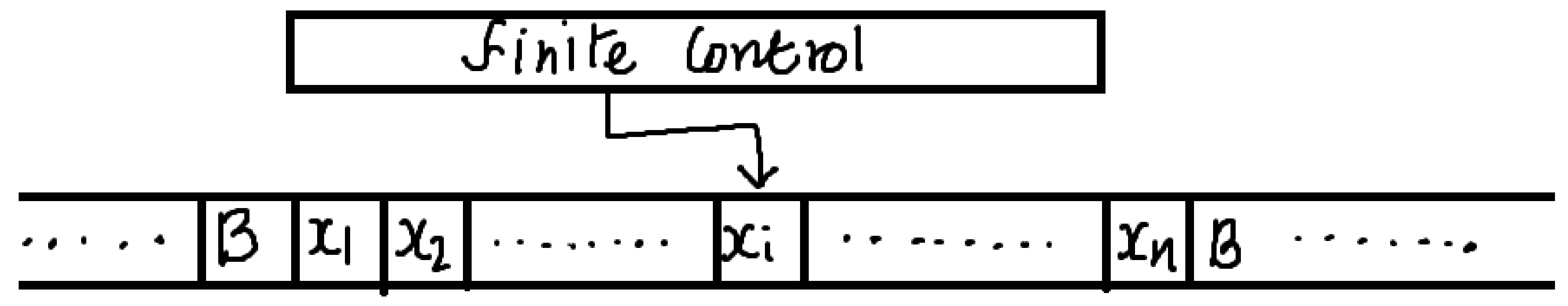
CSC-257
Theory Of Computation
(BSc CSIT, TU)

Ganesh Khatri
kh6ganesh@gmail.com

Chapter 6 - Turing Machine

- Turing machine is an abstract machine developed by an English Mathematician Alan Turing in 1936.
- The model of computation provides a theoretical foundation for modern computers.
- Turing machine accepts the languages generated by type 0(unrestricted) grammars.
- Turing machine will have :
  - A finite set of alphabets
  - A finite set of states
  - A linear tape which is potentially infinite to both end



Turing Machine(TM)

- The tape is marked off into squares, each of which can hold one symbol from the alphabet.
- If there is no symbol in the square then it contains blank.
- The reading and writing is done by a tape head. The tape serves as :
  - Input device (input is simply the string assumed to this)
  - The memory available for use during computations
  - The output device (output is the string of symbols left on the tape at the end of computation).
- A single move of Turing machine is function of the state of TM and the current tape symbol and it consists of three things :
  - Replacing the symbol in the current square by another, possibly different symbol
  - Moving the tape head one square right or left or leaving it where it is
  - Moving from current state to another, possibly different state

Difference between TM and Other Automata (FSA and PDA)

- The most significant difference between the TM and the simpler machine (FSA or PDA) is that - in a Turing Machine, processing a string is no longer restricted to a single left to right pass through input.
- The tape head can move in both directions and erase or modify any symbol it encounters.
- The machine can examine part of the input, modify it, take time to execute some computation in a different area of the tape, return to re-examine the input, repeat any of these actions and perhaps stop the processing before it has locked at all input.

Turing Machine(TM) : Formal Definition

- A Turing Machine M is defined by the seven-tuples,
  $M = (Q, \Sigma, \Gamma, \delta, q0, B, F)$ where,

  Q = the finite set of states of the finite control

  Σ = the finite set of input symbols

  Γ = the complete set of tape symbols; Σ is always a subset of Γ

  q0 = the start state; q0 ε Q

  B = the blank symbol; B ε Γ but B does not belong to Σ.

  F = the set of final or accepting states; F is subset of Q

  δ = the transition function defined by :

  $Q \times \Gamma \rightarrow Q \times \Gamma \times (R, L, S)$; where R, L, S is the direction of movement of head left, or right or stationary. i.e. $\delta(q, X) = \delta(p, Y, D)$; which means TM in state q and current tape symbol X, moves to next state p, replacing tape symbol X with Y and move the head either direction or remains at same cell of input tape.

Instantaneous Description for TM

- The configuration of a TM is described by Instantaneous description(ID) of TM as like PDA
- A string $x_1x_2 \ldots\ldots\ldots\ldots\ldots\ldots x_{i-1}\ q\ x_ix_{i+1} \ldots\ldots\ldots\ldots\ x_n$ represents the ID of TM in which :
  - q is the state of TM
  - the tape head scanning the i^{th} symbol from the left
  - $x_1x_2 \ldots\ldots\ldots\ldots\ x_n$ is the portion of tape between the leftmost and rightmost non-blank.(If the head is to the left of leftmost non blank or to the right of rightmost non-blank then some prefix or suffix of $x_1x_2 \ldots\ldots x_n$ will be blank and i will be 1 or n respectively.)

