CS4092: Project 1 Outline (10%) - Word/Phrase/Sentence Analyser

---

Overview

You are required to develop a Java application to provide the functionality outlined herein to the end user. Your application should be menu-driven. The user should be permitted to repeatedly choose options from your menu until such time as the user selects the option to exit your application. You are expected to (i) validate user input and (ii) make appropriate use of methods. In addition, your code should not undertake any unnecessary comparisons. **(Note:** You are required to analyse words/phrases/sentences in accordance with the English alphabet**)**

You are required to document your source code such that the input to, the processing performed by as well as the output from each programmer-defined method is detailed.

State clearly any assumptions you make.

Functionality

- Accept a word/phrase/sentence from the end user. Analyze the vowel content of this input, reporting
  - whether or not the input contains no vowels e.g. rhythms,
  - whether or not the input contains all the vowels (in any order) e.g. eunoia,
  - whether or not the input contains all the vowels in alphabetical order e.g. aerious,
  - whether or not the input contains all the vowels in reverse alphabetical order e.g. suoidea and
  - the frequency of each vowel in the input, only for those vowels for which there is at least one occurrence in the input.

- Accept a word/phrase/sentence from the end user. Analyze the consonant content of this input, reporting the frequency of each consonant in the input, only for those consonants for which there is at least one occurrence in the input.

- Accept a word/phrase/sentence from the end user. Analyze the character content of this input, reporting the count of the alphabetic characters, the count of the numeric characters and the count of any other symbols. In addition, your program should report the frequency of each character found in this input.

- Accept a word/phrase/sentence from the end user. Determine which row(s) of keys on a QWERTY keyboard need to be used to type this input for example, the word *Typewriter* can be typed using only the 1st row of keys on a QWERTY keyboard.

- Accept a word/phrase/sentence from the end user. Analyze the vowel and consonant content of this input reporting whether or not this input, in its entirety, comprises alternating vowels and consonants. For example, the name of tennis player ***G*o*r*a*n** ***I*v*a*n*e*s*e*v*i*c*** and the country name ***U*n*i*t*e*d** ***A*r*a*b*** ***E*m*i*r*a*t*e*s***, each in its entirety, comprise alternating vowels and consonants. Similarly, the word ***r*e*h*a*b*i*l*i*t*a*t*i*v*e*** in its entirety comprises alternating vowels and consonants.

- Accept a word/phrase/sentence from the end user and analyse its content, reporting the length of the longest word and the length of the shortest word. In addition, your program should report the word(s) that match the length of the longest word and the word(s) that match the length of the shortest word (duplicates are **not** permitted). For calculation purposes a word may comprise alphabetic characters (a..z and A..Z) and digits (0..9) only, all others symbols should be ignored. For example, if sentence is

  Sit quietly, doing nothing, spring comes, and the grass grows by itself.

  The output should show that the maximum length is 7 and the words matching this length are *quietly* and *nothing*. In addition, the output should show that the minimum length is 2 and the word matching this length is *by*. If the input is

  "Advice is what we ask for when we already know the answer but wish we didn't."

The output should show that the maximum length is 7 and the word matching this length is *already*. In addition, the output should show that the minimum length is 2 and the words matching this length are *is* and *we*.

- Accept two words/phrases/sentences from the end user and determine if they are anagrams of each other. Any word/phrase/sentence that **exactly** reproduces the characters of another word/phrase/sentence in a different order is an anagram (an anagram is a rearrangement of the characters in either a word or phrase (using each character exactly once in the word or phrase created)). For example,
    - the word *Loops* can be formed by rearranging the characters of the word *spool*
    - the word *cinema* can be formed by rearranging the characters of the word *Iceman*
    - the phrase *nine thumps* can be formed by rearranging the characters of the word *punishment*
    - the word *listen* can be formed by rearranging the characters of the word *silent*
    - the phrase *the classroom* can be formed by rearranging the characters of the word *schoolmaster*
    - the phrase *voices rant on* can be formed by rearranging the characters of the word *conversation*
    - the phrase *I'm a dot in place* can be formed by rearranging the characters of the phrase *a decimal point*
    - the phrase *The public art galleries* can be formed by rearranging the characters of the phrase *Large picture halls, I bet*

  However, the word *loops* is not an anagram of *polls*.

- Accept a word/phrase/sentence from the end user and analyse its content, reporting whether the input is a palindrome. In this context, ignoring punctuation and capitalization, you should assume that there are two types of palindromes:
    - a palindrome is a word or phrase that reads the same backward or forward for example,

      rotator *Madam* *NOON* *madam I'm Adam*
      Do geese see God? *Never odd or even* *Don't nod*
    - a palindrome is a phrase at the word-unit level i.e. the words form the same sentence in either direction for example,

      You can cage a swallow, can't you, but you can't swallow a cage, can you?
      King, are you glad you are king?
      I am; therefore, am I?

  However, there are phrases that are palindromes in accordance with both (a) and (b) because each word is itself a palindrome for example,

  I did, did I?

Deadline for Submission

All material must be submitted by 13h00 on Friday of Week 1. Please refer to the document titled CS4092: Project Submission Documentation for specific deliverable instructions. Failure to adhere to these instructions could result in your work attaining zero marks.
