B.Tech.

Fifth Semester Examination

Theory of Automata Computation (CSE-305-F)

Note : Attempt any five questions.

Q. 1. (a) Prove that every primitive recursive function is a computable total function.

Ans. A total $f(n)$ from x to y is a rule which assign to every element of x a unique element of $\dot{y}$.

For Example : If R denote the set of all real number is, the rule 'f' from R to itself given by $f(r) = +\sqrt{r}$ is a partial $f(n)$ since $f(r)$ is not defined as a real number when 'r' is negative. But $g(r) = 2r$ is a total $f(n)$ from 'R' to itself.

In this, we consider total $f(n)$ from X^K to X. Where $X = \{0, 1, 2 .....\}$ or $X = \{a, b\}$. So a primitive recursive $f(n)$ is a total $f(n)$ 'f' from X^K to X is also called a $f(n)$ of 'K' variable and denoted by $f(x_1, x_2, ..., x_K)$. For example, $f(x_1, x_2) = 2x_1 + x_2$ is a $f(n)$ of two variables $f(1, 2) = 4$, 1 and 2 are called arguments and '4' is called a value.

$g(w_1, w_2) = w_1 w_2$ is a $f(n)$ of '2' variables. $(w_1, w_2 \in \Sigma^*)$ $g(ab, aa) = abaa$, ab, aa are called arguments and $abaa$ is a value.

Q. 1. (b) Prove that proper subtraction is a primitive recursive function.

Ans. (i) Searching restrict for a 0, M encounters a blank. Then the no's is $0^m 1 0^n$ have all been changed 1's to is and $n + 1$ of the m 0's have been changed to B. M replaces the $n + 1$ 1's by a0 and nB's. Leaving $m - n$ 0's on its, tape.

(ii) Beginning the cycle, M cannot find a 0 to change to a blank, because the I[st] m 0's already have been changed. Then $n \geq m$, so $m - n = 0$, μ replaces all remaining 1's and 0's by B.

(i) $\delta(q_0, 0) = (q_1, B, R)$

(ii) $\delta(q_1, 0) = (q_1, 0, R)$

$\delta(q_1, 1) = (q_2, 1, R)$

(iii) $\delta(q_2, 1) = (q_2, 1, R)$

$\delta(q_2, 0) = (q_3, 1, L)$

(iv) $\delta(q_3, 0) = (q_3, 0, L)$

$\delta(q_3, 1) = (q_3, 1, L)$

$\delta(q_3, 13) = (q_0, B, R)$

(v) $\delta(q_2, B) = (q_4, B, L)$

$\delta(q_4, 1) = (q_4, B, L)$

$\delta(q_4, 0) = (q_4, 0, L)$

$\delta(q_4, B) = (q_6, 0, R)$

(vi) $\delta(q_0, 1) = (q_5, B, R)$

$$\delta(q_5, 0) = (q_5, B, R)$$
$$\delta(q_5, 1) = (q_5, B, R)$$
$$\delta(q_5, B) = (q_6, B, R)$$

A sample computation of μ on input 0010 is :

$$q_0 0010 \vdash Bq_1 010 \vdash B0q_1 10 \vdash B01q_2 0 \vdash B0q_3 11 \vdash Bq_3 011$$
$$\vdash q_3 B011 \vdash Bq_0 011 \vdash BBq_1 11 \vdash BB1q_2 1 \vdash BB11q_2$$
$$\vdash BB1q_4 1 \vdash BBq_4 1 \vdash Bq_4 \vdash B0q_6$$

On input 0100, μ behaves as follows :

$$q_0 0100 \vdash Bq_1 100 \vdash B1q_2 00 \vdash Bq_3 110 \vdash q_3 B110$$
$$\vdash Bq_0 110 \vdash BBq_5 10 \vdash BBBq_5 0 \vdash BBBBq_5$$
$$\vdash BBBBBq_6$$

Q. 2. (a) Write the unrestricted grammar for generating the language

$$L = \{a^i b^i c^i / i \geq 1\}$$

Ans. Before designing the required turing machine M_1. Let us evolve a procedure for processing the input string aabbcc. After processing, we require the ID to be of the form $bbbbbbq_7$. The processing is done by using five steps :

Step 1 : q_1 is the initial state. The R/w head scan the leftmost a, replaces 1 by b & moves to the rights μ enter q_2.

Step 2 : On scanning the leftmost 2, the R/w head replaces 2 by b & moves to the right M enter q_3.

Step 3 : On scanning the leftmost 3, the R/w head replaces 3 by b & moves to the right M enter q_4.