Moves of TM

- The moves of a TM, $M = (Q, \Sigma, \Gamma, \delta, q0, B, F)$ is described by the notation $\vdash$ for single move and by $\vdash^*$ for zero or more moves as in PDA.

1. For $\delta(q, x_i) = (P, Y, L)$ i.e. next move is leftward then, $x_1x_2 \ldots\ldots\ldots x_{i-1}\, q\, x_ix_{i+1} \ldots\ldots\ldots x_n \vdash x_1x_2 \ldots\ldots x_{i-2}\, P\, x_{i-1}Y\, x_{i+1} \ldots\ldots x_n$ reflects the change of state from q to p and the replacement of symbol x_i with Y and then head is positioned at i-1 (next input scan is x_{i-1})
   - If i = 1, M moves to the left of x_1 i.e. $q\, x_1x_2 \ldots\ldots\ldots x_n \vdash p\, B\, Y\, x_2 \ldots\ldots x_n$
   - If i = n and Y = B, then M moves to state p and symbol B written over x_n joins the infinite sequence of trailing blanks which does not appear in next ID as : $x_1x_2 \ldots\ldots\ldots x_{n-1}\, q\, x_n \vdash x_1x_2 \ldots\ldots\ldots x_{n-2}\, p\, x_{n-1}$

Moves of TM

- The moves of a TM, M = (Q, Σ, Γ, δ, q0, B, F) is described by the notation ⊢ for single move and by ⊢* for zero or more moves as in PDA.

2. If $\delta(q, x_i) = (P, Y, R)$ i.e. next move is rightward then, $x_1x_2 \ldots\ldots x_{i-1}\, q\, x_i \ldots\ldots x_n \vdash x_1x_2 \ldots\ldots x_{i-1}\, Y\, p\, x_{i+1} \ldots\ldots x_n$, which reflects that the symbol x_i is replaced with Y and head has moved to cell i+1 with change in state from p to q
   - If i= n, then i+1 cell holds blank which is not part of previous ID; i.e. $x_1x_2 \ldots\ldots x_{n-1}\, q\, x_n \vdash x_1x_2 \ldots\ldots x_{n-1}\, Y\, p\, B$
   - If i = 1 and Y = B, then the symbol B written over x_i joins the infinite sequence of leading blanks and does not appear in next ID; i.e. $qx_1x_2 \ldots\ldots x_n \vdash px_2\, x_3 \ldots\ldots x_n$

Turing Machine

- **Example : Consider a TM that will accept the language $\{ 0^n1^n \mid n >= 1 \}$**
- Initially, it is given a finite sequence of 0's and 1's on its tape, preceded and followed by an infinity of blanks
- The TM will change 0 to an X and then a 1 to Y until all 0's and 1's are matched
- Starting at left end of the input, it repeatedly changes a 0 to an X and moves to the right over whatever 0's and Y's it sees until comes to a 1
- It changes 1 to a Y, and moves left, over Y's and 0's until it finds X
- At that point, it looks for a 0 immediate to the right. If finds a 0 then changes it to X and repeats the process, changing a matching 1 to a Y.

Turing Machine

- **Example : Consider a TM that will accept the language $\{ 0^n 1^n \mid n >= 1 \}$**
- Algorithm :
  - Change '0' to 'X'
  - Move right to first '1'
  - Change '1' to 'Y'
  - Move left to leftmost '0'
  - Repeat the above steps until no more 0’s and 1’s remain in the tape

Turing Machine

- The Turing Machine will look like :

  M = ({q0, q1, q2, q3, q4}, {0, 1}, {0, 1, X, Y, B}, δ, q0, B, {q4})

- The transition rule for the move of M is described by following transition table :

	0	1	X	Y	B
q0	(q1,X,R)			(q3, Y, R)	
q1	(q1,0,R)	(q2,Y,L)		(q1, Y, R)	
q2	(q2,0,L)		(q0, X, R)	(q2, Y, L)	
q3				(q3, Y, R)	(q4, B, R)
q4					

