CO322: Data Structure and Algorithms

Lab3: Simple Filter

Aim: Aim of this laboratory class is to develop an algorithm based on the ones you have studied. You will also consider about suitable data structures to use in your code and analyze your code for runtime complexity.

Introduction:
This particular laboratory has 3 main parts. Only the last two parts will be assessed.

Part I: Regular expressions
Regular expressions or *regex* or *regexp* is a sequence of characters that defines a pattern. They are very useful in text processing and are supported by many programming languages. For example the character sequence {a-z}* denote all the words in the English language which uses simple letters.

Your first task is to study about regular expressions and use them in string editing. It is recommended that you try Linux tools such as *sed* and *awk* with regular expressions.

Part II: Coding exercise
You are required to implement a program using Java or python that will find all the words which are written using only the given characters in a given file. Suppose your program is called *regEx*. When you type *$ regEx words.txt aelp* your program should display on screen the words which only contain letters a, e, l, and/or p. The same letter can appear many times and some of the letters may not appear as well. For example, in the above case *apple and ape* should be displayed provided they are in the *word.txt* file.

You shall use the given *word.txt* from moodle to test your application. In that file, each line contains one word and the words are sorted.

Part II: Runtime
Your next task is to find the runtime complexity of your application and space complexity if there is any additional space you need.

Submission:
Please submit your source code **files** to CMS link as a tar ball. If need be you can have more than one file. Include the complexity analysis to the file which has the main function as a comment.
Strictly no late submissions. Deadline for submission 24th February 2017.