Turing Machine

→ Turing machine is an automata whose temporary storage is a tape.

→ Tape is divided into cells, each capable of holding one symbol.

→ Associated with tape is R/W head that can travel right or left.

→ Input and output is done at tape

Turing machine is defined as

$$M = (Q, \Sigma, \delta, \Gamma, q_0, F, B)$$

where

$Q \rightarrow$ set of states

$\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$

$F \subseteq Q \rightarrow$ set of final state

$B \rightarrow$ blank symbol

$q_0 \rightarrow$ intial state

we assume $\Sigma \subseteq \Gamma - B$, that is input is written on tape and it doesnot include blank.

Scanned with CamScanner

→ In transition current state and tape symbol are read and result is new state of control unit, a new tape symbol replacing old one and move L/R

Halt state - State where no further move defined. Final state is halting state.

A Turing machine is said to halt whenever it reaches a configiration for which δ is not defined, this is possible as δ is partial function

<u>Example-1</u>

$\delta(q_0, a) \vdash (q_0, b, R)$
$\delta(q_0, b) \vdash (q_0, b, R)$ $\quad F = \{q_1\}$
$\delta(q_0, B) \vdash (q_1, B, L)$

Description - TM replace each symbol by b and move ahead, as soon as see B move to q_1, for q_1 no move is defined and as q_1 is final state so TM halts.

Scanned with CamScanner

Ex-2

$\delta(q_0, a) \vdash (q_1, a, R)$

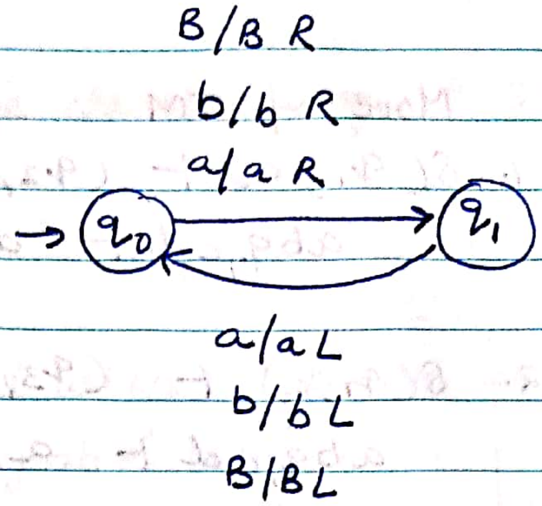
$\delta(q_0, b) \vdash (q_1, b, R)$

$\delta(q_0, B) \vdash (q_1, B, R)$

$\delta(q_1, a) \vdash (q_0, a, L)$

$\delta(q_1, b) \vdash (q_0, b, L)$

$\delta(q_1, B) \vdash (q_1, B, L)$



→ above TM retain symbol as it switch from one state to another first it move right and then left but cannot reach to halting state.

→ This TM is infinite TM.

Main Feature of standard TM

→ unbounded tape in both direction, allowing any number of left and right moves.

→ TM is deterministic that δ defines at most one move for one ~~cogi~~ configration

→ No special i/p file and o/p file some or all content are written on tape.

→ unspecified part of tape is assumed to contain all blanks.

Move of TM

1. $\delta(q_1, c) \vdash (q_2, c, R)$

   $abq_1cd \vdash abcq_2d$

2. $\delta(q_1, c) \vdash (q_3, b, L)$

   $abq_1cd \vdash aq_3bbd$

Acceptance of string - TM enters a final state and halt, then w is considered to be accepted

$w \notin L(M)$ - Machine can halt in non-final state or machine enter an infinite loop and never halt.

Ques. TM that accept language denoted by RE 00^*

Sol $\delta(q_0, 0) \vdash (q_1, 0, R)$

$\delta(q_1, 0) \vdash (q_1, 0, R)$ $F = \{q_2\}$

$\delta(q_1, B) \vdash (q_2, B, R)$

→ Head will move to right when 0 appear and halt

→ It accept when blank symbol is seen

→ If 1 is seen that TM halt as $q_0, 1$ is not defined (halt in non-final state)

