Module 6

Formal Definition of TM and its types

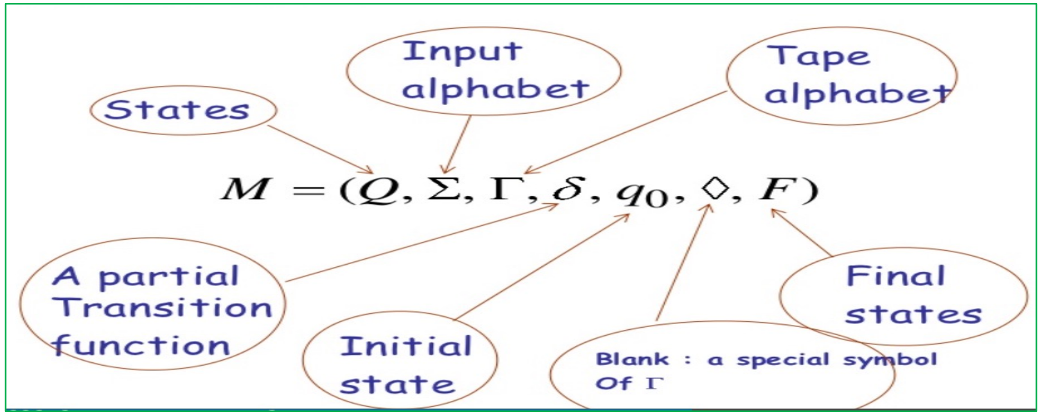
Dr. S. Kiruthika
Assistant Professor – SCOPE
VIT Chenai

Formal Definition of TM

A TM can be formally described as a 7-tuple

(Q, Γ, $\sum$, δ, q_0, B, F)

where −



$\delta : Q \times \sum \rightarrow Q \times X \times \{ \text{Direction}\}$.

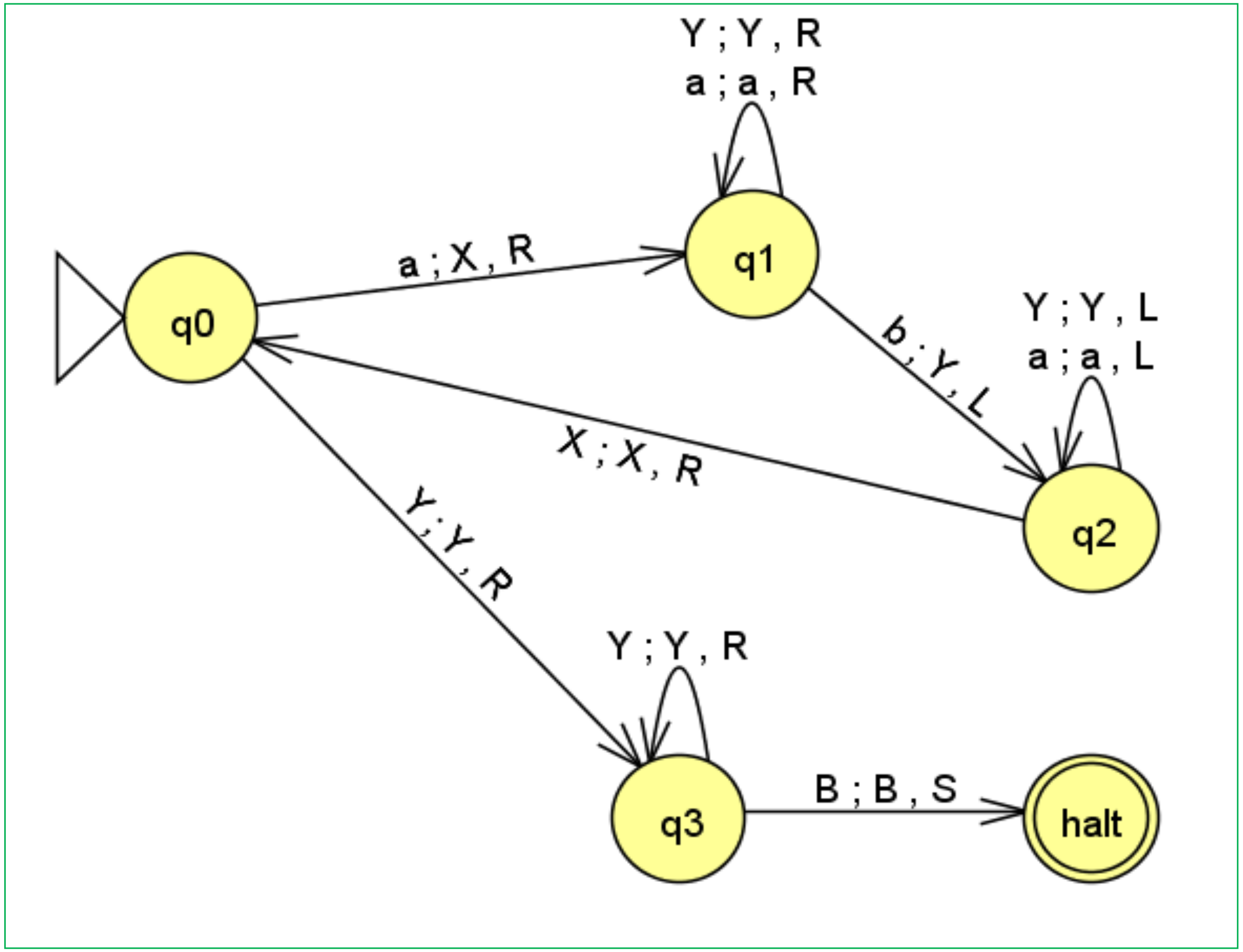
Direction:

