Simple Turing Machines

(based on Lewis & Papadimitriou Book)

By

Dr. Neeraj Kumar Sharma

An Introduction (Turing Machine)

- A Turing machine consists of a finite control, a tape, and a head that can be used for reading or writing on that tape.
- Turing machines are not simply one more class of automata, to be replaced later on by a yet more powerful type.
- Any augmented Turing machine (multi-tape, random access machine) are not stronger in terms of computing power than basic Turing machines. Any augmented Turing machine can be converted in to a simple/basic Turing machine.
- So the Turing machines seems to form a stable and maximal class of computational devices, in terms of the computations they can perform

Turing Machine (Formal Definition)

Definition 4.1.1: A Turing machine is a quintuple $(K, \Sigma, \delta, s, H)$, where

K is a finite set of **states**;

Σ is an alphabet, containing the **blank symbol** $\sqcup$ and the **left end symbol** $\triangleright$, but not containing the symbols $\leftarrow$ and $\rightarrow$;

$s \in K$ is the **initial state**;

$H \subseteq K$ is the set of **halting states**;

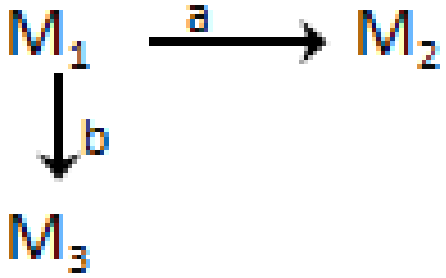
δ, the **transition function**, is a function from $(K - H) \times \Sigma$ to $K \times (\Sigma \cup \{\leftarrow, \rightarrow\})$ such that,

(a) for all $q \in K - H$, if $\delta(q, \triangleright) = (p, b)$, then $b = \rightarrow$

(b) for all $q \in K - H$ and $a \in \Sigma$, if $\delta(q, a) = (p, b)$ then $b \neq \triangleright$.

Combining Turing Machines

- A Turing Machine can be a component of another Turing Machine ("subroutine")
- We will develop a graphical notation to build larger machines for more complex tasks easily. The scheme is hierarchical.
- Combination is possible since all TM's are designed to be "non-hanging" – so the first machine can save something on the left end of the tape.



- The above machine operates until M1 would halt, and if currently scanned symbol is **a** then initiate M_2, otherwise if b then initiate M_3

Combining Turing Machines

- Assumptions, for convenience:
  - From now on, Turing machines can either write a symbol or move head but not both in the same move.
  - All TMs have only one alphabet $\sum$, containing the blank symbol
  - All machines start in this position: #w#
- There are two types of basic machines:
  - Symbol-writing Machines

    Ma or simply a is a symbol writing machine where a
  - Head-moving Machines

    $M_{-->}$ or R is a right moving machine that moves head to one place to the right.

    $M_{<--}$ or L is a light moving machine that moves head to one place to the left.

Basic Machines: Symbol-Writing

- There are $|\sum|$ symbol-writing machines, one for each symbol in $\sum$. Each TM simply writes a specified symbol in the currently scanned tape square and halts.
- Formally, the TM which writes a is

$$M_a = (K, \sum, \delta, s), \text{ where}$$

K = {q} for some arbitrarily chosen state q

s = q and

$\delta(q, b) = (h, a)$ for each $b \in \sum$

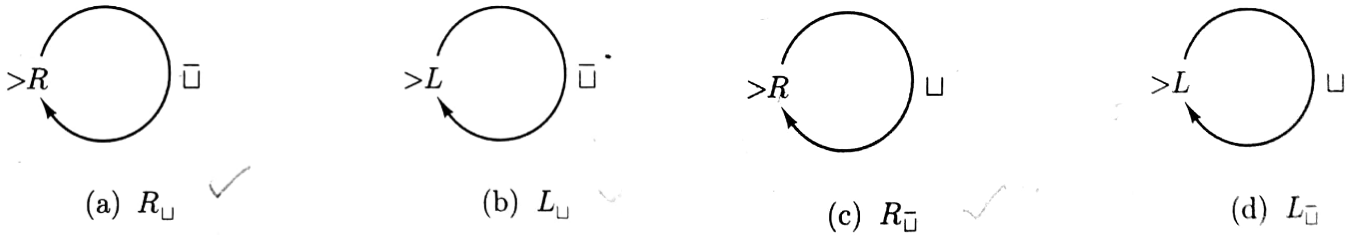
Notation: W_a

Rules for Combining Machines

- Machines may be connected just like a Finite Automaton.
- If two machine are connected, then the first machine has to halt before the other machine starts.

