ALPHABETS, STRINGS AND LANGUAGES -BASIC MATHEMATICAL NOTATIONS

Dr. A. Beulah

AP/CSE

LEARNING OBJECTIVE

- To construct finite automata for any given pattern and find its equivalent regular expressions
  - Before learning FA learn the Basic Mathematical Notation and Techniques

ALPHABET

- An alphabet is a finite, non-empty set of symbols. It is denoted by Σ.

26. $\Sigma = \{a, b, c \dots z\}$

Examples

- **Σ = {a,b}** → alphabet of 2 symbols a and b
- **Σ = {0,1,2}** →an alphabet of 3 symbols 0, 1 and 2

STRING

- A string (or) word is a finite sequence of symbols chosen from some alphabet Σ.

Examples

- **Σ = {a, b}**
  - Strings → abab, aabba, aaabba … a, b, ab bb . . . .
- **Σ = {a}**
  - Strings → a, aa, aaa …
- Notations
  - **a,b,c** → elements of Σ
  - **u, v, w** → string names.

STRING

- $\Sigma = \{0,1\}$
- Strings??

0, 1, 00, 11, 010, 101, ...

00000

11111

OPERATIONS ON STRINGS

1. LENGTH OF A STRING

- The length of a string **x** is the number of symbols contained in the string **x**, denoted by **|x|**.
- Example

  | string | = 6

  | UCS1503 | = 7

  | 101001 | = 6

  $|\varepsilon| = 0$

2. EMPTY (OR) NULL STRING

- The empty string is the string with zero occurrences of symbols or the length of a string is zero.
- It is denoted by ε or λ.
- **$|\varepsilon| = 0 = |\lambda|$**

$\Sigma = \{a, b\}$

a, b, aa, bb....

aba....

ε null string

ε

3. CONCATENATION OF STRING

- Let x and y be strings. Then xy denotes the concatenation of x and y, that is, the string formed by making a copy of x and following it by a copy of y.

$x = a_1\ a_2\ a_3 \ldots\ldots\ldots\ldots\ldots\ldots a_m$

$y = b_1\ b_2\ b_3 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots b_n$

then $xy = a_1\ a_2\ a_3 \ldots\ldots\ldots\ldots\ldots\ldots\ a_m b_1\ b_2\ b_3 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots b_n$

- The length of the string is m+n

Examples

x = 010 y = 1

xy = 0101 yx = 1010.

$x = 010$

$y = \varepsilon$

$xy = 010$

$yx = 010$

$w\varepsilon = \varepsilon w = w$

x = CS y = 6503

xy = CS6503

Empty string is the identity element for concatenation operator ie. wε = εw = w

3. CONCATENATION OF STRING

- x=apple
- y=an
- xy=?

  applean

- yx=?

  anapple

- yy=?

  anan

- xyx=? appleanapple

- x =ε null string
- y={a, b}
- z = ϕ → no string
- xy=?

  $\varepsilon\{a, b\} = \{a, b\}$

- yz=?

  $\{a, b\} \cdot \phi$

  ϕ

4. REVERSE OF A STRING

- The reverse of a string is obtained by writing the symbols in reverse order.

Let w be a string. Then its reverse is w^R

ie. $w = a_1\ a_2\ a_3 .... a_m$

$w^R = a_m\ .... a_2\ a_1$

Example

Let u = 0101011

u^R = 1101010

4. REVERSE OF A STRING

- x=apple
- y=an
- x^R=?

  elppa

- y^R=?

  na

- yy^R=?

  anna

- xy^Rx^R=? applenaelppa

5. POWERS OF AN ALPHABET

- Let Σ be an alphabet.
- Σ^* denotes the set of all strings over the alphabet Σ.
- Σ^m denotes the set of all strings over the alphabet Σ of length m.

Example

If $\Sigma = \{0, 1\}$ then

- $\Sigma^0 = \{\varepsilon\}$ empty string
- $\Sigma^1 = \{0, 1\}$ set of all strings of length one over $\Sigma = \{0, 1\}$
- $\Sigma^2 = \{00, 01, 10, 11\}$ set of all strings of length two over $\Sigma = \{0, 1\}$

$\Sigma^1 \cdot \Sigma^1 = \{0, 1\} \cdot \{0, 1\} = \{00, 01, 10, 11\}$

5. POWERS OF AN ALPHABET

- $\Sigma = \{a,b\}$
- $\Sigma^0 = \{\varepsilon\}$
- $\Sigma^1 = \{a, b\}$
- $\Sigma^2 = \Sigma^1 . \Sigma^1 = \{a, b\} \cdot \{a, b\}$

$= \{aa, ab, ba, bb\}$

$\Sigma^3 = \Sigma^2 . \Sigma^1 = \{aaa, aba, baa, bba, aab, abb, bab, bbb\}$

$\Sigma^4 = \Sigma^3 . \Sigma^1 = \{ \ldots \}$

$\Sigma^m \rightarrow \{ \ \} \ m$

$\Sigma^* = \{ \ \}$

Σ

$\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \ldots$

English

$\Sigma = \{a, \ldots z\}$

$\Sigma^* = \{aa, ab, \ldots zz\}$

a, an,

bat aat

6. KLEENE CLOSURE

- Let Σ be an alphabet. Then "Kleene Closure Σ^*" denotes the set of all strings (including ε, empty string) over the alphabet Σ.

