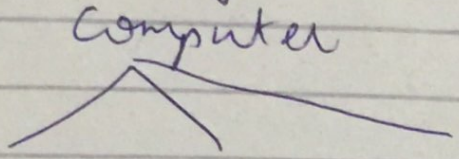
Theory of Automata

↳ Automation machine
↳ self controlled
↳ we give only input and it gives us outcomes

e.g ATM, computer etc

Why we study it ?



=> It allows us to think systematically about what machine do without going hardware details.

=> learning of languages.

=> We are designing theoretical models for machines called Abstract machines.

States ○ ——> ○ (we defined states)
state 1 state 2

=> study of Abstract machines as well as computational problems.

Language

↳ Alphabet
↳ string
↳ word
↳ language

first comes
↳ letters
↳ characters/symbols of which we build language for machine e.g. a, b, c, d 0, 1, 2, 3

Alphabet

↳ letters are combined to form alphabet, denoted by Σ (sigma).

Σ = {a, b} = {0, 1}

2 character — 2 languages

String:-

↳ concentenation of letters.

aa, ab, bb

Language

↳ set of strings with rules.

Example

↳ Make a language for machine in which string starts with 'a' and ends with a from Alphabet {a, b}.

L₁ = {aa, aba, ...}

aa — string

aba — string with rules

(aa), bb
ab, ba
aab, (aba)

Word:-

string that is permissible / means permission / follow in language called word.

String:-

↳ Empty string :- (length = 0)

Λ / λ / ε — gamma denotion

Σ = {a, b} — alphabet — 2 letters (means only 2 types of letters)

a, b, aa, ab, Λ ← empty string, no letter

length of string

↳ no. of letters in a string.

denoted by $|s|$

e.g $s = abab$ $|s| = 4$, length $|s| = 4$

length $|abab| = 4$

Reverse of string: reversing of letters

$s = abab$ $rev(s) = baba$

Power of alphabet

determine the string made from alphabet equal to length 2.

power $\Sigma = \{a, b\}^2 = (aa, ab, ba, bb)$ ← (no more than 2 included)

n^m (formula)

total letters $\Rightarrow (2)^2 \Rightarrow 4$

Power of string:- → determine length of string

e.g $(bab)^2 = babbab$

$ba^2b = baab$

e.g consider the language Σ^*, where $\Sigma = \{a, b\}$, How many words does this language have of length 2 ? of length 3 ? of length n ?

, empty check nahi sktay.

ba^+b => bab, baab, baaaab

$\Sigma\{a,b\}$

↳ $\Sigma^2 = \{a,b\}^2 = \{aa, ab, ba, bb\}$

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

$\Sigma^+ = \{a,b\}^+$
$= \{a, b, aa, ab, ba, bb, aaa \ldots\}$

<u>Lexigraphic order</u>

↳ shortest length first
↳ small to large
↳ strings are grouped by length:

$\Sigma^+\{a, b, aa, ab, ba, bb, aaa, \ldots\}$

(order)

<u>Formal language:-</u>

<u>language</u> -> set of strings with rules.

e.g lang. start with a end with a

$L_1 = \{aa, aba, aaa, abba, \ldots\}$

<u>Syntactic language</u>

↳ concerned with rules/syntax

<u>Informal language</u>

↳ Normal life language
e.g work etc.

Descriptive definition

↳ language defined for machine.

↳ defining methods.

↳ to language that word and string and imposed conditions on string called descriptive.

=> language name = { define definition }

Example

① ↳ Any language that is not start with zero and it is finite

L_1 = { any finite string of letters that doesnot start with letter }

② L_2 = { set of all string of letters that start with a & ends with a }

Question

Define Palindrome language with descriptive definition.

Null string called palindrome

$x = 12321$ > Palindrome

$Rev(x) = 12321$

palindrome = { ~~Null~~ $\wedge$, and all the string x such that reverse ($\wedge$) = $\wedge$ }

e.g {a, b, ~~Null~~ b, aa, aba ...}

Recursive definition

↳ one of language defining methods.

↳ we describe three rules/steps.

1) first we specify some basic objects in set. The no. of basic objects specified must be finite.

P-Even = { 2, 4, 6, 8, 10, ... }

find basic no = ?

basic no = 2

then we plus 2 and plus 2 and so on.

2 + 2 = 4, 4 + 2 = 6, 6 + 2 = 8

x = x + 2 (basic)

2) Second, we give a finite of rules for constructing more object

2 basic no

⇒ like agay walay / kaisay banaygay!

3) No objects except those in step ① & ②

⇒ In 2 rules k iklaw koi nahi aye gay!

e.g Recursive def for +ve even no's.

Rule 1 : 2 is P-even (basic)

Rule 2. If x is in P. Even, then so is x + 2

Rule 3: The only element is set P-even are those can be produced from two rules above