جامعة المستقبل
AL MUSTAQBAL UNIVERSITY

قسم الامن السيبراني
DEPARTMENT OF CYBER SECURITY

SUBJECT: COMPUTATION THEORY
CLASS: 3rd

LECTURER: MSC :MUNTATHER AL-MUSSAWEE

LECTURE: (10)

TURING MACHINE (TM)

Turing Machine (TM)

Turing Machine is an abstract model of computation. It proposed by Alan Turing in 1936. Turing Machine is a FSM with unlimited, unrestricted memory, it can do anything a real computer can do.

It contains infinite Tape, the Tape head can read and write symbols (only one at a time), and can move left or right, initially the Tape contains the input string.

The language called Turing-recognizable if some Turing machine recognizes it, also known as a recursively enumerable language. And the language called Turing-decidable if some Turing machine decides it, also known as decidable or a recursive language.

A Turing machine (**TM**) is a collection of six things:

1- An alphabet Σ of input letters.

2- A TAPE divided into a sequence of numbered cells each containing one character or a blank.

3- A TAPE HEAD that can in one step read the contains of a cell on the TAPE, replace it with some other character, and reposition itself to the next cell to the right or to the left of the one it has just read.

4- An alphabet Γ of character that can be printed on the TAPE by the TAPE HEAD.

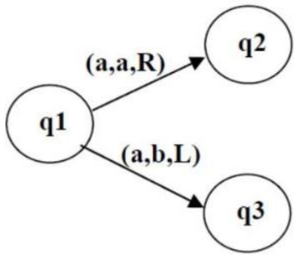
5- A finite set of states including exactly one START state from which we begin execution, and some (may be none) HALT states that cause execution to terminate when we enter them. The other states have no functions, only names: q1, q2, q3 … or 1, 2, 3, …

6- A program, which is a set of rules that tell us on the basis of the letter the TAPE HEAD has just read, how to change states, what to print and where to move the TAPE HEAD. We depict the program as a collection of directed edge connecting the states. Each edge is labeled with a triplet of information:

(letter, letter, direction).

The first letter (either Δ or from Σ or Γ) is the character that the TAPE HEAD reads from the cell to which it is pointing, the second letter (also Δ or from Γ) is what the TAPE HEAD prints in the cell before it leaves, the third component, the direction, tells the TAPE HEAD whether to move one cell to the right(R) or to the left (L).

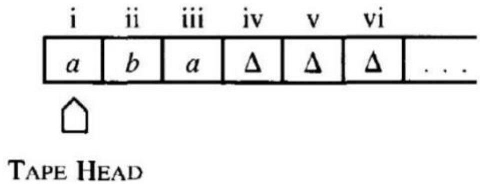
✍ **Note**: TM is deterministic. This means that there is no state q that has two or more edges leaving it labeled with the same first letter. For example, the following TM is not allowed:



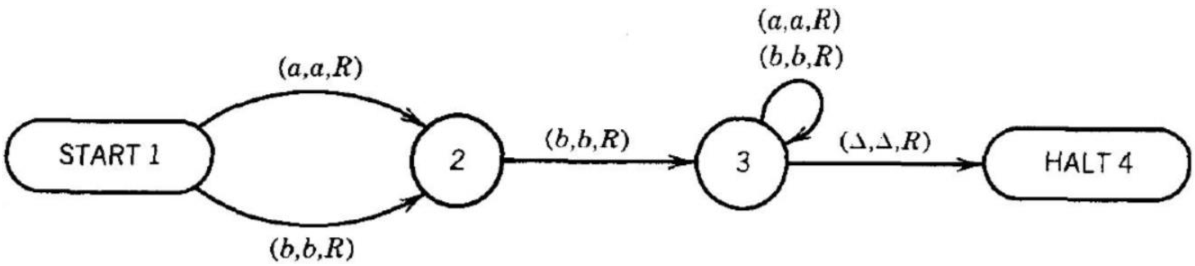
Example: Find TM that can accepts the language defined by the regular expression:

(a+b)b(a+b)*

The following is the TAPE from a Turing machine about to run on the input *aba*:



The program for this TM is given as a directed graph with labeled edges as shown below:

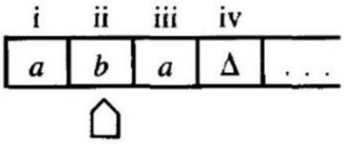


Notice that the loop at state 3 has two labels. The edges from state 1 to state 2 could have been drawn as one edge with two labels. We start, as always, with the TAPE HEAD reading cell i and the program in the start state, which is here labeled state 1. We depict this as:

1

$\underline{a}ba$

The number on top is the number of the state we are in. Below that is the current meaningful contents of the string on the TAPE up to the beginning of the infinite run of blanks. It is possible that there may be Δ inside this string. We underline the character in the cell that is about to be read. At this point in our example, the TAPE HEAD reads the letter *a* and we follow the edge (*a,a,R*) to state 2. The instructions of this edge to the TAPE HEAD are "read an *a*, print an *a*, move right" The TAPE now looks like this:



We can record the execution process by writing:

$$\begin{matrix} 1 & & 2 \\ \underline{a}ba & \rightarrow & a\underline{b}a \end{matrix}$$

At this point we are in state 2. Since we are reading the *b* in cell ii, we must take the ride to state 3 on the edge labeled (*b,b,R*). The TAPE HEAD replaces the *b* with *b* and moves right one cell. The idea of replacing a letter with itself may seem silly, but it unifies the structure of Turing machines.

We could instead have constructed a machine that uses two different types of instructions: either print or move, not both at once. Our system allows us to formulate two possible meanings in a single type of instruction.