Step 4 : After scanning the rightmost 3, the R/w heads moves to the left until it find the leftmost 1 As a result, the leftmost 1, 2, and 3 are replaced by b.

$$q_1 112233 \vdash bq_2 12233 \vdash b1q_2 2233 \vdash b1bq_3 233$$
$$\vdash b1b2q_3 33 \vdash b1b2bq_4 3 \vdash b1b2q_5 b3 \vdash b1bq_5 2b3 \vdash b1q_5 b2b3$$
$$\vdash bq_5 1b2b3 \vdash q_6 b1b2b3 \vdash bq_1 1b2b3 \vdash bbq_2 b2b3 \vdash bbbq_2 2b3$$
$$\vdash bbbbq_3 b3 \vdash bbbbbq_3 3 \vdash bbbbbbq_4 b \vdash bbbbbq_7 bb$$

Thus, $q_1 112233 \vdash^* q_7 bbbbbb$

	Σ			
P.S.	**1**	**2**	**3**	**b**
$\rightarrow q_1$	bRq_2			bRq_1
q_2	$1Rq_2$	bRq_3		bRq_2
q_3		$2Rq_3$	bRq_4	bRq_3
q_4			$3Lq_5$	bLq_7
q_5	$1Lq_6$	$2Lq_5$		bLq_7
q_6	$1Lq_6$			bLq_5
q_7				bRq_1

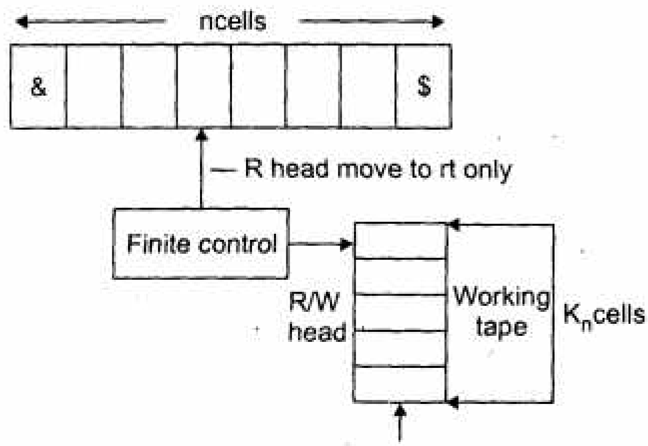
Q. 2. (b) Show that if $L \subseteq \Sigma^*$ is a context sensitive language, there is a linear bounded automation accepting L.

Ans. (i) The set of context-sensitive language is accepted by the model.

(ii) The infinite storage is restricted in size but not in accessibility to the storage in comparison with the turing machine model. It is called Linear Bounded Automaton (LBA) because a linear $f(n)$ is used to restrict the length of the tape.

(iii) A LBA is a non-deterministic, turing machine which has a single tape whose length is not infinite but bounded by a linear $f(n)$ of the length of the input string.

$$M = (Q, \Sigma, \Gamma, \delta, q_0, b, \&, \$, F)$$



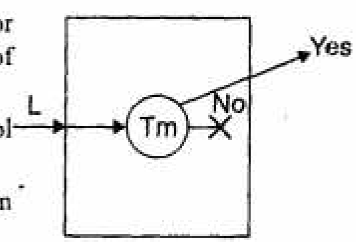
Q. 3. (a) What is undecidability? Show that the problem "Given an Arbitrary Turing machine M and arbitrary string w, does M halts on w" is undecidable.

Ans. Undecidability : A problem whose language is recursive is said to be decidable. Otherwise the problem is undecidable. That is a problem is undecidable if there is no algorithm that takes as input an instance of the problem and determines whether the answer to that instance is 'Yes' or 'No' :

(i) The theory of undecidability is concerned with the existence or non-existence of algorithms for solving problems with an infinity of instance.

(ii) L is undecidable if turing machine halts on applying input symbol 'w'.

If 'w' accepts then it produce answer 'Yes'. If 'w' does not accepts then it halts forever.



Q. 3. (b) Design a TM for accepting the set of strings with an equal number of 0^a s and 1^a s.

Ans. Design a TM for the Following language :

$$L = \{a^n b^n / n \geq 1\}$$

We are going to design TM which accepts a set of strings in which every string start with 'a' followed by any number of a's every string ends in equal number of b's as a's. Number a's is encountered after first 'b' is read.

