UNIT - IV

FORMAL LANGUAGES

Languages:-

A Languages is a dynamic set of visual, Auditory, or tactile symbols of communication and the elements used to manipulate them. Language can also refer to the use of such systems as a general phenomenon.

Symbols and Alphabet:-

A symbol is an abstract entity. It cannot be formerly defined as points in geometry.

Ex: Letters, digits, (or) special symbols like $, @, #, etc.

Alphabet:-

A finite collection of symbols denoted by Σ.

Ex: English Alphabet $\Sigma = \{a, b, \ldots z\}$

Binary Alphabet $\Sigma = \{0, 1\}$

String / word:-

A set of symbols from alphabet.

Ex: 001, 110, 1111 .Strings from binary alphabet.

a01 is not a string from binary alphabet.

Word:-

A word over an alphabet can be any finite sequence (or) string, group of letters.

Note:-

* The set of all words over an alphabet Σ is usually denoted by Σ^*.
* For any alphabet there is only one word of length 0 the empty word is denoted by ϵ, $\in$ (or) $\wedge$.
* An empty string can be denoted by $\in$.

Note:- * string : "abc"

Prefix : a, ab

Suffix : c, bc

Substring : a, b, c, bc, ab.

* Language is a set of words (or) sentences.

Operations on Languages:-

If L_1 and L_2 are two Languages then,

(i) Union → $L_1 + L_2$ (or) $L_1 \cup L_2$

(ii) Concatenation → $L_1 L_2$

(iii) Kleene's closure → Σ^*

EX:1 If $\Sigma = \{x\}$ then find $\Sigma^* = \{\Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \ldots\}$

$\Sigma^* = \{\varepsilon, x, xx, xxx \ldots\}$

EX:2 If $\Sigma = \{a, b\}$ then find $\Sigma \cup \Sigma^*$

$\Sigma \cup \Sigma^* = \{\varepsilon, a, b, aa, bb, ab, ba, aaa, aab, \ldots\}$

Note:- Positive closure $\Sigma^+ = \Sigma^* - \{\varepsilon\}$

$\Sigma^* = \Sigma^+ + \varepsilon$

EX:3 If $L_1 = \{good, bad\}$, $L_2 = \{boy, girl\}$ then find $L_1 \cup L_2$ and $L_1 L_2$?

$L_1 \cup L_2 = \{good, bad, boy, girl\}$

$L_1 L_2 = \{goodboy, goodgirl, badboy, badgirl\}$

Grammars

Grammar is basically defined as set of 4-tuples $G = (V, T, P, S)$, where V is set of Non terminals (variables)

T → is set of Terminals (Primitive symbols)

P → P is the set of Productions (rules) which is relate the non terminals and terminals.

A and S → is start symbol with which strings in grammar are derived.

A Production rules has the form $\alpha \to \beta$ where α and β are strings on V U T and atleast one symbol of of α belongs to V.

EX: $G = (\{S, A, B\}, \{a, b\}, S, \{S \to AB, A \to a, B \to b\}$

$V = \{S, A, B\}$

$T = \{a, b\}$

$S = S$

$P = S \to AB, A \to a, B \to b$

Ans:
$S \to AB$
$\to aB$
$\to aa$.

Note:- Terminals → {. a, b, ... z}, 0-9, (,) , .

Non-terminals → {A, ... z}.

α, β, γ - both terminal, and non-terminals.

Regular Grammar

Naam Chomsky gave a mathematical model of Grammar which is effective for writing computer Languages.

Type-0 Grammar:-

(i) Unrestricted Grammar:-(or) Recursive Grammar:-

A Phrase type grammar with no restriction on production.

Production of the form $\boxed{\alpha \rightarrow \beta}$

α, β contains any number of terminals and non-terminals.

This Grammar can be modeled using "Turing Machine".

EX: $AaC \rightarrow bBDE$

$aBD \rightarrow abcDE$.

Type-2 Context Sensitive Grammar.

*. This grammar generates context sensitive Languages.

*. The Production is in the form of

$\boxed{\alpha A \beta \rightarrow \alpha \gamma \beta}$

where A - non terminal.

α, β, γ - contains any number of terminals, & non-terminal grammar modeled by "Linear bounded Automata"

Ex: $S \rightarrow aAB$

$AB \rightarrow aaB$

$B \rightarrow b$.

Type-2 Context free Grammar

The Production is of the form $A \rightarrow \gamma$

left side A should be non-terminal.

right side γ contain any no of terminal and non terminal.

grammar modeled by Pushdown Automata [PDA].

