Theoretical Computer Science

UG-CS-604)

Unit-1. Mathematical Preliminaries

1.1 Symbol, Alphabet, String, Formal Language, Operation on languages

1.2 Sets, Relations

- 1.2.1 Sets and Subsets
- 1.2.2 Relations
- 1.2.3 Closure of Relations

1.3 Graphs & Trees

- 1.3.1 Graphs
- 1.3.2 Trees

1.4 Principal of Induction

- 1.4.1 Method of Proof by Induction

Symbol

Lower case letters or digits or special characters are referred as symbols.

Example:

a, b, 0,[, *, l,c,…. are symbols

Alphabet

An alphabet is a finite, non-empty set of symbols

- We use the symbol ∑ (sigma) to denote an alphabet
- Examples:
  - Binary: ∑ = {0,1}
  - All lower case letters: ∑ = {a,b,c,..z}
  - Alphanumeric: ∑ = {a-z, A-Z, 0-9}

Powers of an Alphabet

Let Σ be an alphabet.

- Σ^k = the set of all strings of length k
- $\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \ldots$
- $\Sigma^+ = \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \ldots$

String

A string or word is a finite sequence of symbols chosen from ∑

- ***Empty string is ε (or "epsilon")***

- Length of a string *w*, denoted by "|*w*|", is equal to the *number of (non- ε) characters in the string*
  - *E.g., x = 010100* |*x*| = *6*
  - *x = 01 ε 0 ε 1 ε 00 ε* |*x*| = *?*

- *xy* = concatentation of two strings *x* and *y*

Language

L is a said to be a language over alphabet Σ, only if $L \subseteq \Sigma^*$

→ this is because Σ^* is the set of all strings (of all possible length including 0) over the given alphabet Σ

Examples:

1. Let L be *the* language of all strings consisting of *n* 0's followed by *n* 1's:

   L = {ε, 01, 0011, 000111,…}

2. Let L be *the* language of all strings of with equal number of 0's and 1's:

   L = {ε, 01, 10, 0011, 1100, 0101, 1010, 1001,…}

   Canonical ordering of strings in the language

Definition: Ø denotes the Empty language

- Let L = {ε}; Is L=Ø?

Language Examples:

1. The language of all strings consisting of n 0's followed by n 1's, for some $n \geq 0$: $\{\epsilon, 01, 0011, 000111, \ldots\}$.
2. The set of strings of 0's and 1's with an equal number of each:
$$\{\epsilon, 01, 10, 0011, 0101, 1001, \ldots\}$$
3. The set of binary numbers whose value is a prime:
$$\{10, 11, 101, 111, 1011, \ldots\}$$
4. Σ^* is a language for any alphabet Σ.
5. $\emptyset$, the empty language, is a language over any alphabet.
6. $\{\epsilon\}$, the language consisting of only the empty string, is also a language over any alphabet. Notice that $\emptyset \neq \{\epsilon\}$; the former has no strings and the latter has one string.

Operation on languages:

Concatenation of Strings

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. More precisely, if x is the string composed of i symbols $x = a_1 a_2 \cdots a_i$ and y is the string composed of j symbols $y = b_1 b_2 \cdots b_j$, then xy is the string of length $i + j$: $xy = a_1 a_2 \cdots a_i b_1 b_2 \cdots b_j$.

Example 1.25: Let $x = 01101$ and $y = 110$. Then $xy = 01101110$ and $yx = 11001101$. For any string w, the equations $\epsilon w = w\epsilon = w$ hold. That is, ϵ is the *identity for concatenation*, since when concatenated with any string it yields the other string as a result (analogously to the way 0, the identity for addition, can be added to any number x and yields x as a result). $\square$

* Set Notations :-

"The set is well defined collection of objec(ts) (elements) without repeatation".

The objects are called members or element(s) of sets.

Generally Capital letters like A, B, C, ... e(tc) are used to denote sets & all small letters o(r) numerics [(a, b, c, ...) / (0-9)] are used to denot(e) elements of set.

eg :-

A = {0, 1}

B = {2, 3}

* Subset :-

A set 'A' is said to be subset of 'B' if ever elements of 'A' is also elements 'B' & is denoted as '$A \subseteq B$'.

Two sets A & B are equal if their members are same & denoted as '$A = B$'. This can be prove by denoting $A \subseteq B$ & $B \subseteq A$.

A set with no element is called an empty set or null set & is denoted by 'ϕ'.

* Operations On Sets :-

① $A \cup B = \{x / x \in A$ or $x \in B\}$

② $A \cap B = \{x / x \in A$ and $x \in B\}$

③ $A-B = \{x / x \in A$ and $x \notin B\}$ called difference of $A \not\subset B$.

④ $B-A = \{x / x \in B$ and $x \notin A\}$ called difference of $B \not\subset A$

⑤ 2^A = The set of all subset of set A is called the power set of A.