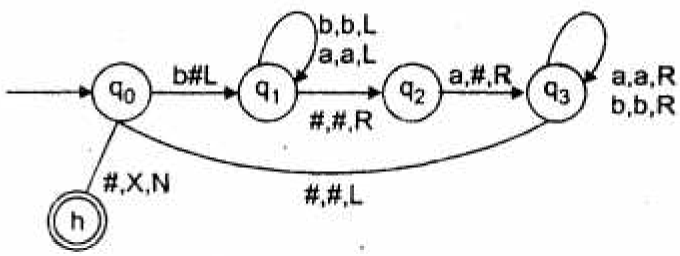
$\in$, a, aa, bb, abb, $aabb$, $aaabbb$ and so on

$$T_m = \{Q, \Sigma, \Gamma, \delta, q_0, h\}$$
$$Q = \{q_0, q_1, q_2, q_3, h\}$$
$$\Sigma = \{a, b\}$$
$$\Gamma = \{a, b, \#\}$$
$$q_0 = \text{Initial state}$$
$$h = \text{Halt state}$$

The moves which also unaddressed, reads towards rejecting states :

$$w = \# \frac{ab}{q_0} \#$$

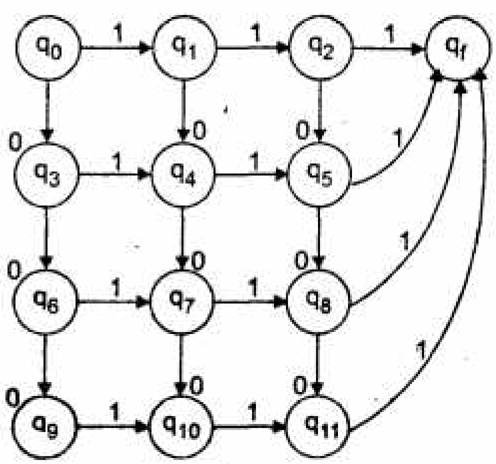
$$\# abq_0 \vdash \# aq_1 \#$$



$$\vdash \# q_1 a \# \vdash \# aq_2 \#$$
$$\vdash \#\#\# q_3 \vdash \#\# q_0 \#$$
$$\vdash \# Yh \#$$

Q. 4. (a) Construct a Finite Automata which will accept those string of binary number which are divisible by 3.

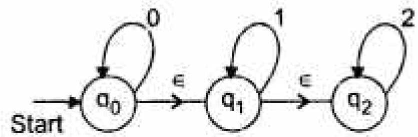
Ans.



Test this finite automata with taking a string as example.

Process the string, if it reaches to the final state by processing the input string then the number is divisible by 3.

Q. 4. (b) Remove the ∈-moves from following NFA :



Ans. Apply ∈ - closure of initial states

$$\in\text{-closure}(q_0) = \{q_0, q_1, q_2\} - A$$

Apply input symbols on set A.

$$\in\text{-closure}(\delta(A, 0)) = \in\text{-closure}(q_0) = \{q_0, q_1, q_2\} - A$$
$$\in\text{-closure}(\delta(A, 1)) = \in\text{-closure}(q_1) = \{q_1, q_2\} - B$$
$$\in\text{-closure}(\delta(A, 2)) = \in\text{-closure}(q_2) = \in\text{-closure}\{q_2\} - C$$

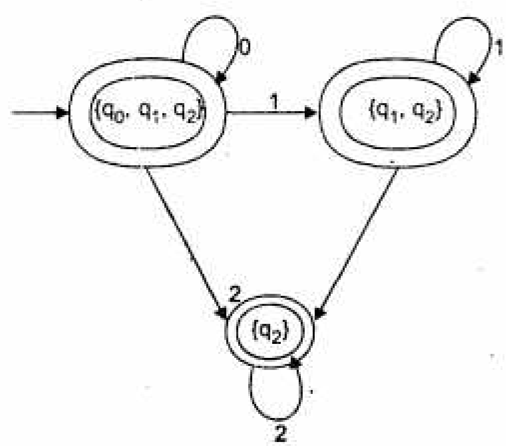
Apply input symbol on set 'B'.

$$\in\text{-closure}(\delta(B, 0)) = \in\text{-closure}(\varphi) = \varphi$$
$$\in\text{-closure}(\delta(B, 1)) = \in\text{-closure}\{q_1\} = \{q_1, q_2\} - B$$
$$\in\text{-closure}(\delta(B, 2)) = \in\text{-closure}\{q_2\} = \{q_2\} - C$$

