< Lecture 21 >

Chapter 3 : Intro to Turing machines

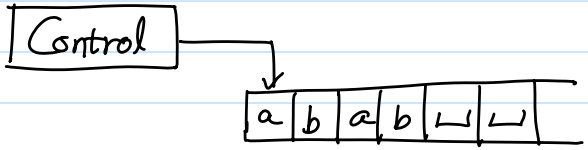
Chapter 4 : What Turing machines can do

Chapter 5 : What Turing machines cannot do.

Some advanced topics if time permits

Turing Machines – Glimpse

- Turing machines are an abstraction of real computers
  (In fact it's more powerful. Why?)
- Uses an infinite tape for memory.
- Has a tape head that can read and write symbols and move around.



- Initially the tape contains only the input string
- tape can be used as scratch storage.
- Has accepting and rejecting states
- Can also run forever.

Difference b/w TM and FA.

- TM can both write on the tape and read from it
- The head can move both left and write
- The tape is infinite
- Special accepting and rejecting states take effect immediately.

Example

$A = \{w\#w \mid w \in \{0,1\}^*\}$

↓

0 1 1 0 0 0 # 0 1 1 0 0 0 ␣ ...

Informal Description of the Example TM

M_1 = "On input w:

1. Zig-zag across the tape to corresponding portions on either side of the # symbol to check whether these positions contain the same symbol.
   If they do not, or if no # is found, reject.
   Cross off symbols as they are checked.
2. When all symbols to the left of # have been crossed off, check for any remaining symbols to the right of #
   If any symbols remain, reject; otherwise, accept

Formal definion of TM

A TM is a 7 tuple $(Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$ where

- Q is a finite set of states
- Σ is the input alphabet not containing ␣
- Γ is the tape alphabet containing ␣ and Σ.
- $\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$
- $q_0 \in Q$ is the start state
- $q_{accept} \in Q$ is the accept state
- $q_{reject} \in Q$ is the reject state $\neq$

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

  $\delta(q, a) = (q', b, R\}$ means

  in state q where the head reads tape symbol a,
  - the machine overwrites a with b,
  - moves to state q'
  - moves the head to the right