⑥ Cartesian Product :-

$A \times B = \{(a,b) / a \in A$ and $b \in B\}$ is called cartesian product of A & B, where (a,b) is called ordered pair & it is different from (b,a)

eg :-

① Let, $A = \{a, b\}$

$B = \{b, c\}$

$\Longrightarrow$

① $A \cup B = \{a, b, c\}$

② $A \cap B = \{b\}$

③ $A - B = \{a\}$

④ $B - A = \{c\}$

⑤ $2^A = \{\phi, \{a\}, \{b\}, \{a, b\}\}$

$2^B = \{\phi, \{b\}, \{c\}, \{b, c\}\}$

⑥ $A \times B = \{(a, b), (a, c), (b, b), (b, c)\}$

* Note :-

If A has 'm' element & B has 'n' element then $A \times B = mn$

The power set of a will have 2^m elements

* Graph and trees :-

* Graph :-

A graph (undirected graph) consist of,

① A non-empty set 'V' called the set of vertices

② A set 'E' called the set of edges.

③ A map ϕ which assigns to every edge a unique unordered pair of vertices.

* Directed Graph :-

A directed graph (digraph) consist of,

① A non-empty set V.

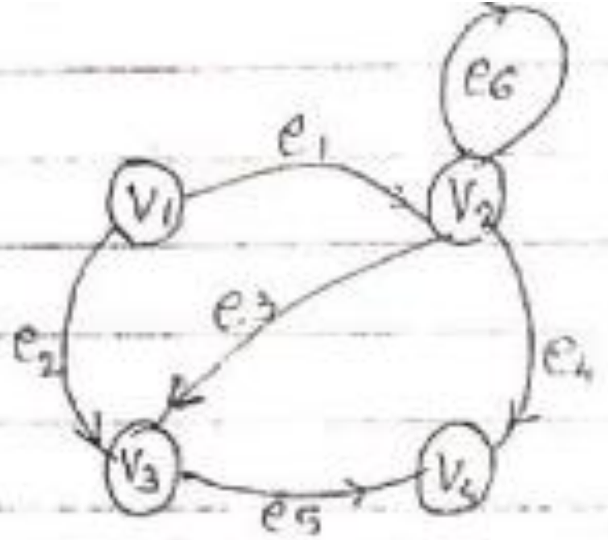
② A set of E

③ A map ϕ which assigns to every edge a unique ordered pair of vertices.

Representation of Graph :-

The representation of digraph same as undirected graph except that edges are represente by directed arcs.

eg :-



A directed graph

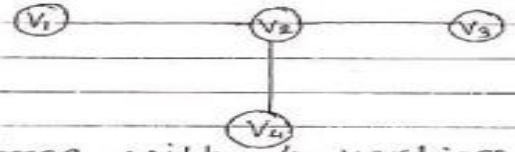
* Trees :-

A graph (directed or undirected) is called a tree, if it is connected & has no circuits (loop).

A directed graph 'G' is a tree iff the corresponding undirected graph is a tree.

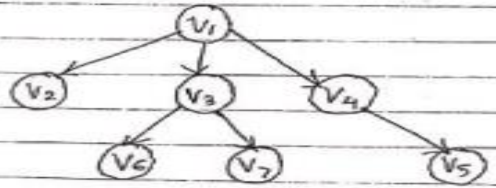
eg :-

①



A tree with 4-vertices (undirected)

②

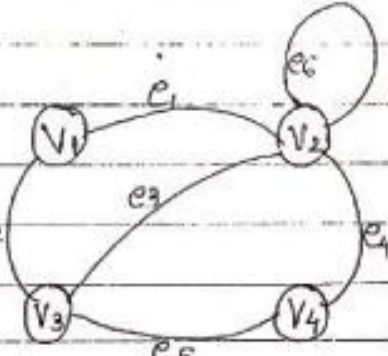


A tree with 7-vertices (directed)

* Representation of Graph :-

A graph is represented by a diagram, where vertices are represented by points or small circles & edges by arcs joining the vertice of the associated pair (given by the map ϕ).

eg :-



An undirected graph

* Properties Of (Graph) Trees:-

A properties of graph is,

① A tree is a connected graph with no circuit or loops.

② In a tree their is one ~~or~~ & only one path bet every pair of vertices.

③ If in a graph their is a unique path betn every pair of vertices, then the graph is tree.

④ A tree with n-vertices has 'n-1' edges.

⑤ If a connected graph with 'n' vertices has 'n-1' edges then it is a tree.

⑥ In a graph with no circuits has 'n' vertices & 'n-1' edges then it is a tree.

* Order Directed Tree :-

An order directed tree is a ~~gr~~ digraph satisfying the following condⁿ.

① Their is one vertex called the root of the tree which is distinguish from the other vertices & the root has no predicessors.

② Their is a directed path from the root to every other vertex.