Left (L)

Right (R)

Same(S) | Halt (H)

State Diagram



Construct TM for the language L $=\{a^n b^n\}$ where n>=1

Logic:

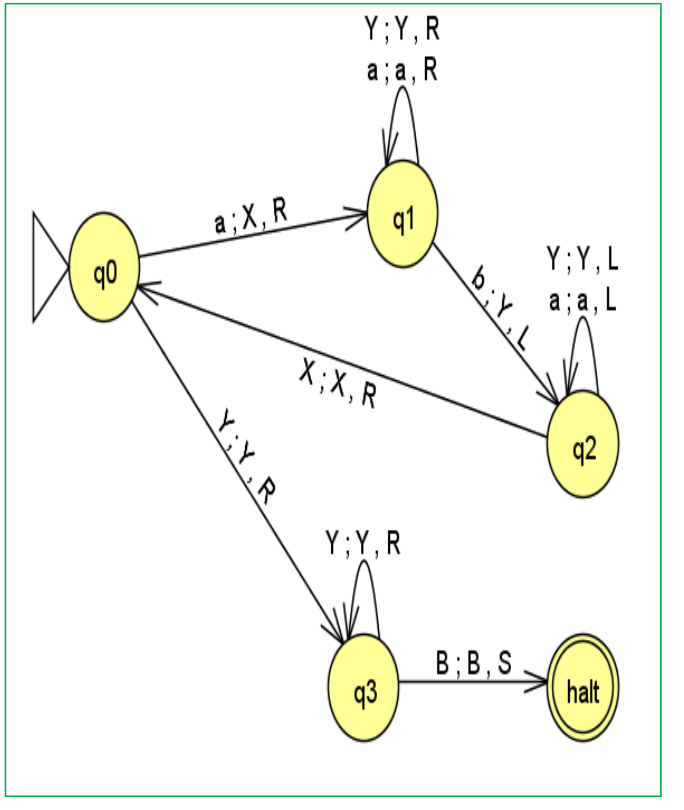
Read out each ‘a' mark it by X and then move ahead along with the input tape and find out b convert it to Y. Now, repeat this process for all a's and b’s.

