Theory of Automata
&
Formal Languages

Handouts 2

Department of Information Technology
Hazara University, Mansehra

Defining Languages

The languages can be defined in different ways , such as Descriptive definition, Recursive definition, using Regular Expressions(RE) and using Finite Automaton(FA) etc.

Descriptive definition of language

The language is defined, describing the conditions imposed on its words.

Example

The language L of strings of odd length, defined over Σ={a}, can be written as

L={a, aaa, aaaaa, ...}

Example

The language L of strings that does not start with a, defined over Σ ={a,b,c}, can be written as

L ={Λ, b, c, ba, bb, bc, ca, cb, cc, ⋯}

Example

The language L of strings of length 2, defined over Σ ={0,1,2}, can be written as

L={00, 01, 02,10, 11,12,20,21,22}

Example

The language L of strings ending in 0, defined over Σ ={0,1}, can be written as

L={0,00,10,000,010,100,110,⋯}

Example

The language **EQUAL**, of strings with number of a's equal to number of b's, defined over Σ={a,b}, can be written as

{Λ ,ab,ba,aabb,abab,baba,abba,..}

Example

The language **EVEN-EVEN**, of strings with even number of a's and even number of b's, defined over Σ ={a,b},

can be written as

{Λ, aa, bb, aaaa,aabb,abab, abba, baab, baba, bbaa, bbbb,⋯}

Example

The language **INTEGER**, of strings defined over Σ={-,0,1,2,3,4,5,6,7,8,9}, can be written as

INTEGER = {⋯,-2,-1,0,1,2,⋯}

Example

The language **EVEN**, of stings defined over Σ={-,0,1,2,3,4,5,6,7,8,9}, can be written as

EVEN = { ⋯,-4,-2,0,2,4,⋯}

Example

The language {$a^n b^n$ }, of strings defined over Σ={a,b}, as

{a^n b^n : n=1,2,3,⋯}, can be written as

{ab, aabb, aaabbb,aaaabbbb,⋯}

Example

The language {$a^n b^n a^n$ }, of strings defined over Σ={a,b}, as

{a^n b^n a^n: n=1,2,3,⋯}, can be written as

{aba, aabbaa, aaabbbaaa,aaaabbbbaaaa,⋯}

Example

The language **factorial**, of strings defined over Σ={0,1,2,3,4,5,6,7,8,9} *i.e.*

{1,2,6,24,120,⋯}

Example

The language **FACTORIAL**, of strings defined over Σ={a}, as

{$a^{n!}$: n=1,2,3,⋯}, can be written as

{a,aa,aaaaaa,⋯}. It is to be noted that the language FACTORIAL can be defined over any single letter alphabet.

Example

The language **DOUBLEFACTORIAL**, of strings defined over Σ={a, b}, as

{$a^{n!}b^{n!}$: n=1,2,3,⋯}, can be written as

{ab, aabb, aaaaaabbbbbb,⋯}

Example

The language **SQUARE**, of strings defined over Σ={a}, as
{a^{n2} : n=1,2,3,…}, can be written as
{a, aaaa, aaaaaaaaa,…}

Example

The language **PRIME**, of strings defined over Σ={a}, as

{a^p : p is prime}, can be written as

{aa,aaa,aaaaa,aaaaaaa,aaaaaaaaaaa⋯}

An Important language

PALINDROME

The language consisting of Λ and the strings s defined over Σ such that Rev(s)=s.

It is to be denoted that the words of PALINDROME are called palindromes.

Example

For Σ={a,b},

PALINDROME={Λ , a, b, aa, bb, aaa, aba, bab, bbb, ...}

Kleene Star Closure

Given Σ, then the Kleene Star Closure of the alphabet Σ, denoted by Σ^*, is the collection of all strings defined over Σ, including Λ.

It is to be noted that Kleene Star Closure can be defined over any set of strings.

Examples

If Σ = {x}

Then Σ^* = {Λ, x, xx, xxx, xxxx, $\cdots$.}

If Σ = {0,1}

Then Σ^* = {Λ, 0, 1, 00, 01, 10, 11, $\cdots$.}

If Σ = {aaB, c}

Then Σ^* = {Λ, aaB, c, aaBaaB, aaBc, caaB, cc, $\cdots$.}

Note

Languages generated by Kleene Star Closure of set of strings, are infinite languages. (By infinite language, it is supposed that the language contains infinite many words, each of finite length).

PLUS Operation (+)

Plus Operation is same as Kleene Star Closure except that it does not generate Λ (null string), automatically.

Example

If Σ = {0,1}

Then Σ^+ = {0, 1, 00, 01, 10, 11, $\cdots$.}

If Σ = {aab, c}

Then Σ^+ = {aab, c, aabaab, aabc, caab, cc, $\cdots$.}

Remark

It is to be noted that Kleene Star can also be operated on any string *i.e.* a* can be considered to be all possible strings defined over {a}, which shows that a* generates Λ, a, aa, aaa, $\cdots$

It may also be noted that a^+ can be considered to be all possible non empty strings defined over {a}, which shows that a^+ generates a, aa, aaa, aaaa, $\cdots$