Abbreviations

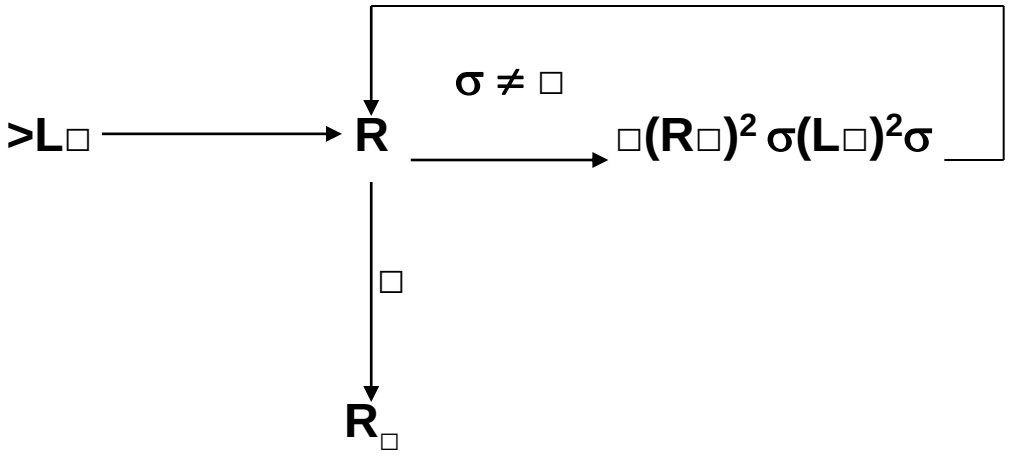
- L/R – A TM that moves the head one cell to the left/right
- R □ - A TM that keeps on moving to the Right until it finds a blank (□)
- L □ - A TM that moves to the Left seeking □
- $R_{\square}$ - A TM that keeps on moving to the Right until it finds a non blank symbol
- $L_{\square}$ - A TM that keeps on moving to the Left until it finds a non blank symbol



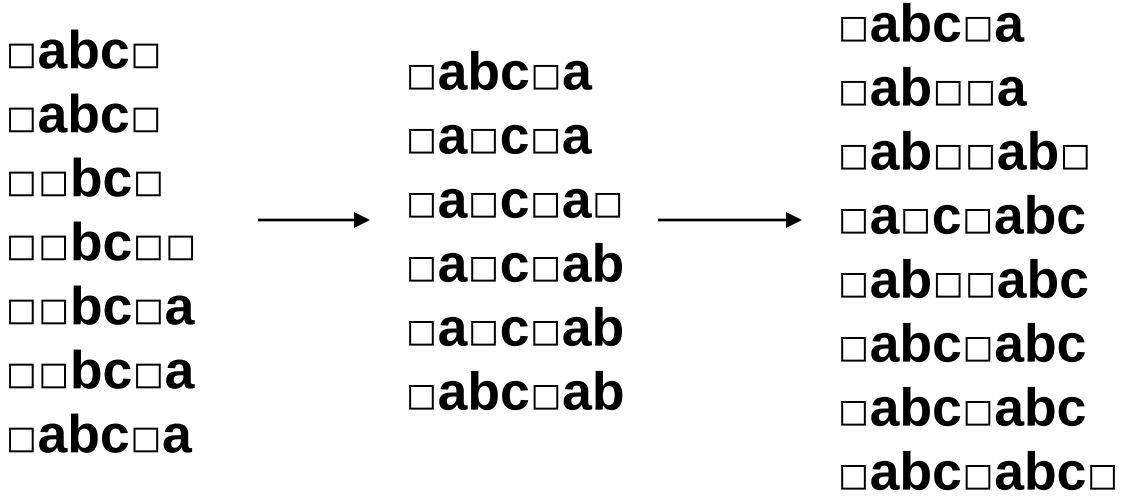
(a) $R_{\sqcup}$ (b) $L_{\sqcup}$ (c) $R_{\bar{\sqcup}}$ (d) $L_{\bar{\sqcup}}$

Example 1

Explain the task performed by the following machine.



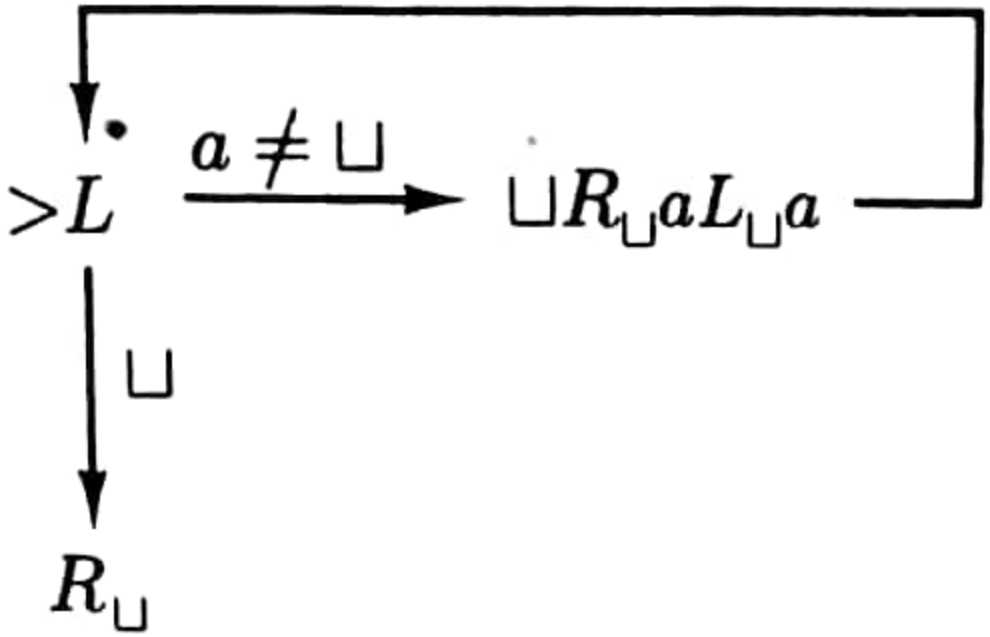
Example 1 (contd.)