How this Turing machine work for aabb:

1. B a a b b B	6. B X a Y b B	11. B X X Y Y B
2. B a a b b B	7. B X a Y b B	12. B X X Y Y B
3. B X a b b B	8. B X X Y b B	13.B X X Y Y B
4. B X a b b B	9. B X XY b B	14.B X X Y Y B
5. B X a Y b B	10. B X X Y Y B	15.B X X Y Y B

Transition Table

STATE	INPUT-a	INPUT-b	B	X	Y
q0	(q1,X,R)	-	-	-	(q3,Y,R)
q1	(q1,a,R)	(q2,Y,L)	-	-	(q1,Y,R)
q2	(q2,a,L)	-	-	(q0,X,R)	(q2,Y,L)
q3	-	-	(q4,B,B)	-	(q3,Y,R)
q4 Accept| Halt					



ID

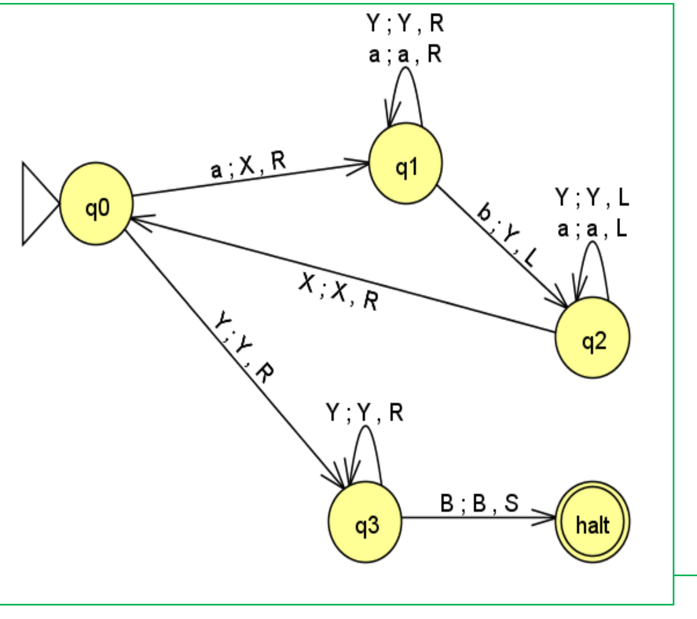
- ID's for TM's are strings of the form:

  $\alpha - q - \beta$,

  where $\alpha, \beta \in \Gamma^*$ and $q \in Q$

Example: Trace the string aabb:

q0 a a b b B



|-Xq1a b b B

|- X a q1b b B

|-X q2a Y b B

|- q2X a Y b B

|-X q0a Y b B

|-X X q1Y b B

|-X X Y q1b B

|-X X q2 Y Y B

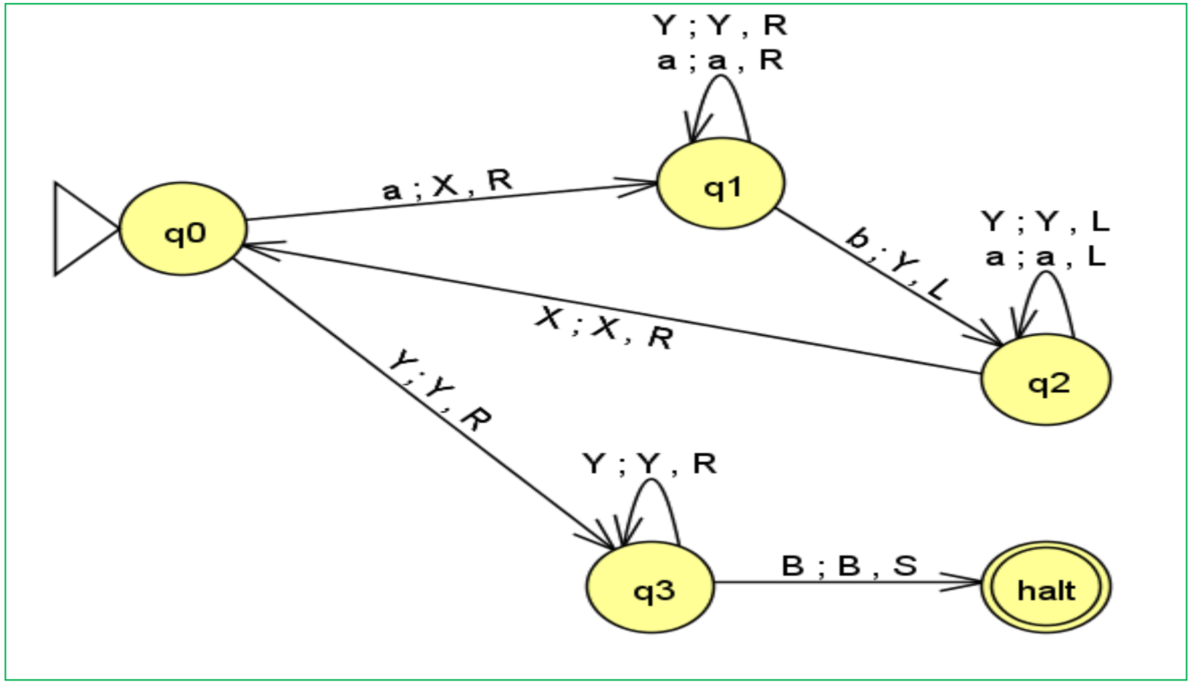
|-X q2X Y Y B

|-X X q0Y Y B

|-XXYq3YB

|- XXYYq3B

|-Halt (Accept | halt)



Universal Turing Machine

- A Universal Computing Machine
- It is a TM, whose input consists of 2 parts
  - A string specifying some other (special purpose) TM- T1
  - A string Z(input to the TM- T1)
  - The TM, Tu then simulates the processing of Z by T1

Construction of Tu

- Step 1: Formulate a notational system
  - For each tape symbol (including Blank)
  - For each state (including halt state)
  - Three directions

For Input Symbols:

Δ- 0

a-00

b-000

For each State:

Halt- 0

q0-00

q1-000

q2-0000

For Directions:

S-0

L-00

R-000

For beginning of string and ending of string – 11

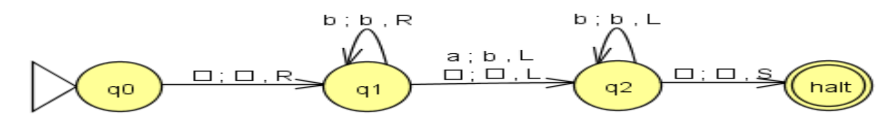
For comma encoding is- 1

Tu- represents the universal TM

T1 – represents the name of special TM

$$Tu = e(T1) . e(Z)$$

Construct a Tu for the TM



Transition Function:

δ(q0, Δ)=(q1, Δ, R)

00 1 0 1 000 1 0 1 000

δ(q1, b)=(q1, b, R)

000 1 000 1 000 1 000 1 000

δ(q1, Δ)=(q2, Δ, L)

000 1 0 1 0000 1 0 1 00

δ(q1, a)=(q2, b, L)

000 1 00 1 0000 1 000 1 00

δ(q2, b)=(q2, b, L)

0000 1 000 1 0000 1 000 1 00

δ(q2, Δ)=(Halt, Δ, S)

0000 1 0 1 0 1 0 1 0

Input	State	Direction
Δ- 0	Halt- 0	S-0
a-00	q0-00	L-00
b-000	P-000	R-000
	R-0000	

e(T) = δ(q0, ∆)=(q1, ∆, R) 11 δ(q1, b)=(q1, b, R) 11δ(q1, ∆)=(q2, ∆, L) 11 δ(q1, a)=(q2, b, L) 11 δ(q2, b)=(q2, b, L) 11δ(q2, ∆)=(Halt, ∆, S)

e(T) = 00 1 0 1 000 1 0 1 000 11 000 1 000 1 000 1 000 1 000 11 000 1 0 1 0000 1 0 1 00 11 000 1 00 1 0000 1 000 1 00 11 0000 1 000 1 0000 1 000 1 00 11 0000 1 0 1 0 1 0 1 0

Assume Z=baa

e(Z) = 000 1 00 1 00

Tu = e(T1) . e(Z)