③ Every vertex except the root has exactly one predicessor.

④ The successor ~~is~~ of each ~~vere~~vertex are order from the left

* **String :-**

The string over & alphabet set 'Σ' is the finite sequence of symbols from 'Σ' (alphabet).

Σ^* - denotes the set of all strings (including ϵ - empty string /null string) over the alphabet set Σ. [Σ^* - kleene closure]

$\Sigma^+ = \Sigma^* - \{\epsilon\}$ [Σ^+ - positive closure]

* Operation on Strings :-

① Concatination :-

"The basic operation for strings is bina[ry] concatination operation. Let "x & y be two strin[gs] in Σ^* therefore a new string z is define as by placing y after x."

$z = xy$

"The process of joining the two strings together to form a composite word is called concatination."

Concatination of two strings is a string formed by writing 1^{st} string followed by [2nd] string with no intervaning space.

eg :-

'wel' & 'come', the concatination of these tw[o] words (strings) is welcome.

eg :-

① $x = aaa$, $y = bbb$

The concatination of x & y is,

$x.y = aaabbb$

② $x = dog$, $y = house$

$x.y = dog.house$

$= doghouse$

③ $x = a^2$, $y = b^3$

$x.y = a^2.b^3$

$= a^2b^3 = aabbb$

② Cardinality of a string :- [length]

let 'w' is the string, $|w|$ is length of the string 'w'.

eg :-

$w = aaabb$

$|w| = |aaabb|$

$= 5$

* $|w|_b$:-

The length of the word 'w' with respective to 'b' within the word.

eg :-

$w = aaabb$

$|w|_b = |aaabb|_{b}$

$= 2$

$|w|_a = |aaabb|_{a}$

$= 3$

③ Empty String / word :-

Given an alphabet, the empty word denoted by 'ϵ' or 'λ' or 'Λ', is define to be the word consisting of zero ~~word~~ letters.

Empty string is not the member of any alphabet (Σ).

i.e. ϵ is not in ($\notin$) Σ

or $\lambda \notin \Sigma$

* Alphabets :-

Different applications may employ different character sets therefore, their should be take care to explicitely mention alphabet under consideration.

* "Σ is an alphabet iff Σ is a finite non-empty set of symbols.

An elements of an alphabet is often called letter. Letters are a-z or 0-9 or ASCII character set.

eg :-

{0,1}, {a,b,c}, etc.

* Languages :-

Language is set of string of symbols from an alphabet. The language is denoted as Σ^*.

eg :-

If $\Sigma = \{a\}$ then

$\Sigma^* = \{\epsilon, a, aa, aaa, \ldots\}$

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

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

$$\Sigma^+ = \bigcup_{k=1}^{\infty} \Sigma^k$$

$$= \Sigma^1 \cup \Sigma^2 \ldots \cup \Sigma^\infty$$

Σ^k - The collection of all words of exactly length 'K' that can be constructed from letters (symbols) of an Σ.

eg :-

If $\Sigma = \{(,)\}$

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

Imp 4M * Equivalence relations :-

A relation 'R' in a set 'S' is called an equivalence relation iff it is reflexive, symmetric, transitive.

* Closure of relations :-

① Let, R be a relation in a set 'S' then transitive closure of R is denoted by R^+ is smallest transitive, relation containing R.

② Let, 'R' be a relation in set S then reflexive transitive closure of R denoted by R^* is the smallest reflexive, transitive relation containing R.

Transitive closure of R (R^+) is defined by

① If (a,b) is in R, then (a,b) is in R^+.

② If (a,b) is in R^+ & (b,c) is in R then (a,c) is in R^+.

③ Nothing is in R^+ unless it so follows from ① & ②

eg :-

Let $S = \{1, 2, 3\}$ then

$R = \{(1,2), (2,2), (2,3)\}$

$R^+ = \{(1,2), (2,2), (2,3), (1,3)\}$

$R^* = \{(1,1), (2,2), (3,3), (1,2), (2,3), (1,3)\}$

(2,2) (2,3)

An *alphabet* is a finite, nonempty set of symbols. Conventionally, we use the symbol Σ for an alphabet. Common alphabets include:

1. $\Sigma = \{0, 1\}$, the *binary* alphabet.

2. $\Sigma = \{a, b, \ldots, z\}$, the set of all lower-case letters.

3. The set of all ASCII characters, or the set of all printable ASCII characters.

A *string* (or sometimes *word*) is a finite sequence of symbols chosen from some alphabet. For example, 01101 is a string from the binary alphabet $\Sigma = \{0, 1\}$. The string 111 is another string chosen from this alphabet.

The Empty String

The *empty string* is the string with zero occurrences of symbols. This string, denoted ϵ, is a string that may be chosen from any alphabet whatsoever.

Length of a String

The standard notation for the length of a string w is $|w|$. For example, $|011| = 3$ and $|\epsilon| = 0$.