Regular Languages and Regular Expressions

Regular Language: In theoretical computer science and formal language theory, a *Regular Language* (also called a *rational language*) is a formal language that can be expressed using a Regular Expression.

OR,

A language is said to be a *Regular Language* if some Finite State Machine recognizes it.

Regular Expression:

- Regular Expressions are used for representing certain sets of strings in an algebraic fashion.
- Regular Expressions denote languages.
- Regular Expressions are used to generate pattern of strings.
- Example: $\mathbf{01^* + 10^*}$ denotes the language consisting of all strings that are either a single 0 followed by any number of 1's or a single 1 followed by any number of 0's.
- Practical Example: Valid email address or phone number checking.
- Two regular expressions are equivalent if languages generated by them are same.

Regular Expression Operators:

Before describing the regular expression notation, we need to learn the three operations on regular languages that the operators of regular expressions represent. These operations are:

1. UNION: The *union* of two languages L and M, denoted $L \cup M$, is the set of strings that are in either L or M, or both. For example, if L = {001, 10, 111} and M = {ε, 001}, then $L \cup M$ = {ε, 10, 001, 111}.
2. CONCATANETION: The *concatenation* of languages L and M is the set of strings that can be formed by taking any string in L and concatenating it with any string in M. For example, if L = {001, 10, 111} and M = {ε, 001}, then L.M or just LM, is {001, 10, 111, 001001, 10001, 111001}.
3. STAR: The *star (*or, *closure* or *Kleene closure)* of a language L is denoted L^* and represents the set of those strings that can be formed by taking any number of strings from L, possibly with repetitions (i.e., the same string may be selected more than once) and concatenating all of them. If L = {0, 11}, then L^* consists of those strings of 0's and 1's such that the 1's come in pairs, e.g. 011, 11110, and ε but not 01011 or 101.

Order of precedence of operators: Star, Concatenation and Union; From highest to lowest.

Two theorems to remember:

Theorem 1: The class of Regular Languages is closed under UNION.

Theorem 2: The class of Regular Languages is closed under CONCATANETION.

Rules to remember about Regular Expressions:

Let Σ be an alphabet. The regular expressions over Σ and the sets that they denote are defined below:

- $\emptyset$ is a regular expression that stands for the regular language {} or $\emptyset$, and we write $L(\emptyset)=\emptyset$
- ε is a regular expression that stands for the regular language {ε}, and we write $L(\varepsilon)=\{\varepsilon\}$.
- For any symbol $a \in \Sigma$, **a** is a regular expression that stands for the regular language {a} over Σ, and we write $L(a)=\{a\}$.
- If *E* and *F* are regular expressions, then Union of E and F i.e., **E+F** or **E|F** is also a regular expression denoting the union of *L(E)* and *L(F)*. That is, $L(E+F) = L(E) \cup L(F)$.
- If *E* and *F* are regular expressions, then Concatenation of E and F i.e., **EF** or **E.F** is also a regular expression denoting the concatenation of *L(E)* and *L(F)*. That is, $L(EF) = L(E)L(F)$.
- If E is a regular expression, then Closure of E i.e., E* is also a regular expression, denoting the closure of L(E). That is $L(E^*) = (L(E))^*$

<u>Construction of Regular Languages from Regular Expressions and vice-versa:</u>

1. $L(01) = L(0)L(1) = \{0\}\{1\} = \{01\}$
2. $L(0+1) = L(0) \cup L(1) = \{0\} \cup \{1\} = \{0, 1\}$
3. $L(0^*1) = L(0^*)L(1) = L(\varepsilon+0+00+000+..)\ L(1) = (L(\varepsilon) \cup L(0) \cup L(00) \cup L(000) \cup \ldots)\ L(1) = (\{\varepsilon\} \cup \{0\} \cup \{00\} \cup \{000\} \cup \ldots)\{1\} = \{\varepsilon, 0, 00, 000, \ldots\}\{1\} = \{1, 01, 001, 0001, \ldots\}$
4. $L((0+1)^*) = L(\varepsilon) \cup L(0+1) \cup L((0+1)(0+1)) \cup L((0+1)(0+1)(0+1)) \cup \ldots = \{\varepsilon\} \cup \{0, 1\} \cup \{00, 01, 10, 11\} \cup \{000, 001, \ldots, 111\} \cup \ldots = \{0, 1\}^*$
5. $\{11, 110, 111, 1100, 1101, 1110, 1111, 11000, \ldots\} = L(11(0|1)^*)$
6. $\{0111, 01011, 01111, 010011, 010111, 011011, 011111, 0100011, \ldots\} = L(01(0|1)^*11)$

<u>Linguistic description of regular languages and regular expressions</u>:

1. A language consisting of strings of a's and b's containing aab. Find the regular expression. (For a language consisting of a single string w, we use w itself as the regular expression.)
2. What is the linguistic description of the language denoted by **(0+1)*011.**

** To know more about Regular Expression:
https://en.wikipedia.org/wiki/Regular_expression