Ques $\Sigma = \{a, b\}$ design TM that accepts

$L = a^n b^n \ : n \geq 1$

sol $\delta(q_0, a) \vdash (q_1, x, R)$

$\delta(q_1, a) \vdash (q_1, a, R)$

$\delta(q_1, b) \vdash (q_2, y, L)$

$\delta(q_1, y) \vdash (q_1, y, R)$

$\delta(q_2, y) \vdash (q_2, y, L)$

$\delta(q_2, a) \vdash (q_2, a, L)$

$\delta(q_2, x) \vdash (q_0, x, R)$

Method - TM will match an a with corresponding b

→ if all a are finished it will check whether b are finished, if yes move to final state

Q. Construct TM that will accept following language on $\{a,b\}$

1. $L = ab^*a$

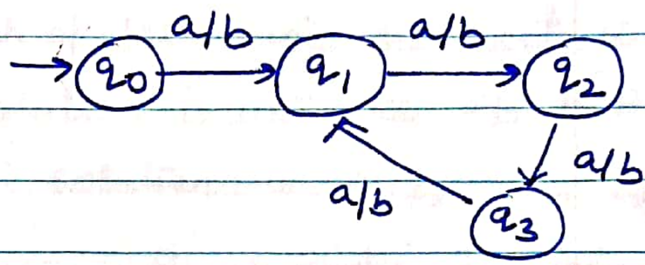
	a	b
$\rightarrow q_0$	$q_1 a R$	
q_1	$q_2 b R$	$q_1 b R$
q_2		

2. $L = \{w : |w| \text{ is even}\}$

	a	b
$\rightarrow q_0$	$q_1 a R$	$q_1 b R$
q_1	$q_0 a R$	$q_1 b R$

3. $L = \{w : |w| \text{ is multiple of } 3\}$

	a	b	B
$\rightarrow q_0$	$q_1 a R$	$q_1 b R$	
q_1	$q_2 a R$	$q_2 b R$	
q_2	$q_3 a R$	$q_3 b R$	
q_3	$q_1 a R$	$q_2 b R$	$q_f B R$
q_f			



4. $L = a^n b^m : n \geq 1, n \neq m\}$

→ Match single a with single b.

→ if a remain and b finish accept string or b

5. $L = \{w : n_a(w) = n_b(w)\}$

→ Match single a with single b

→ when a is read change the state to remember it match with corresponding b.

→ if all a and b are matched move to final state

	a	b	x	B
$\rightarrow q_0$	q_1xR	q_2xR	q_0xR	q_4BR
q_1	q_1aR	q_3xR		
q_3	q_3aR	q_3bR	q_3xR	q_0BR
q_4				
q_2	q_3xR	q_2bR		

6. $L = a^n b^m a^{n+m}$

7. $L = a^n b^{m+n} c^m \quad n,m \geq 0$

8. $L = a^n b^{2n} \quad n \geq 1$

Turing Machine as Transducer

Transducer – It takes input and generate corresponding o/p. Similarly TM not only accept/reject any string but it can generate an o/p for given i/p string.

$\hat{\omega} = f(\omega)$ $\hat{\omega} \to$ o/p and $\omega \to$ i/p
provided that $q_0 \omega \vdash q_f \hat{\omega}$
function f is called turing computable or computable.

Ex.1. Design a turing machine to find 1's complement of a number

	0	1	B
$\to q_0$	$q_0 1 R$	$q_0 0 R$	$q_1 B L$
q_1			

Ex.2 Design a TM to find 2's complement of a no.

	0	1	B
$\to q_0$	$q_0 0 R$	$q_1 1 R$	
q_1	$q_1 1 R$	$q_1 0 R$	$q_2 B R$
q_2			

String is read for LSB
110010

	0	1	B
$\rightarrow q_2$	$q_2 0 R$	$q_2 1 R$	$q_0 L B$
q_0	$q_0 0 R$	$q_1 1 R$	
q_1	$q_1 1 R$	$q_1 0 R$	$q_3 B R$
q_3			

→ when reading the string from left then move at last, without changing symbol then goto state q_0

Q3. Given two positive integer x & y design a TM that compute $x+y$

Sol:- x & y are seperated by 0, and x & y are string of 1

$q_0 w_x 0 w_y \vdash q_f (w(x+y)) 0$

Method - replace first 0 by 1 and last 1 by 0

	1	0	B
$\rightarrow q_0$	$q_0 1 R$	$q_1 1 R$	
q_1	$q_1 1 R$		$q_2 B L$
q_2	$q_3 0 L$		
q_3			

Scanned with CamScanner