Examples

- If $\Sigma = \{a\}$ then $\Sigma^* = \{\varepsilon, a, aa, aaa, \ldots\}$ i.e.

  $\Sigma^0 = \{\varepsilon\}$

  $\Sigma^1 = \{a\}$

  $\Sigma^2 = \{aa\}$
- If $\Sigma = \{0, 1\}$ then $\Sigma^* = \{\varepsilon, 0, 1, 00, 01, 10, \ldots\}$
- If $\Sigma = \{0\}$ then $\Sigma^* = \{\varepsilon, 0, 00, 000, \ldots)$
- $\therefore \Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \ldots$

$\Sigma^* = \bigcup_{i=0}^{\infty} \Sigma^i$

6. KLEENE CLOSURE

- $\Sigma = \{a,b\}$
- $\Sigma^* = ?$

- $\Sigma = \{0,1,2\}$
- $\Sigma^* = ?$

$\Sigma^0 = \{\varepsilon\}$

$\Sigma^1 = \{0,1,2\}$

$\Sigma^2 = \{00, 01, 02, 10, 20 \cdots\}$

$\Sigma^3 = \{000, 111, 222 \cdots\}$

$\Sigma^* = \{\varepsilon, 0, 1, 2, 00 \cdots$
$000 \cdots\}$

Σ^* set of all strings over the alphabet Σ

7. SUBSTRING

- A string v appears within another string w (w=uv) is called "substring of w." If w=uv, then substrings u & v are said to be prefix and suffix of w respectively.

Examples

- w=abbab

  Substring ={a, ab, abb, ba, bab,...}

- w = 123

  Prefixes = {ε, 1, 12, 123}

  Suffixes = {ε, 3, 23, 123}

- w = abbab

  Prefixes = {ε, a, ab, abb, abba, abbab}

  Suffixes = {ε, b, ab, bab, bbab, abbab}

8. PALINDROME

- A palindrome is a string, which is same whether written forward (or) backward.
- **Example**

  madam, malayalam, noon, nun, 121.
- If the length of a palindrome is even, then it can be obtained by concatenation of a string and its reverse.
- **Example**

  If $u = 01$ $u^R = 10$.

  $w = 0$

  uu^R

  then even palindrome = 0110

9. PROPERTIES OF STRING OPERATIONS

- Concatenation is associative ; that is for all strings u,v and w,

$$(uv)\,w = u\,(vw)$$

- If u and v are strings, then the length of their concatenation is the sum of the individual lengths, i.e.,

$$|uv| = |u| + |v|.$$

Example

x = abc   y = 123   xy = abc123

$|xy| = 6 \quad |x| = 3 \quad |y| = 3$

hence $|xy| = |x| + |y|$

LANGUAGES

- A language is a set of strings which are made up of characters from a specified alphabet, or set of symbols.
- A set of stings all of which are chosen from some Σ*, where Σ is a particular alphabet, is called a language.
- If Σ is an alphabet, and **L⊆ Σ***, then L is a language over Σ*.

Examples

L

English → Σ={a, b, c, ...z}

Binary strings : {0, 1, 01, 10, 0101,.....} → Σ={0,1}

Σ* = {ε, a, b, aa, ab, ...} → Σ = {a, b}

L ← meaningful strings.

Σ

$L \subseteq \Sigma^*$

Σ = {0, 1}

set of all strings which ends with 000 Σ = {0, 1}

L = {000, 1000, 0000, 11000, 10000, 010000 ....}

L = {00, 11, 0000, 1010, 0101, 1100, 0011, ...}

LANGUAGES

Notations

$L = \{\varepsilon\}$

- {λ} (or) {ε}→ Empty string (or) Null string language.

  It is a language over every alphabet and it contains exactly one string ε (or) λ.

$L = \{\}$

- ϕ : Empty language

  It contains no strings. $\{\varepsilon\} \neq \{\}$

- Σ* : Universal language

  It contains all (finite) string over the alphabet Σ.

Note

- ϕ ≠ {λ} ie ϕ has no string where as {ε} (or) {λ} has one string ε (or) λ.

