

# Turing Machine

Prof. (Dr.) K.R. Chowdhary  
*Email: kr.chowdhary@iitj.ac.in*

Formerly at department of Computer Science and Engineering  
MBM Engineering College, Jodhpur

Monday 10<sup>th</sup> April, 2017

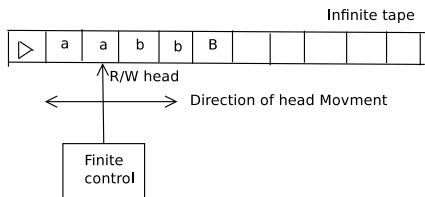
- Alan M. Turing
- Church-Turing Thesis
- Definitions
- Computation
- TM Configurations

- Alan Turing was one of the founding fathers of CS.
- His computer model - the Turing Machine (TM) - was inspiration /premonition of the electronic computer which came two decades after the TM model
- He was instrumental in cracking the Nazi Enigma cryptosystem in WW-II
- Invented the *Turing Test* for in AI
- Legacy: *The Turing Award, eminent award in CS research*

## Definition

(Church-Turing Thesis:) TM is ultimate model for computation. Any thing, which is solvable, i.e., has an algorithm, what ever the computation model is used to compute that algorithm, it is ultimately the TM model.

# Turing Machine Model for computation



$$M = (Q, \Sigma, \Gamma, \delta, s, H),$$

where,  $Q$  is set of states

$H$  is set of Halting states,  $H \subseteq Q$

$\Sigma$  is set of input symbols

$\Gamma$  is tape alphabet,  $\Gamma = \Sigma \cup \{B, \triangleright\}$

$\delta$  is transition function (a partial function),

$\delta : (Q - H) \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$

## Definition

Language acceptability by TM:

$$L = \{w \mid w \in \Sigma^*, q_0 w \vdash^* \alpha p \beta, p \in H; \alpha, \beta \in \Gamma^*\}$$

- TM has a infinite amount of read / write memory. Input is assumed to reside on the tape.
- 1936(Alan M. Turing): Any logical/arithmetic computation, for which complete instructions for carrying out this are supplied, it is always possible to design a TM that can perform this computation.
- TM v/s Human: TM model is based on human problem solving process using pencil

and paper. As we do this, our mental state changes, for every smallest step.

Correspondingly, TM has tape (=paper), R/W head (=pencil), and state (=state of our mind).

- $\{a^n b^n | n \geq 0\}$  v/s  $\{a^n b^n c^n | n \geq 0\}$
- Powers of TM: Power is problem solving capability, and not about how fast or slow it can do.

These are Automata, as simple as possible - to define formally, describe and reason about them, as general as possible (any computation can be represented by them).

## Definition

(Acceptability by Turing machine:) A string  $w$  is accepted by TM  $M$  if after being put on the tape with the TM head set to the left-most position, and letting  $M$  run,  $M$  eventually enters the halting state state. In this case  $w$  is an element of  $L(M)$ , the language accepted by  $M$ :-

$$L(M) = \{w \mid w \in \Sigma^* \wedge q_0 w \vdash^* \alpha y \beta\},$$

where,  $y$  is halting configuration, and  $\alpha, \beta \in \Gamma^*$

# Turing Machine solves a Problem: Erase entire tape

Consider a TM  $M = (Q, \Sigma, \Gamma, \delta, s, H)$ ,  $Q = \{q_0, q_1\}$ ,  
 $\Sigma = \{a\}$ ,  $\Gamma = \{a, B, \triangleright\}$ ,  $s = q_0$ ,  
 $B$  is blank character,  $\triangleright$  is left  
end marker.  $H = \{q_1\}$

$\delta(q_0, a) = (q_0, B, R)$

$\delta(q_0, B) = (q_1, B, L)$

Let  $w = aaaa$

$q_0 aaaa B$

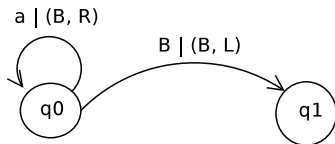
$\vdash Bq_0aaaB$

$\vdash BBq_0aaB$

$\vdash BBBq_0aB$

$\vdash BBBBq_0B$

$\vdash BBBq_1B$



# Configuration-1

- A configuration of a TM:

- Current state
- Symbols on tape
- position of RW head

- A formal specification of configuration:

- $uqv$ , where  $u, v$  are strings on  $\Sigma$ , and  $uv$  is current content on tape,  $q$  is current state, and head is at first symbol of  $v$ .

For example,  $00101q_5011$  where read head points at  $0$  (third digit from end) and state is  $q_5$ .



# Configuration-2

- For Two configurations:

$uaq_i bv$  and  $uq_j acv$ , where  $a, b, c \in \Sigma$  and  $u, v \in \Sigma^*$

$uaq_i bv \vdash uq_j acv$  if  $\delta(q_i, b) = (q_j, c, L)$

$uaq_i bv \vdash uacq_j v$  if  $\delta(q_i, b) = (q_j, c, R)$

- Two special cases:

- The left most cell:

$q_i bv \vdash q_j cv$  for  $\delta(q_i, b) = (q_j, c, L)$

$q_i bv \vdash cq_j v$  for  $\delta(q_i, b) = (q_j, c, R)$

- On the cell with blank symbol:

$uaq_i$  is equivalent to  $uaq_i B$

# Example: of language recognition

Design TM to accept:  $\{a^n b^n, n \geq 0\}$

Let  $M = (Q, \Sigma, \Gamma, \delta, s, H)$ . The algorithm can be specified as:

1.  $M$  replaces left most  $a$  by  $X$ , and then head moves to right until it encounters left most  $b$
2. Replaces this  $b$  by  $Y$ , and then moves left to find the right most  $X$ . Then moves one step right to left most  $a$
3. Repeat Step 2 and 3 in order, i.e., 2, 3, 2, 3, ...
4. When searching for  $b$ , if finds a blank character B (i.e.,  $|a^n| > |b^n|$ ), then  $M$  does not accept  $w$ .
5. If  $a$  is not found but it finds  $b$ , then also  $M$  does not accept, (i.e.,  $|a^n| < |b^n|$ ).
6. After changing last  $b$  to  $Y$ , if  $M$  finds no more  $a$  then it checks that no more  $b$  remains. If this is true then  $a^n b^n$  is accepted by  $M$  i.e.,  $|a^n| = |b^n|$

## Example: of language recognition

$$Q = \{q_0, q_1, q_2, q_3\}$$

$$\Sigma = \{a, b\}$$

$$\Gamma = \{\triangleright, a, b, X, Y, B\}$$

$$s = q_0$$

$$H = \{q_3, q_0\}$$

- Design TM to accept:  $\{a^n b^n, n \geq 0\}$
- 1.  $\delta(q_0, a) = (q_1, X, R)$   
 $\delta(q_1, a) = (q_1, a, R)$ ; skip through  $a$ 's and  
 $\delta(q_1, Y) = (q_1, Y, R)$ ; then  $Y$ 's  
 $\delta(q_1, b) = (q_2, Y, L)$   
 $\delta(q_2, Y) = (q_2, Y, L)$ , traverse through  $Y$ 's and then  
 $\delta(q_2, a) = (q_2, a, L)$ , traverse  $a$ 's

# Example: of language recognition...

TM to accept:  $\{a^n b^n, n \geq 0\}$

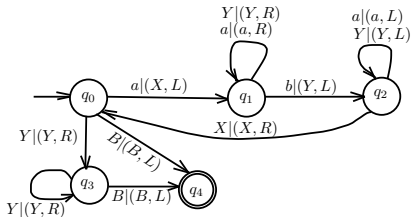


Figure 1: Transition diagram for TM accepting Language  $L = \{a^n b^n \mid n \geq 0\}$ .

- Move from R to L until X is found and start back:  
 $\delta(q_2, X) = (q_0, X, R)$ , right most X is found. Now repeat from 1 else from 2.
- 2.  $\delta(q_0, Y) = (q_3, Y, R)$ , scan through Y's  
 $\delta(q_3, Y) = (q_3, Y, R)$ ,  
 $\delta(q_3, B) = (q_4, B, L)$ , accept  $w$ , and halt.

# Example: of language recognition Dry Run

- TM to accept:  $\{a^n b^n, n \geq 0\}$ , Let  $w = aabb$

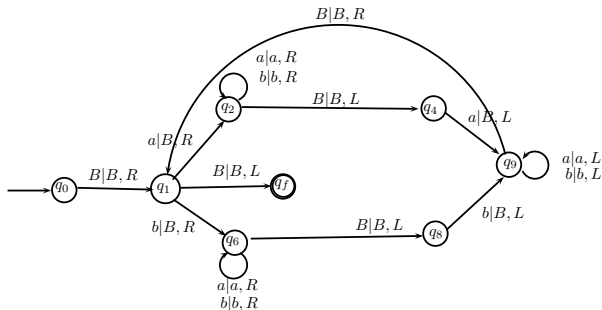
$$\begin{aligned} q_0 a a b b B &\vdash X q_1 a b b B \vdash X a q_1 b b B \vdash X q_2 a Y b B \vdash X q_2 a B b B \\ &\vdash q_2 X a Y b B \vdash X q_0 a Y b B \vdash X X q_1 Y b B \vdash X X Y q_1 b B \\ &\vdash X X q_2 Y Y B \vdash X q_2 X Y Y B \vdash X X q_0 Y Y B \vdash X X Y q_3 Y B \\ &\vdash X X Y Y q_3 B \vdash X X Y q_4 Y B \text{ (accept the input)} \end{aligned}$$

Total number of transitions for  $|w| = n$  are:  $n/2$  forward and  $n/2$  in backward, in each to and fro round, i.e.,  $n$ . Since, in each trip, two symbols are marked, therefore, there will be total  $n/2$  trips, making total transitions:  $n \times n/2 = n^2/2$ . Time complexity,  $O(n^2/2) = O(n^2)$ , which is polynomial (P) time complexity.

# Example: Construct TM to recognize

$$L = \{ww^R \mid w \in \{a,b\}^*\}.$$

Approach: We can traverse  $w$ ; each time if  $a/b$  is found in begin, replace it by  $B$  and also, replace  $a/b$  at end by  $B$ .



Time complexity: Let  $|w| = n$ . Total of trips =  $n/2$ . Number of transitions in successive trips are:  $n + (n - 2) + (n - 4) + \dots + 4 + 2 = n^2/2 - n/2$  in even palindromes, and  $n^2/2 - n/2 + 2$  in odd palindromes, which is  $O(n^2)$  is polynomial time complexity.

# Acceptors v/s Deciders

- Let  $M$  is  $TM$ .
- Three possibilities occur on a given input  $w$ :
- $M$  eventually enters  $q_{acc}$  and therefore halts and **accepts**.  $w \in L(M)$
- $M$  eventually enters  $q_{rej}$  or crashes somewhere.  $M$  **rejects**  $w$ , i.e.,  $w \notin L(M)$
- $M$  never halts its computation and is caught up in an infinite loop. In this case  $w$  is neither accepted nor rejected. However, any string not explicitly accepted is considered to be outside the accepted language.  $w \notin L(M)$
- **Decider:**  $M$  never enters infinite loop(A **recursive language**).