CSO-322 Theory of Computation

Yash Verma
19075083 CSE

4/01/2022

Computation :

Easy and hard problems :

Searching a name in directory

Hard : - factoring 300-digit into its prime factors

Automata Theory :

→ definition and properties of computation models.

Study :

- Automata Theory
- Computability

- Mathematical limitations of hardware
- Design of new programming languges, compiler, string searching , AI etc.
- Problem solving skills.
- Analysing capabilities and limitations.

Computability Theory:

→ Godel, Turing and Church discovered that some of mathematical problem can't be solved by a computer.

- Let Σ be an alphabet, Let Σ^* be the set of all strings over Σ. A language is a subset of Σ^*

→ Let $\Sigma = \{a, b\}$, $\Sigma^* = \{\epsilon, a, b, aa, bb, ab \ldots\}$

Example of languages over Σ are.

$L_1 = \{\epsilon, a, aa, aab\}$

$L_2 = (x \in \Sigma^* : |x| \leq 8\}$

$L_3 = \{x \in \Sigma^* : n_a(x) > n_b(x)$

n_a - no. of a's

n_b = no. of b's

$L_4 = \{x \in \Sigma^* : |x| \geq 2$ x begins and end with b.

- New language can be constructed using set operation, since languages are sets of strings.

→ For any language over an alphabet Σ, their $\cup$, $\cap$ and $-$ are also languages.

→ Compliment of a language over Σ is defined by $L' = \Sigma^* - L$

- Possibility of complement L_1 ·

$L_1' = \Sigma^* - L_1$

② Concatenation operation on strings will allows us to construct new languages.

x and $y \in \Sigma^*$. concatenation .is

For any string x, $x_{\epsilon} = \epsilon X = X$

For string x, y, z $(xy)z = \& x(yz)$

associative

③ Substring of a string

- x is a substring of other string y
- prefix of string
- suffix of a string.

④ Concatenation of Languages:

If $L_1, L_2 \subseteq \Sigma^*$, $L_1L_2 = \{xy \mid x \in L_1, y \in L_2\}$

$L_1 = \{\epsilon\ 01, 011\}$

$L_2 = \{\epsilon, 11, 10, 110\}$

$L_1L_2 = \{\epsilon, 01, 10, 110, 011 \ldots\ldots \}$