OPERATIONS ON LANGUAGES

A. PRODUCT (OR) CONCATENATION

- $L_1 . L_2 = L_1 L_2 = \{xy : x \in L_1, y \in L_2\}$
- ie., the concatenation of two languages L_1 and L_2 are set of all strings contained by
- concatenating any element of L_1 with any element of L_2.

A. PRODUCT (OR) CONCATENATION

- **L1 = {a,b,aa,bb}**
- **L2={0,1,00,11,01,10}**
- **L1.L2=?**

{a0, b0 aa0, bb0, a1, b1 bb1.....}

B. REVERSAL

- The reverse of a language is the set of all string reversals. ie., $L^R = \{w^R : w \in L\}$
- **L1 = {a,b,ab,ba,bba,baba,}**
- **L2={0,1,01,10,1011,0101}**
- **$L1^R$=?**

{a, b, ba, ab, abb, abab}

w^R

w = 100

w^R = 001

- **$L2^R$**

{0, 1, 10, 01, 1101, 1010}

C. POWER

- For a given language L, $L^0 = \{\lambda\}$
- We define L^n as L concatenated itself n times

ie $L^0 = \{x\}$

$L^1 = L$

$L^K = L \,.\, L^{K-1}$

(or)

$L^K = \{x_1 .... x_K : x_i \in L\}$ where i ranges from 1 to K.

Σ^*

Σ^0

Σ^1

D. KLEENE STAR (OR) STAR CLOSURE

- For a language L,

$$L^* = \bigcup_{i=0}^{\infty} L^i = \underline{L^0 \cup L^1 \cup} L^2 \cup \ldots$$

- **L1 = {a,b,ab,ba,bba,baba,}**
- **$L1^*$=?**

$L^0 = \{\varepsilon\}$

$L^1 = \{$

$L^2 = L^1 \cdot L^1 = \{ \ldots \ldots$

$L^3 = L^2 L^1 = \{ \quad \}$ $\quad L^*$

E. KLEENE PLUS (OR) POSITIVE CLOSURE

- $L^{+} = \bigcup_{i=1}^{\infty} L^{i} = L^{1} \cup L^{2} \cup \ldots$

- **L1 = {a,b,ab,ba,bba,baba,}**
- **$L1^{+}$=?**

$\{\varepsilon$

$L^{*} = \{\varepsilon$

$L^{+} = \{ \ldots \quad \ldots \}$

F. UNION

- The union of L_1 and L_2 denoted by $L_1 \cup L_2$ is
- $L_1 \cup L_2 = \{w: w \in L_1 \text{ or } w \in L_2\}$

- **L1 = {a,b,ab,ba,bba,baba,}**
- **L2={0,1,01,10,1011,0101}**
- **L1UL2=?**

$L_1 \cap L_2 = \{\ \} = \phi$

G. INTERSECTION

- The intersection of L_1 and L_2 denoted by $L_1 \cap L_2$ is
- $L_1 \cap L_2 = \{w : w \in L_1 \text{ and } w \in L_2\}$
- **L1 = {a,b,ab,ba,bba,baba,}**
- **L2={0,1,01,10,1011,0101}**
- **L3={0,1,a,b}**
- **L1 ∩ L2 =?**
- **L1 ∩ L3 =?**

GRAPHS

- A graph, denoted by G = (V,E) consists of a finite set of vertices (or) nodes V and a set E, a pair of vertices called edges.
- A *path in a graph is a sequence of vertices v1, v2, v3, ... vk, k≥1 such that there is an edge (vi, vi+1)* for each *i, 1 ≤ i < k.*
- The length of the path is k-1.
- If *v1=vk, then the path is said to be cycle (because starting and ending at same vertex).*

TREES

- A *tree (strictly speaking ordered, directed tree) is a digraph satisfying following properties:*

(i) There is one vertex called the root, of the tree which is distinguished from all other

vertices and the root has no predecessors.

(ii) There is a directed path from the root to every other vertex.

(iii) Every vertex except the root has exactly one predecessor.

(iv) The successors of each vertex are ordered from left to right.

TEST YOUR KNOWLEDGE

- For any languages L_1, L_2 , L over $\Sigma \neq \emptyset$,

  $(L_1 \cup L_2) \cap L = (L_1 \cap L) \cup (L_2 \cap L)$

  Justify your answer

- For any language L over an alphabet Σ,

  $L^+ = L \cup L^*$

  True or false

SUMMARY

- Introduction about basic mathematical notations - alphabet, strings, languages
- Discussion about different operations on strings
- Languages and operations on languages
- Definition on graph, trees

LEARNING OUTCOME

On successful completion of this topic, the student will be able to:

- Understand the need of basic mathematical notations (K2)

REFERENCE

- Hopcroft J.E., Motwani R. and Ullman J.D, “Introduction to Automata Theory, Languages and Computations”, Second Edition, Pearson Education, 2008