Therefore, this is a copying TM that converts □w□ into □w□w□)

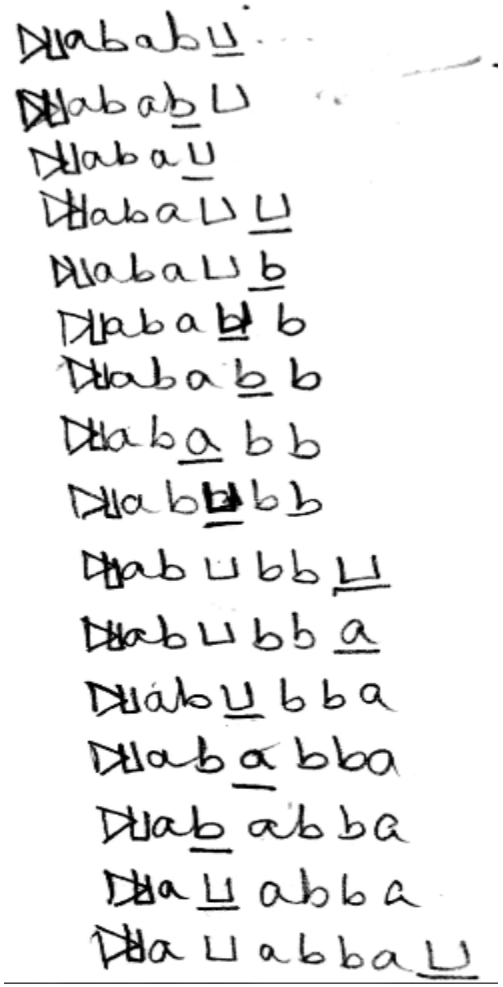
Example 3

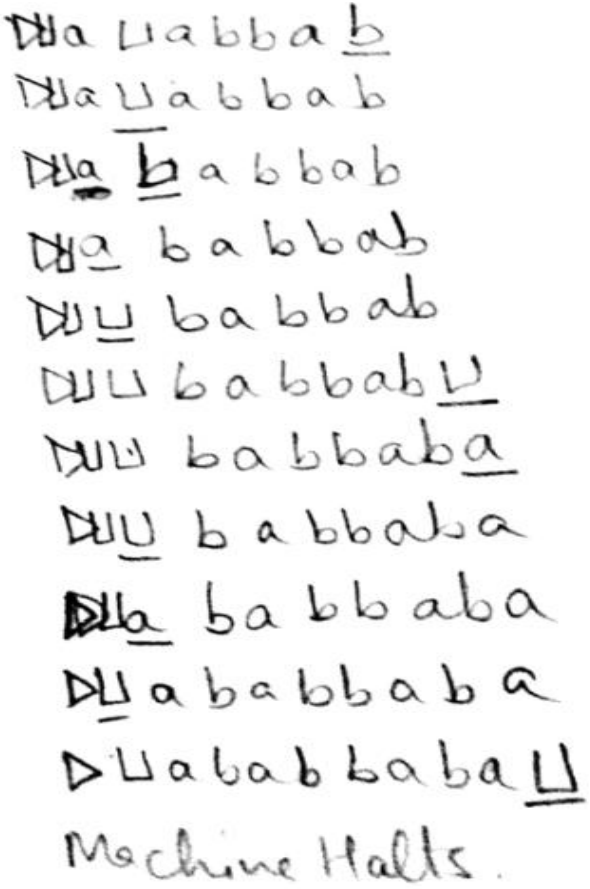
- Trace the output of the following machine



Example 3 (contd.)

- This turning machine reverses a string and in doing so computes the following function from strings in {a, b}* to strings in {a, b}*: $f(w) = ww^R$





Turing Machine Models

- Variants of TM model
  - Two-way infinite tape
  - Multiple tapes
  - Multiple heads on each tape
  - Multi-dimensional tapes, and
  - Combinations of the above.

Are the augmented variants of TMs are more powerful than the simple TM?

- Ans: No

- All "reasonable" extensions including those listed before lead to the same classes of languages or functions. Proved by showing that the basic model can simulate the extensions.