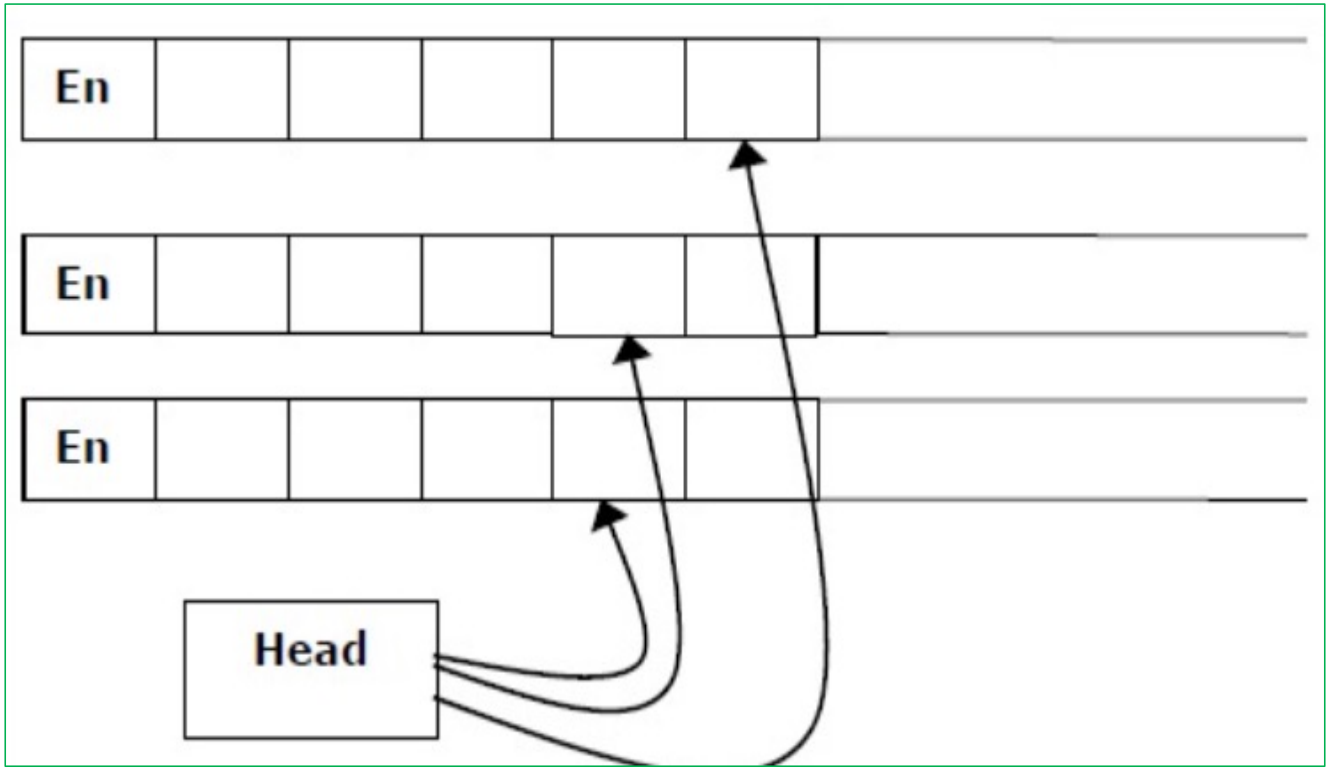
Tu = e(T1) . e(Z)

Tu= 11 00 1 0 1 000 1 0 1 000 11 000 1 000 1 000 1 000 1 000 11 000 1 0 1 0000 1 0 1 00 11 000 1 00 1 0000 1 000 1 00 11 0000 1 000 1 0000 1 000 1 00 11 0000 1 0 1 0 1 0 1 0 11 000 1 00 1 00 11

Turing Machine Types

- Multi-Head Turing Machine
- 2-way infinite tape TM
- Multi-tape Turing machine
- Offline Turing Machine

Multi-Tape Turing Machine



Multi-Tape Turing Machine

- It has multiple tapes and controlled by a single head.
- The Multi-tape Turing machine is different from k-track Turing machine but expressive power is same.
- Multi-tape Turing machine can be simulated by single-tape Turing machine.

