1. Give a formal definition of a Turing machine that on each move can either change the tape symbol or move the read/write head, but not both. Then describe how it can be simulated by a standard Turing machine.

Page No.
Date

1. A universal turing machine is a one that gets simulated by other turing machines. It is known to be automatic function system which has some finite & infinite value.

It has a head which is all to read and write symbols & can move left and right, but only one at a time.

$M = (Q, \Sigma, T, \delta, q_0, \beta, F)$.

Here, M is known as turing machine,
Σ set of symbols on which turing machine is working. contains all alpha $\neq \beta$.
T is known as tape, it also includes B on it.
δ is transaction function, it works on a formula $Q \times T = Q \times T \times \{L, R\}$.

This shows read head on tape will change the state of turing machine q_0, then writes a symbol on tape and can move left or right.

It stores product have multiplication sign represents that all symbols are used in machine.
B → represents blank symbol
q_0 → represents initial state
F → final state

$Q = \{A, B, C, \text{Halt (set of final state)}\}$

Page No.
Date

$T = \{0, 1\}$
$B = \{0\}$
$\Sigma = \{1\}$

Tape Symbol	Current state A WRITE	Tape	Initial	Current state B WRITE	Tape	Next	State C Write		
0	1	L	B	1	R	C	1	R	B
1	1	L	C	1	R	B	1	L	right

Symbol	q_0	q_1	q_2
a	$1Rq_4$	$1Lq_0$	$1Lq_3$
b	$1Lq_2$	$1Rq_1$	$1Rq_4$

eg- $M = (Q, X, \Sigma, \delta, q_0, B, F)$, $Q = \{q_0, q_1, q_2, q_3, q_4\}$
$X = \{a, b\}$, $\Sigma = \{1\}$, $q_0 = \{q_0\}$
B = blank, $F = \{q_f\}$

Symbol	q_0	q_1	q_2
a	$1Rq_1$	$1Lq_0$	$1Lq_3$
b	$1Lq_2$	$1Rq_1$	$1Rq_4$

It moves right & next state is q_1. The transaction $1Lq_2$ implies write symbol is 1. The tap moves left and next is q_2.

From above, $\delta(q_k, a) = (q_2, b)$ or $\delta(q_3, a) = (q_4, R)$.

where q_k is some newly chosen state and first transaction is change transaction and second being more transaction.

2. A two-stack npda is an npda with two independent stacks. A move depends on the tops of the two stacks and results in new values being pushed on these two stacks. Show that two-stack npdas are equivalent to standard Turing machines. As usual, showing equivalency requires showing both directions.

Page No.
Date

2. To see identification, simple construct the main stack the substance of the tape to one scale of the position, and second at the content of the right.

Push the ordering "button of stack" masters on the two stacks.
Push the input to one left stack.
Move everything else to the right stack.

Presently, you can disregard the info & ~~and~~ do everything on content of the stack. You pop to read and push to write. At that point, we can simulate the Turing machine by popping from the right stack and pushing to the left to move right, and the other way around to go left.

In the event that we hit button of the left stack, we carry on appropriately on the off choice that we hit the button part of the right stack, we simply push a blank symbol onto the left.

For the other direction, the idea is to use 2 symbols to mark the top of both states b1, b2, b1, b2 and use the linear space to store the 2 bound stacks that have "overlapped".

Page No.
Date

By further expanding the alphabet, you can store in the same linear space, the original input, the head position of PDA + 2B and an enumeration of the non-deterministic choices.

Then you can scan all non-deterministic choices and simulate the behavior of PDA + 2B using top stack markers.

If you allow epsilon transition, you must run 2 enumerations in parallel in order to check if the PDA + 2B enters the same configuration due to $\epsilon\epsilon$ transitions (i.e. detect if it loops forever).

3. Assume that S_1 and S_2 are countable sets. Prove that $S_1 \cup S_2$ and $S_1 \times S_2$ are also countable.

3. If S_1 & S_2 are both finite set, then ~~so~~ is $S_1 \cup S_2$ & $S_1 \times S_2$ is coutable.

Suppose S_1 is finite & S_2 is a countably infinite. let $a_1, a_2, \ldots$, and $\in S_1$ and $b_1, b_2, \ldots, b_n \in S_2$ that contains every element of sets resp.

So, $a_1, a_2, \ldots, a_n$ & $b_1, b_2, \ldots, b_n$ are an infinite sequence that contains every element of $A \cup B$, so $A \cup B$ is countable.

Similarly, if B is finite and A is countabley infinite, then $A \cup B$ is countable.
Suppose that A & B are both countable, then there are finite sequences that contain every element of the sets respectively.

Thus, $a_1, b_1, a_2, b_2, \ldots, a_n, b_n$ is a sequence that contains every element of $A \cup B$, so its countable.

If S_1 & S_2 are countable, then $S_1 \times S_2$ is countable.
For a fixed $a \in S_1$, let $S_2 a = \{(a,b) \in S_1 \times S_2 \mid b \in S_2\}$

Note that $\forall a \in S_1$ S_2a is the countable union of countable sets.
Since $S_1 \times S_2 = \cup_{a \in S_1} S_2 a$, we have $S_1 + S_2$ is countable.

Page No.
Date

Also, given S_1 & S_2 are 2 countables, then there exists two one-to-one map $f: S_1 \to N$, $g: S_2 \to N$
$h: S_1 \cup S_2 \to N$.

$$h(n) = \begin{cases} 2f(n) & \text{if } n \in S_1 \\ 2f(n)+1 & \text{if } n \in S_2 \end{cases}$$

As both the map f, g are one to one and image of S_1 are in even integer and that of S_2 are in odd integer.

So, map h is also one to one. As there exists a one to one map form, $S_1 \cup S_2$ to N;
Hence, $S_1 \cup S_2$ is countable.

$h: S_1 \times S_2 \to N$.
$h(m,n) = 2^{f(m)} * 3^{g(n)}$

let $h(m_1, n_1) = h(m_2, n_2)$
$2^{fm_1} \cdot 3^{gn_1} = 2^{fm_2} \cdot 3^{gm_2}$

$f(m_1) = f(m_2)$ & $g(n_1) = g(n_2)$.
$\therefore m_1 = m_2$ & $n_1 = n_2$.
as f, g are one to one func^n.
$\therefore (m_1, n_1) = (m_2, n_2)$

So, h is one to one and so $S_1 \times S_2$ is countable as well.