Ex: $S \rightarrow aA$

$S \rightarrow b$

$S \rightarrow aaA$.

Type-3 Regular Grammar:-

Production is in the form of $A \rightarrow a$, $A \rightarrow aB$ → should contain any no. of terminal and non-terminal grammar.

should be non-terminal

This grammar is modelled using "Finite Automata".

Right Linear Grammar:-

A Grammar is said to be right linear if all Productions are of the form.

$A \rightarrow xB$

$A \rightarrow x$.

where $A, B \in V$ and $x \in T$.

Eg: $S \rightarrow abS/b$, ~~$S \Rightarrow Sbb/b$~~.

Left Linear Grammar:-

A Grammar is said to be left linear grammar. if all Productions are of the form.

$$A \rightarrow Bx$$

$$A \rightarrow x.$$

where $A, B \in V$ and $x \in T$.

Eg:- $S \rightarrow Sbb \mid b$

.

~~Pumping~~

Pumping Lemma for Regular Language:-

A Language is said to be regular. if it is accepted either by a finite Automaton (or) it has a regular grammar generating it. In order to Prove that a Language is not regular The most commonly used technique is "Pumping lemma". The lemma gives a pumping Property that a sufficiently long word has a Subword (non-empty) that can be pumped.

Theorem: (Pumping lemma).

Statement:-

Let L be a regular Language over T. then there exists a constant k depending on L such that for each $w \in L$ with $|w| \geq k$ there exists $x, y, z \in T^*$ such that $w = xyz$ and

1. $|xy| \leq k$
2. $|y| > 1$
3. $xy^i z \in L \quad \forall \; i \geq 0$

where k is no more than the number of states in the minimum state Automaton accepting L.

Proof:-

Let $M = (K, \Sigma, \delta, q_1, F)$ be a deterministic finite state Automaton (DFSA) accepting L. Let $K = \{q_1, \ldots q_k\}$.

Let $w = a_1 \cdots a_m \in L$, where $a_i \in \Sigma$, $1 \leq i \leq m$, $m \geq k$.

Let the transitions on w be as shown below

$$q_1 a_1 \ldots a_m \vdash a_1 q_2 a_2 \cdots a_m \vdash \cdots \vdash a_1 \ldots a_m a_{m+1}$$

where $q_j \in K$, $1 \leq j \leq m+1$. Here $a_1 \cdots a_{i-1} q_i a_i \cdots a_m$ means the FSA is in q_i state after reading $a_1 \ldots a_{i-1}$ and the input head is pointing to a_i. Clearly in the above transitions. $m+1$ states are visited but M has only k states. Hence there exists q_i, q_j such that $q_i = q_j$. Hence for

$q_1 a_1 \ldots a_m \vdash a_1 q_2 a_2 \ldots a_m \vdash (a_1 \ldots a_{i-1} q_i a_i \ldots a_m \ldots \vdash$
$a_1 \ldots a_{j-1} q_j a_j \ldots a_m) \vdash \ldots \vdash a_1 \ldots a_m q_{m+1}$

where the transitions between the brackets start and end at q_i. Processing a string a^t for $t \geq 0$. Hence if

$x = a_1 \cdots a_{j-1}$, $y = a_j \cdots a_j$, $z = a_{j+1} \cdots a_m$, $xy^t z \in L \ \forall \ t \geq 0$

where $|xy| \leq k$ since q_j is the first state identified to repeat in the transition and $|y| \geq 1$ Hence the lemma.

The Potential application of this lemma is that it can be used to show that some languages are non-regular.

Ex:1 show that $L = \{a^n b^n \mid n \leq i\}$ is not regular.

let $L = \{a^n b^n \mid n \leq i\}$

If L is regular, then by Pumping lemma

$\exists$ a constant 'k' which is satisfying the Pumping lemma conditions.

Now choose $w = a^k b^k$

Clearly $|w| > k$, then $w = xyz$, $|xy| \leq k$ and $|y| \geq 1$.

If $|x| = p$, $|y| = q$, $|z| = r$, $p+q+r = 2k$ and $p+q \leq k$.

Hence xy consists of only a's and since $|y| \geq 1$, $xz \notin L$ as number of a's in x is less than k and $|z|_b = k$

Hence the Pumping lemma is not true for $i=0$ as $xy^i z$ must be in L for $i \geq 0$.

Hence L is not regular.

