
1234567890123
123456789012
12345678901
1234567890
123456789
12345678
1234567
123456
12345
1234
123
12
1
```

Run **`scrabble_solution.exe`** and **`triangle_solution.exe`** to see examples of these programs. Your programs must behave exactly like these programs with the same input and output.

1. Program **`scrabble.cpp`**:

    a. To read in a single character, use `cin` to read a value of type **`char`**.

    b. Use a ***while loop*** to read and count the characters in the input. Read in a character, check that the input character is not the period, '.', nor the exclamation mark, '!', and then enter the while loop.

c. Be sure to read in the next character at the end of the while loop.

d. Lower case and upper case vowels should be counted together. Thus a text with three characters 'e' and four characters 'E' will have a total of 7 occurrences of 'e'.

e. You can compute the total score using the point values of each letter we are counting as you count each letter in the while loop or after you have finished determining the counts, i.e. after the while loop.

f. Run your program on various input including boundary cases such as just a period, ".", or just an exclamation mark, '!', as input.

g. Make sure to place empty lines where appropriate to match the sample solution.

2. Program **`triangle.cpp`**:

   a. Read in the number of rows as an integer.

   b. Use two ***while loops*** one inside the other, i.e. nested, to print the digits in each row.

   c. Run your program on sample input, e.g. 1, 2, 10, and 15.

3. Be sure to add the header comments "File", "Created by", "Creation Date" and "Synopsis" at the top of the file. Each synopsis should contain a brief description of what the program does.

4. Be sure that there is a comment documenting each variable.

5. Be sure that your *if* statements, *while* loops, and blocks are properly indented.

6. Check your output against the output from the solution executables provided.

Submit Your Work

Important: Any program which does not compile and run will receive no credit!
If you are not sure what this means please ask your instructor.

Submit the files **`scrabble.cpp`** and **`triangle.cpp`** using the *Lab5* drop box on Carmen. **DO NOT** submit the file **`a.out`**. **DO NOT** submit work from other assignments. This will not be graded.