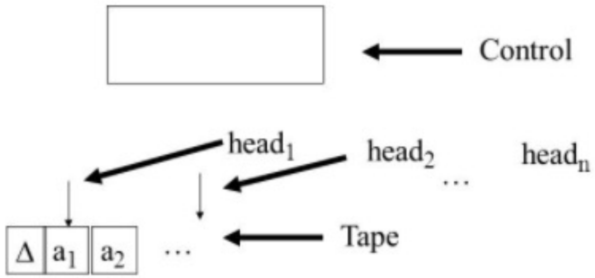
Multi-Head Turing Machine

- A multi-head Turing machine contain two or more heads to read the symbols on the same tape.
- In one step all the heads sense the scanned symbols and move or write independently.
- Multi-head Turing machine can be simulated by single head Turing machine.

MULTIHEAD TURING MACHINE

- Multihead TM has a number of heads instead of one.

- Each head indepently read/ write symbols and move left / right or keep stationery.



We add a finite number of heads for the same tape

2-way infinite tape TM

Both ends are infinite

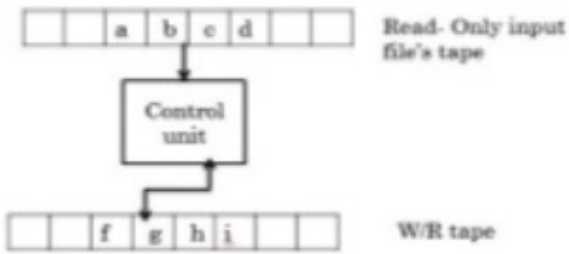
2-way infinite tape TM

- Infinite tape of two-way infinite tape Turing machine is unbounded in both directions left and right.
- Two-way infinite tape Turing machine can be simulated by one-way infinite Turing machine(standard Turing machine).

OFF- LINE TURING MACHINE

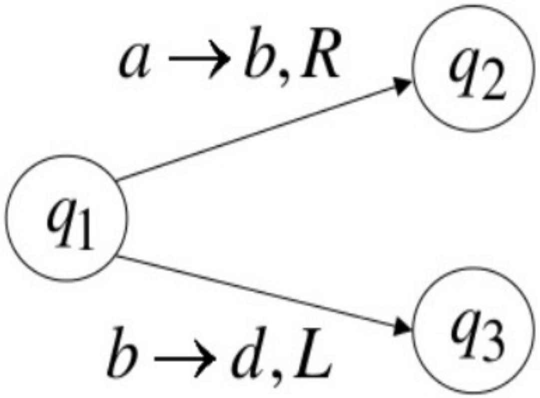
- An *Offline Turing Machine has two tapes*

1. *One tape is read-only* and contains the input
2. The other is read-write and is initially blank.

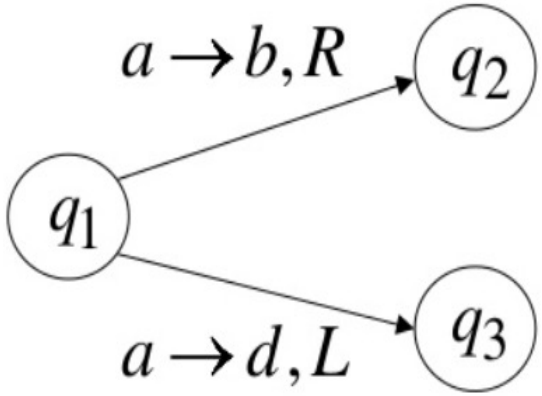


Deterministic TM

Allowed



Not Allowed



Non-Deterministic Turing Machine

- For every state and symbol, there are a group of actions the TM have.
- The computation of a non-deterministic Turing Machine is a tree
- If all branches of the computational tree halt on all inputs, the non-deterministic Turing Machine is called a **Decider**
- If for some input, all branches are rejected, the input is also rejected.