CSE 165: Object Oriented Programming
Spring 2022
Lab 3 (100 points)

This programming assignment has six tasks, complete each task as instructed. To submit your assignment, please organize your code in the folder "Lab3" by placing your code in it's corresponding sub-folder. For example, store your code for task 1 in the following directory "Lab3/1/". Then, submit the compressed version of folder Lab3 to CatCourses. Submissions must arrive one minute before the lab section of week 5 (2/14 – 2/18).

1 Reading Strings of Varied Sizes (10 points)

Write a program that keeps reading in strings of varied sizes. If an input string has length greater than one store it in a vector. When an input string has length one (a single character) you will output the string stored in your vector that has the first letter matching the input character. Keep doing this while you read string "quit". In the following example, the user enters four words with more than one letter, followed by the character "a". The resulting output contains all the words previously read that start with the letter "a". Note that the string matching should be case insensitive as seen in the example, the single character is a lowercase "a", whereas the matched strings start with uppercase "A".

```
Input:
    Apple
    Orange
    Banana
    Apricot
    a
    quit
Output:
    Apple
    Apricot
```

2 Run Length Encoder (10 points)

Write a simple run-length encoder: you will read a sequence of pairs containing a character and a number. For each pair (C,N), output the character N times without spaces. When a pair has number -1 print a newline, if the number is -2 then stop. In the following example, the user enters the first pair, "h" and 3, which results in the letter "h" printing 3 times. The second pair, "b" and -1, results in a newline being printed. The third pair, "a" and 2, results in the letter "a" printing twice. Finally, the pair "f" and -2 exits the program.

```
Input:
    h
    3
    b
    -1
    a
    2
    f
    -2
Output:
    hhh

    aa
    (program exit)
```

3 Reading Integer Values (20 points)

Write a program that keeps reading in integer values. If an input value is positive, store it in a vector. If the input value is negative, you will remove all existing values in your vector with the same absolute value. When 0 is read output the number of entries that remained in the vector and the sum of all entries. Then stop. In the following example, a sequence of numbers beginning at 1 and ending at 6 are input, followed by a -5, and finally a 0 to stop reading. The result is the sequence of numbers with the number 5 removed because of the -5 from the input, then the sum of the remaining values, 16.

```
Input:
    1
    2
    3
    4
    5
    6
    -5
    0
Output:
    1 2 3 4 6 16
```

4 Organizing Strings by Length (20 points)

Write a program where you will have a vector V where each entry is a pointer to a vector of strings. This means that each entry in V points to a vector of strings. Your program will then read input strings. For each string, if the number of characters in the string is N, then add it to the string vector in entry V[N-1]. Be sure to allocate the string vector in each entry as needed. The input string will have a maximum of 10 characters so you can initialize V with 10 entries. Do not add repeated entries. Stop when string "quit" is read. String "quit" should not be processed. Then output the contents of each V entry in order from V[0] to V[9], separated by spaces within the same V entry and by a new line when switching to the next entry. Skip empty entries. In the following example a seqeuence of strings, each less than 10 characters, are read, followed by the string "quit" to stop reading. The output prints the strings grouped by character lengths on the same line, starting with length 1, all the way to longest length string, "happiness", which has 9 characters.

```
Input:
    A
    Box
    Cube
    to
    happy
    hello
    computer
    mouse
    happiness
    quit
Output:
    A
    to
    Box
    Cube
    happy hello mouse
    computer
    happiness
```

5 Organizing Strings by Length++ (20 points)

Extend the program from the previous exercise in the following way: after all input strings are read, you will output for each non-empty entry of V the number of letters in that entry and the number of strings in that entry. In the following example the same input is used from the previous exercise, and the resulting outputs are pairs for each index in V. For the first output line, 1 and 1, means there is one string with length 1, for a total of 1 character. Whereas the fifth output line, there is a 15 and 3, because there are 3 strings that have length 5 for a total of 15 characters. The order of the output is similar to the previous exercise, except this time the output is the total characters instead of the strings.

```
Input:
    A
    Box
    Cube
    to
    happy
    hello
    computer
    mouse
    happiness
    quit
Output:
    1 1
    2 1
    3 1
    4 1
    15 3
    8 1
    9 1
```

6 Searching for a String (20 points)

Write a program where you keep reading strings from the input, and then: if the number of characters is four or more, you will store the string in a vector; if the number of characters is less than or equals to three, you will find the strings which begin with the input string in the vector of strings, and then you will output the found strings. Stop when string "quit" is read. String "quit" should not be processed. If no match is found, continue reading. In the following example, a sequence of five strings, each with more 3 characters, are read, followed by a 3 character string, then the 'quit' to stop reading. The output prints all the strings that were read up to that point, that begin with the same characters as the 3 character string, "Hel". Note the matching of strings should be case insensitive, so if in the example we matched "hel" with a lowercase "h", it should produce the same result.

```
Input:
    Hello
    HelloWorld
    HelloHarry
    Care
    Careless
    Hel
    quit
Output:
    Hello HelloWorld HelloHarry
```