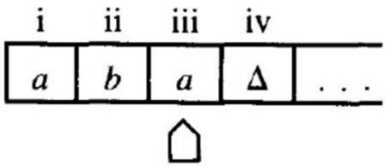
(a, a, R) means move, but do not change the TAPE cell

(a, b, R) means move and change the TAPE cell

This system does not give us a one-step way of changing the contents of the TAPE cell without moving the TAPE HEAD, but we shall see that this too can be done by our TM's. Back to our machine. We are now up to

$$\begin{matrix} 1 & & 2 & & 3 \\ \underline{a}ba & \rightarrow & a\underline{b}a & \rightarrow & ab\underline{a} \end{matrix}$$

The TAPE now looks like this.



We are in state 3 reading an *a*, so we loop. That means we stay in state 3 but we move the TAPE HEAD to cell iv.

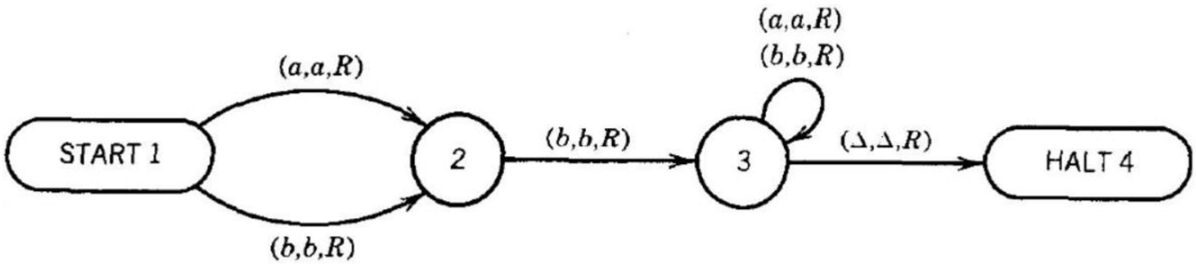
$$\begin{matrix} 3 & & 3 \\ ab\underline{a} & \rightarrow & aba\underline{\Delta} \end{matrix}$$

This is one of those times when we must indicate a Δ as part of the meaningful contents of the TAPE. We are now in state 3 reading a Δ, so we move to state 4.

3
aba<u>Δ</u> → HALT

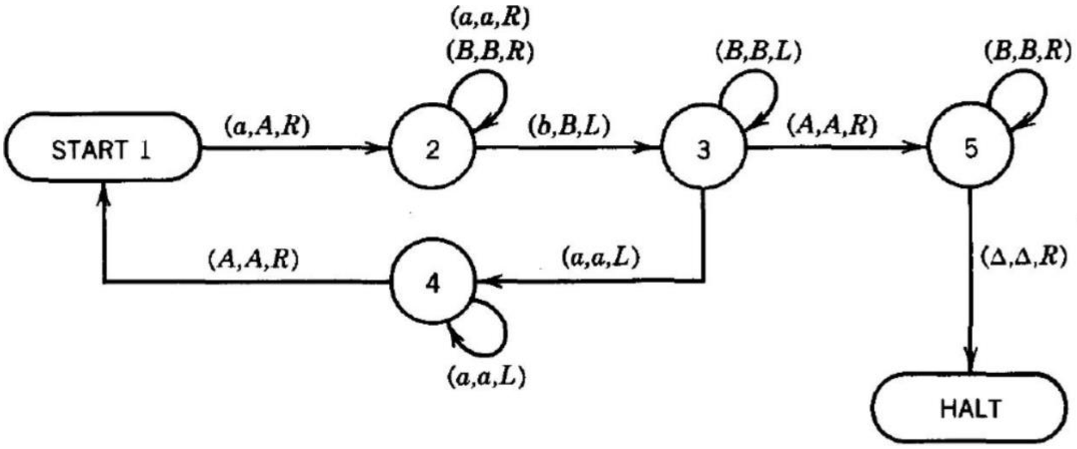
The input string aba has been accepted by this TM. Now we trace the acceptance of the string: **aba**

1 → 2 → 3 → 3
<u>a</u>ba ab<u>a</u> aba<u>Δ</u> aba<u>Δ</u> → HALT



State	Tape head position
Start 1	**<u>a</u>**ba
2	a**<u>b</u>**a
3	ab**<u>a</u>**
3	aba**<u>Δ</u>**
HALT4	aba**<u>Δ</u>**

Example: Find TM that can accepts the language $\{a^n b^n\}$



Now we trace the acceptance of the string: ***aaabbb***

State	Tape head position
Start 1	**<u>a</u>**aabbb
2	A**<u>a</u>**abbb
2	Aa**<u>a</u>**bbb
2	Aaa**<u>b</u>**bb
3	Aa**<u>a</u>**Bbb
4	A**<u>a</u>**aBbb
4	**<u>A</u>**aaBbb
Start 1	A**<u>a</u>**aBbb
2	AA**<u>a</u>**Bbb
2	AAa**<u>B</u>**bb
2	AAaB**<u>b</u>**b
3	AAa**<u>B</u>**Bb
3	AA**<u>a</u>**BBb
4	A**<u>A</u>**aBBb
Start 1	AA**<u>a</u>**BBb
2	AAA**<u>B</u>**Bb
2	AAAB**<u>B</u>**b
2	AAABB**<u>b</u>**
3	AAAB**<u>B</u>**B
3	AAA**<u>B</u>**BB
3	AA**<u>A</u>**BBB
5	AAA**<u>B</u>**BB
5	AAAB**<u>B</u>**B
5	AAABB**<u>B</u>**
HALT	AAABBB<u>Δ</u>

Accept in this TM

Homework: Find a TM that can accepts the language L = $\{0^n1^n2^n\}$ where $n \geq 1$ by trace the acceptance of the string: "001122"