COMP 370, Fall l2015
Program #3, Searching text for string that are in the language of a regular expression
Date Assigned: 11/18/15
Date Due: 12/11/15, 3AM
Possible Points: 20

Write a program that reads from a file (the file name should be specified by the first command line argument)

- The alphabet of the language. This appears by itself on the first line of input. Every character in the line (not including the terminating newline character) is a symbol of the alphabet. The alphabet cannot include the letter e, the letter o, the letter U, the letter N, or the symbol *, as these have specific meanings in the regular expressions. (See below.)
- A regular expression that is on the first line of the file, and
- A sequence of strings, one per line of the remainder of the file

. The program then outputs (to a file whose name is specified by the second command line argument) true or false for each string in the input file, true if the string is in the language described by the regular expression, and false if not. The output file should contain one true or false value per line.

Your program should work as follows:

- Write code that converts a regular expression into an NFA. Use the algorithm described in our textbook.
- Convert the NFA to a DFA, using code that your wrote for pa2. Do not use pa2 as a standalone program - you can lift the code out of pa2 and use it as a function/method in this program.
- Simulate the resulting DFA on the strings to be tested, using code that you wrote for pa1, and output the result.

Recall the formal definition of a regular expression from section 1.3 in our text. This is all you have to support. You can require that the input regular expressions be fully parenthesized (as the formal definition implies). All of my test input will be fully parenthesized. Also, you can assume that there will not be any spaces in the regular expressions.

Note that regular expressions have symbols in them that are not standard keyboard characters, so you will use the following substitutions:

1. Substitute e for ϵ (epsilon).
2. Substitute N for $\emptyset$ (the empty set).

3. Substitute U for $\cup$ (the union operator).

4. Substitute o (lower case o) for $\circ$ (the concatenation operator).

5. The * character (which is on the keyboard) is the star operator.

Of course, then, none of these characters can be in the alphabet of the language.

You can program in Java, C, C++, or Python. You can ask about programming in another language, but I must approve it.

You should work with a programming partner, using the pair programming software development technique.

Use good programming style in writing your program (use descriptive identifiers, format your code nicely, structure your code clearly).

Upload your source code only onto blackboard, on the page for the assignment, by the due date/time.