CSE-165-Lab 03

Points: 100

Write a separate .cpp file for each of the following tasks.

1. **(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".
   Input: For example, the user inputs words having more than 1 letter are ***"Apple", "Orange", "Banana", "Apricot"*** and, then a character ***"a"***
   Output:
   Apple
   Apricot

2. **(10points)** 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.
   Input: For example, the user inputs h and then ***3***
   Output: hhh

3. **(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.
   Input: For example, the user inputs in sequence, ***1 2 3 4 5 6 -5 0***
   Output: 1 2 3 4 6 16

4. **(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.
   Input:
   Enter the input: A
   Enter the input: Box
   Enter the input: Cube

Enter the input: to
Enter the input: happy
Enter the input: hello
Enter the input: computer
Enter the input: mouse
Enter the input: happiness
Enter the input: quit
Output:
A
to
Box
Cube
Happy hello mouse

computer
happiness

5. **(20 Points)** Extend the program in exercise 4 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.

Input: Same as question 4
Output:
1 1
2 1
3 1
4 1
15 3
8 1
9 1

6. **(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.
Input:
Hello
HelloWorld
HelloHarry
Care

CSE-165-Lab 03

Points: 100

Careless
Hel

Output:
Hello
HelloWorld
HelloHarry

Instruction for submission:

1. Zip all your .cpp files together
2. Submit the zip folder to catcourses