**Definition:** A sequence of symbols and characters expressing a string or pattern to be searched for within a longer piece of text.

Another words to say a regular expression is a method used in programming for pattern matching. Regular expressions provide a flexible and concise means to match strings of text.

The regular expressions are built recursively out of smaller regular expressions, using some rules.

Each regular expression r denotes a language L(r), which is also defined recursively from the languages denoted by r ' s subexpressions.

Here are the rules that define the regular expressions over some alphabet £ and the languages that those expressions denote.

* + Basis
  + Induction
  + Precedence

**BASIS:** There are two rules that form the basis:

* + E is a regular expression, and *L(E)* is {E}, that is, the language whose sole member is the empty string.
  + If *a* is a symbol in E, then **a** is a regular expression, and L**(a) =** *{a},* that is, the language with one string, of length one, with *a* in its one position. Here italics is used for symbols, and boldface for their corresponding regular expression.

**INDUCTION:** There are four parts to the induction. Suppose r and *s* are regular expressions denoting languages *L(r)* and *L(s),* respectively.

* + (r)|(s) is a regular expression denoting the language *L(r)* U *L(s).*
  + (r)(s) is a regular expression denoting the language *L(r)L(s).*
  + (r)\* is a regular expression denoting (L(r))\*.
  + (r) is a regular expression denoting *L(r).*The last rule says that we can add additional pairs of parentheses around expressions without changing the language they denote.

**Let E = {a, b}.**

* + 1. The regular expression **a|b** denotes the language *{a, b}.*
  + 2. **(a|b)(a|b)** denotes *{aa, ab, ba, bb},* the language of all strings of length two over the alphabet E.
  + Another regular expression for the same language is **aa|ab|ba|bb.**
  + 3. **a\*** denotes the language consisting of all strings of zero or more a's, that is, { E, *a , a a , a a a , . . . }.*

**Let E = {a, b}.**

* + 4. **(a|b)\*** denotes the set of all strings consisting of zero or more instances of *a* or *b,* that is, all strings of a's and b's: {E ,*a, b,aa, ab, ba, bb,aaa,...}.*
  + Another regular expression for the same language is **(a\*b\*)\*.**
  + **a|a\*b** denotes the language *{a, b, ab, aab, aaab,...*}, that is, the string *a* and all strings consisting of zero or more a's and ending in *b.*

**Operations:**

The various operations on languages are:

* + Union of two languages L and M is written as

L U M = {s | s is in L or s is in M}

* + Concatenation of two languages L and M is written as

LM = {st | s is in L and t is in M}

* + The Kleene Closure of a language L is written as

L\* = Zero or more occurrence of language L.