

A regular expression is a special kind of string that describes a pattern of input. These expressions are commonly used in parsers, natural language processing, and in performing several important system administration tasks. Regular expressions can be used in search and replace algorithms, for processing text files, or for implementing various kinds of filters (such as keyword filtering in web proxies).

Background:

An example of a regular expression is: $1(1\cup 3\cup 5\cup 7\cup 9)^*2$

This expression describes the pattern: all strings that consist of a one, followed by zero or more odd digits, followed by a two. The string "17357352" matches the regular expression, but the string "165" does not.

Regular expressions are built from the following components:

Atoms:

An atom is a single character, usually a letter or numerical digit. For example, the regular expression "a" consists of a single atom, the character 'a'. The only strings that match an atom are those strings that have exactly the same characters as the atom. (So, the only string that matches the regular expression "a" is the string "a".)

Choice:

The union symbol can be used to indicate several alternatives for a string. For instance, the regular expression "a \cup b" matches either the string "a" or the string "b". The regular expression "2 \cup 3" matches the string "2" or the string "3", but not the string "7".

Concatenation:

Regular expressions can be combined by concatenation to form new regular expressions. For instance, the atoms "a" and "b" can be concatenated to form the regular expression "ab", which matches only the string "ab". (The only string that matches the regular expression "abc" is the string "abc").

Parenthesization:

An expression may be parenthesized to avoid ambiguity. For instance, the expression "2(3 \cup 4)" matches the string "23" or the string "24", but not the string "4".

The Kleene Star:

A special character, the asterisk, is called the "Kleene Star". The Kleene Star indicates that the preceding element of the expression can be repeated zero or more times. For example, the regular expression "(12)*" describes the set of strings that consist of 0 or more repetitions of the sequence "12". So, "" (the empty string) and "121212" match the expression "(12)*", but "121" does not.

Some more examples of regular expressions:

All binary strings that begin with a 1: $1(0\cup 1)^*$

All words that end in "ing":

$(a\cup b\cup c\cup d\cup e\cup f\cup g\cup h\cup i\cup j\cup k\cup l\cup m\cup n\cup o\cup p\cup q\cup r\cup s\cup t\cup u\cup v\cup w\cup x\cup y\cup z)^*\text{ing}$

Problem:

Create a regular expression that matches "all positive even integers that do not contain a 3 unless they also contain a 7", (you may assume that all numbers are in base ten).

This problem also appears at <http://narnia.homeunix.com/~robert/potm.html>

Solutions: While this problem does not count for Cults, I will give a dollar to the first Longwood student who submits a correct solution. Solutions should be e-mailed to marmorsteinrm@longwood.edu (don't forget to include your name in the e-mail).