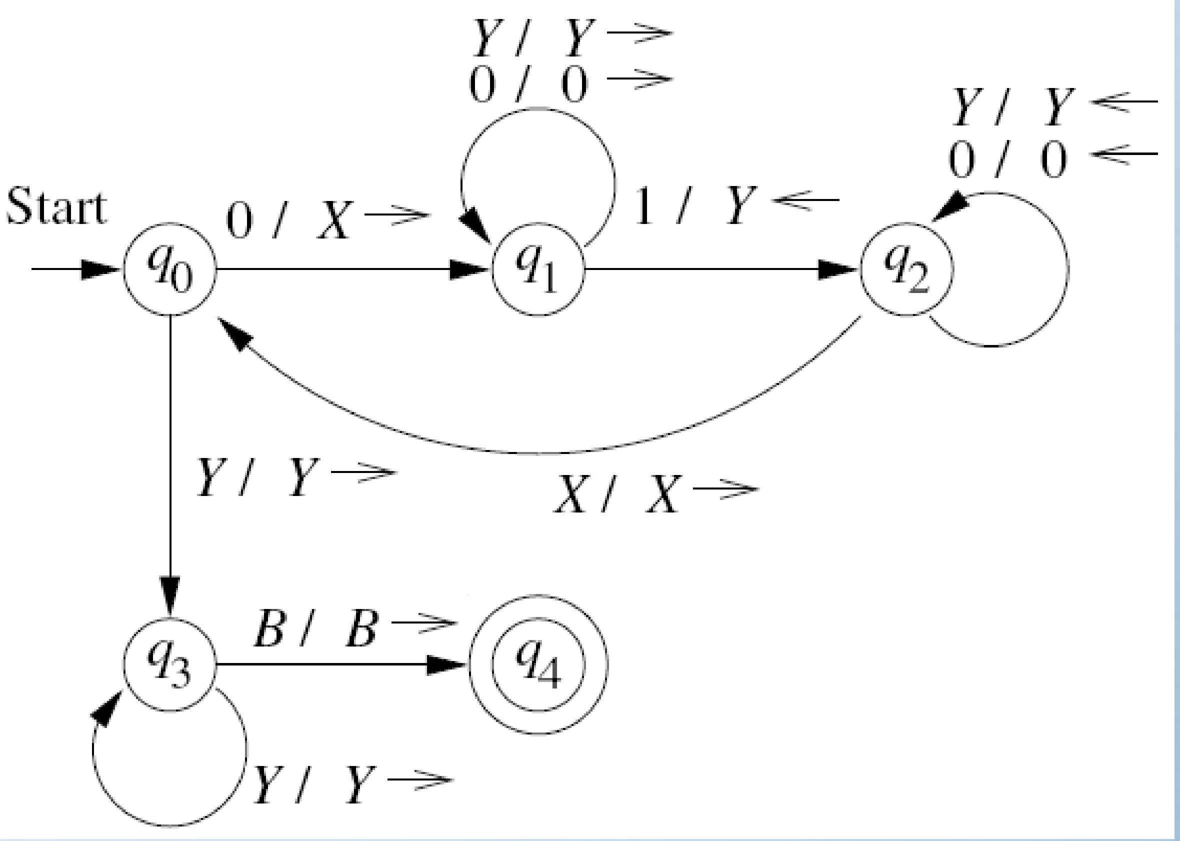
Turing Machine : Transition Diagram

- A transition diagram of TM consists of :
  - A set of nodes representing states of TM.
  - An arc from any state, q to p is labeled by the items of the form X / YD, where X and Y are tape symbols, and D is a direction, either L or R. i.e. whenever $\delta(q, x) = (P, Y, D)$, we find the label x / YD on the arc from q to p.
  - However, in diagram, the direction D is represented by ← for left (L) and → for right (R)

Turing Machine : Transition Diagram

	0	1	X	Y	B
q0	(q1,X,R)			(q3, Y, R)	
q1	(q1,0,R)	(q2,Y,L)		(q1, Y, R)	
q2	(q2,0,L)		(q0, X, R)	(q2, Y, L)	
q3				(q3, Y, R)	(q4, B, R)
q4					

- transition diagram for the TM for $L = \{0^n 1^n \;/\; n \geq 1\}$ is :



Turing Machine : Transition Diagram

- Now, the acceptance of input 0011 by the TM, can be described by following sequence of moves

	0	1	X	Y	B
q0	(q1,X,R)			(q3, Y, R)	
q1	(q1,0,R)	(q2,Y,L)		(q1, Y, R)	
q2	(q2,0,L)		(q0, X, R)	(q2, Y, L)	
q3				(q3, Y, R)	(q4, B, R)
q4					

$q_0\,0011 \vdash X\,q_1\,011$

$\vdash X\,0\,q_1\,11$

$\vdash X\,q_2\,0\,Y\,1$

$\vdash q_2\,X\,0\,Y\,1$

$\vdash X\,q_0\,0\,Y\,1$

$\vdash XX\,q_1\,Y\,1$

$\vdash XXY\,q_1\,1$

$\vdash XX\,q_2\,YY$

$\vdash X\,q_2\,XYY$

$\vdash XX\,q_0\,YY$

$\vdash XXY\,q_3\,Y$

$\vdash XXYY\,q_3\,B$

$\vdash XXYYB\,q_4\,B$ ➔ Halt and accepted

Turing Machine : Transition Diagram

- **Example : Design a TM which recognizes the language L = 01*0**
- Solution : https://www.youtube.com/watch?v=D9eF_B8URnw

Turing Machine : Transition Diagram

- More Examples :
  - https://www.javatpoint.com/examples-of-turing-machine