MODULE 4
TURING MACHINES

Regular languages form a proper subset of CFG.So push down automata is more powerful than FA.The nature of storage devices differentiated FA and PDA . A PDA and NPDA are not capable of recognizing even simpler languages as{ $a^n b^n c^n$; n>=0}. So turing machine is the next model of computation under machine based approach.

A TM is a very simple machine and has all the power of any digital computer. A TM as a physical computing device can be represented as

Finite Control

X y z

RD/WR head

It has a finite control, an input tape that is divided into cells and a tape head that scans one cell of the tape at a time.The tape has a leftmost cell but it is infinite to the right.Each cell of the tape may hold exactly one of the finite number of tape symbols.Initially, the 'n' leftmost cells , for some finite n>=0 , hold an input which is a string of symbols chosen from a subset of tape symbols called input symbols.The remaining infinity of cells each hold the blank, which is the special tape symbol that is not an input symbol.

x y z B B B

In one move the TM depends upon the symbol scanned by the tape head and the state of the finite control.

1. Changes state
2. Prints a symbol on the tape cell scanned replacing what was written there
3. Moves its head let or right one cell.

A TM is denoted by M =($Q,\Sigma,\Gamma,B,\delta,q_0,F$) where

Q is the finite set of states

Γ is the finite set of allowable tape symbols

Σ a subset of Γ not including B,is the set of input symbols

Δ is the transition function which maps Q * ˥ ->Q * ˥ *{ L ,R}
q_0 is the initial state
F is the set of final states.

An instantaneous description (ID) of the TM M is denoted by $\alpha_1 q \alpha_2$ .Here q is the current state of M and $\alpha_1\ \alpha_2$ is the string in ˥* ie the content of the tape upto the rightmost non-blank symbol.

Language accepted by the turing machine M denoted by L(M) is the set of those words in Σ* that cause M to enter a final state when placed on the tape of M with M in state q_0 and the tapehead of the M at the leftmost cell.

TM starts in the given initial state.It goes through a sequence of steps controlled by the transition function , δ.During this process contents of the tape may be examined and changed.Finally whole process terminate by putting TM into halt state.A TM is said to halt whenever it reaches a configuration for which δ is not defined.ie no transitions aredefined for final state.So TM will halt whenever it enters a final state.

Design a TM M to accept the language, L={$0^n 1^n$ / n>=1 }

Initially tape of M contains $0^n\ 1^n$ followed by blank symbols·Repeatedly M replaces leftmost 0 by 'X' , moves right to the leftmost 1, replacing it by 'Y' moves left to findout rightmost 'X', then moves one cell to the leftmost '0' and repeats the cycle.When searching for a 1, M finds a blank ,then M halts without accepting.If after changing a 1 to Y, M finds no more zeroes then M checks that no more 1's remain accepting if there are none.

Turing Thesis

The power of any computational process is captured within the class of Turing machines.It is only a statement not a theorm.This thesis can be false if a more powerful computational model is proposed that can recognize all the languages which are recognized by the TM model and also recognizes atleast one more language that is not recognized by the TM.

The turing machines are designed to play atleast the following 3 different roles

1. As accepting devices for the languages
2. As a computer of functions –A TM represents a particular function.Initial input is treated as representing an argument of the function and the final string on the tape when the TM enters halt state is treated as representative of the value obtained by the application of the function to the argument represented by the input string.
3. As an enumerator of strings of a language – a device that outputs the strings of a language.

A language that is accepted by a TM is said to be recursively enumerable.

Turing Machine for computing functions

Turing machines can be used to compute functions.For this, the input string is represented in the form BwB , ie the string 'w' is surrounded by blank symbols from both sides and is placed on the leftmost square of the input tape.

Modifications of Turing Machines

1. **TM with stay option**

In standard TM, the read-write head must move either to the right or to the left.Now the third option is to make the read-write head stay in a place after rewriting the cell content.

2. **TM with two way infinite tape**

A TM with a two way infinite tape is denoted by M.

Here the tape is infinite to the left as well as to the right.That is there is an infinity of blank symbols both to the left and right of the current non blank position of the tape.

3. **Multitape TM**

It is an extension to the basic TM consisting of a finite

control and some finite no: of tapes.It has k tapes and k tape heads.Each tape is divided into cells and is infinite to both sides.On a single move, depending on the state of finite control and symbol scanned by each of the tape heads, the machine can

- Changes state
- Print a new symbol on each of the cells scanned by its tape head
- Move each of its tape head independently, one cell to the left or right or keep it stationary.

Initially the head of the first tape is at the left end of the input and all other tape heads are at some arbitrary cell.Each tape is scanned by a tape head that can read, write , move right or left.

4. **Offline Turing Machine**

An offline TM is a multitape TM whose input tape is read only.It has separate input and output tape.Each move is governed by the internal state, what is currently read from the input file and what is seen by the RD/WR head.It uses left and right end markers.An offline TM can simulate a TM M by using one more tape than M.ie it copies its own input onto the extra tape and it then simulates M as if the extra tape were M's input.

5. **Multidimensional TM**

It has a finite control but the tape consists of a k-dimensional array of cells infinite in all directions.Depending on the state and symbol scanned, the device changes state, prints a new symbol and moves its tape head in one of the 2k directions, either positively or negatively.Initially the input is along one axis and the head is at the left end of the input.

6. **Non-Deterministic TM**

A NDTM is a device with finite control and a single , one way onfinite tape.For a given state and tape symbol scanned by the tape head, the machine has a finite no: of choices for the next move.The NDTM accepts its input if any sequence of choices of moves leads to an accepting state.

Universal Turing Machine

A TM is presented as a special purpose computer.Once δ is defined the machine is restricted to carry out one particular type of computation.Computer on the other hand are general purpose machine that can be programmed to do different jobs.Therefore, a TM can't be considered equivalent to general purpose digital computer.This objection can be overcommed by designing a reprogrammable TM called the Universal TM. A universal TM, Mu is an automaton that gives as input the description of any TM, M and a string 'w', can simulate the computation of M on 'w'.

Finite Control

Tape 3

X y z

X y z

tape 1

tape 2

X y z

Tape 1 has description of TM (M) and tape 2 has input tape and tape 3 has internal state of M.

To construct such an Mu, first choose a standard method of describing TM.Assume Q={q1,q2,….} with q1 as the initial state and qn is the final state.⅂={a1,a2……an} where a1=blank.

We can then select an encoding in which q1 is 1,q2 is 11 etc.a1 is 1,a2 is 11 etc.The symbol 0 is used as separator.ie TM has finite encoding as a string in {0,1}*

A universal TM has an input alphabet that is {0,1} and structure of a multitape machine.For any input M and w, tape 1 will will keep an encoded definition of M.Tape 2 contains the tape contents of M and tape 3 stores the internal states of M,Mu looks at tape 2 And 3 to determine the configuration of M.It then consults tape 1 to see what would do in this configuration.Finally tape 2 and tape 3 will be modified to reflect the result of this move.

Halting problem of TM

Given a description of TM, M and an input string ‘w’ , does the turing machine M halts for the input string ‘w’.Halting problem is undecidable.

Decidable and Undecidable Problems

A problem is decidable if there is an algorithm that gives answer to that problem

A problem is undecidable if there is no algorithm that takes as input an instance of the problem and determines whether the answer to that instance is yes or no**.**

Godelization /Godel Numbering

The technique of coding any finite non-numerical string so as to associate an arithmetic integer is called Godel numbering.