Ex:2 show that $L = \{a^i b^j \mid i, j \geq 1, i \neq j\}$ is not regular.

Definition:- A Grammar $G = (V, T, P, S)$ is said to be Context free if all Production in P have the form $A \to \alpha$, where $A \in V$ and $\alpha \in (V \cup T)^*$.

$V \to$ non-terminals.

$T \to$ Terminals.

$P \to$ Productions

$S \to$ The start symbol.

~~Context free Language~~.

Context free Language:- (CFL)

A Language generated by a CFG is called a Context Free Language. (CFL).

Ex:1 terminal: a

nonterminal: S

Productions: $S \to aS$

$S \to \varepsilon$

Is a simple CFG that defines $L(G) = a^*$

where $V = \{S\}$ $T = \{a\}$

Ex:2 The CFG for defining Palindrome over $\{a, b\}$.

The Productions P are $S \to \varepsilon | a | b$

$S \to aSa$

$S \to bSb$.

and the Grammar is $G = (\{S\}, \{a, b\}, P, S)$

EX:3 Derive 'a⁴' from a Grammar.

Terminal : a

Nonterminal : S

Productions : $S \to aS$

$S \to \varepsilon$

Solution:-

$S \Rightarrow aS$

$\Rightarrow aaS$

$\Rightarrow aaaS$

$\Rightarrow aaaaS$

$\Rightarrow aaaa\varepsilon$

$\Rightarrow aaaa.$

The Language has strings $\{\varepsilon, a, aa, aaa, \ldots\}$

EX:4 Find Language and derive 'abbaaba' from the following grammar.

terminals : a, b

Non-terminals : S, X

Productions : $S \to XaaX$

$X \to aX \mid bX \mid \varepsilon$

Solution:-

$S \Rightarrow XaaX$

$\Rightarrow aXaaX$

$\Rightarrow abXaaX$

$\Rightarrow abbXaaX$

$\Rightarrow abb\varepsilon aaX$

$\Rightarrow abbaabX$

$\Rightarrow abbaabaX$

$\Rightarrow abbaaba\varepsilon$

$\Rightarrow abbaaba.$

Ex: 5 Give the Language defined by grammar G.

$G = \{\{S, C\}, \{a, b\}, P, S\}$, where P is given by

$S \to aCa \quad C \to aCa \mid b$

Solution:-

$S \Rightarrow aCa$

$\Rightarrow aaCaa$

$\Rightarrow aaaCaaa$

$L(G) = a^n b a^n$ for $n \geq 1$.

Ex: 6 Give the Language defined by grammar G.

$G = \{\{S\}, \{0, 1\}, P, S\}$, where P is given by $S \to 0S1 \mid \epsilon$

Solution:-

$S \Rightarrow 0S1$

$\Rightarrow 00S11$

$\Rightarrow 000S111$

$\Rightarrow 000\epsilon 111$

$L(G) = 0^n 1^n \; \forall \; n \geq 0$

Left most and Right most Derivations:-

Ex: 1 Consider the CFG : $(\{S, X\}, \{a, b\}, P, S)$, where

Productions are $S \to baXaS \mid ab$

$X \to Xab \mid aa$.

Find LMD and RMD for string $w = baaaababaaab$.

The following is a LMD:

$S \Rightarrow baXas$

$\Rightarrow baXabas$

$\Rightarrow baXababas$

$\Rightarrow baaaababas$

$\Rightarrow baaaababaab.$

The following is a RMD:

$S \Rightarrow baXas$

$\Rightarrow baXaab$

$\Rightarrow baXabaab$

$\Rightarrow baxababaab.$

$\Rightarrow baaaababaab.$

Note:- Any word that can be generated by a given CFG can have LMD|RMD.

H.W

Ex:2 Consider the CFG

$S \rightarrow aB \mid bA$

$A \rightarrow a \mid aS \mid bAA$

$B \rightarrow b \mid bS \mid aBB$

Find LMD and RMD for string $w = aabbabba$.

Problems:-

Problem:1 Find LMD and RMD for string 00101 in grammar given below $S \to B|A$, $A \to 0A|\epsilon$, $B \to 1B|0B|\epsilon$

Solution:- LMD, RMD are same.

$S \Rightarrow B$

$\Rightarrow 0B$

$\Rightarrow 00B$

$\Rightarrow 001B$

$\Rightarrow 0010B$

$\Rightarrow 00101B$

$\Rightarrow 00101\epsilon$

$\Rightarrow 00101.$