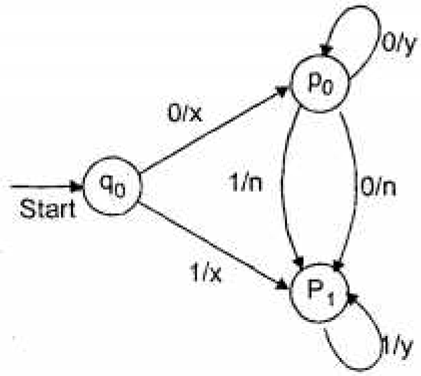
Apply input symbol on set C

$$\in\text{-closure}(\delta(C, 0)) = \varphi$$
$$\in\text{-closure}(\delta(C, 1)) = \varphi$$
$$\in\text{-closure}(\delta(C, 2)) = \{q_2\} - C$$

	0	1	2
A	A	B	C
B	φ	B	C
C	φ	φ	C



Q. 5. (a) Convert the following mealy machine to Moore machine :



Ans.

State	a = 0 State	a = 0 Output	a = 1 State	a = 1 Output
q_0	p_0	x	p_1	x
p_0	p_0	y	p_1	x
p_1	p_0	x	p_1	y

State	a = 0 State	a = 0 Output	a = 1 State	a = 1 Output
q_0	p_0x	x	p_1x	x
p_0x	p_0y	y	p_1x	x
p_0y	p_0y	y	p_1x	x
p_1x	p_0x	x	p_1y	y
p_1y	p_0x	x	p_1y	y

State	States a = 0	States a = 1	Output
q_0	p_0x	p_1x	x
p_0x	p_0y	p_1x	x
p_0y	p_0y	p_1x	y
p_1x	p_0x	p_1y	x
p_1y	p_0x	p_1y	y

Q. 5. (b) What are the limitations of FSM.

Ans. The limitation of finite state machine is the word finite. The memory of finite automata is finite & the counting ability is also finite.

It cann't recognise all the string of upto 'n' number of times.

For Example : (i) Suppose we have a string of d $L:a^n b^n$: The automata corresponding to this $R.E$ is impossible because finite automata cannot recognise any number of 0's and any number of 0's because FA have limited memory.

(ii) It cann't recognise all the string.

Q. 6. Show that if there are strings x and y in the language L so that x is a prefix of y and $x \neq y$, then no DPDA can accept L by empty stack.

Ans. $L = \{x^m y^n / m \neq n\}$

Suppose we pass the string aaabb

- Pass this string to stack one 'a' remain in the stack, so this string is not accepted by PDA.

$$(q_0, a, z) = (q_0, az)$$
$$(q_0, b, z) = (q_0, bz)$$
$$(q_0, a, az) = (q_0, aaz)$$
$$(q_0, b, az) = (q_0, \lambda z)$$
$$(q_0, a, bz) = (q_0, \lambda z)$$
$$(q_0, b, bz) = (q_0, bbz)$$
$$(q_0, \lambda, z) = (q_f, z)$$

But by these transition there is one 'a' remain in the stack 1.

So this DPDA does not exist for this type of string.

Q. 7. (a) Write and explain the algorithm for minimizing the number of states of on DFA.

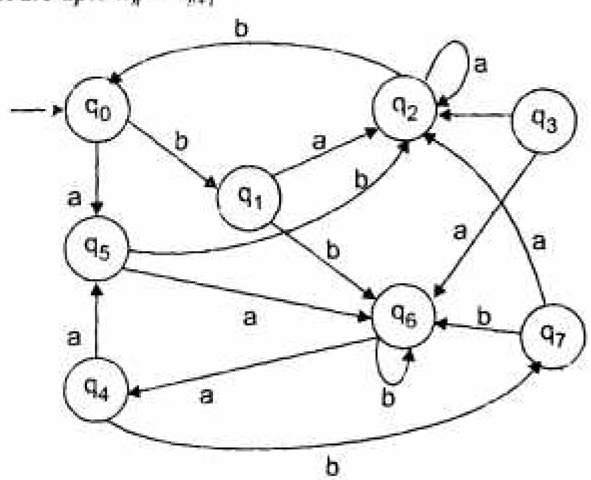
Ans. In the minimizing of FA we use minimizing algorithm :

(i) π_0 = {Final state} {all state except final state}

(ii) Construct π_1 from π_0

(iii) Construct π_2 from π_1 upto $n = 0, 1, 2, \ldots \ldots\ldots, n$.

(iv) Repeat this procedure upto $\pi_n = \pi_{n+1}$



	a	b	π_1	π_2	π_3
$\rightarrow q_0$	q_5	q_1	2 2 √	4 3 —	1 3
q_1	q_2	q_6	1 2 √	1 2 —	1 2

q_2	q_2	q_0	1 2 √	1 2 —	1 2
q_3	q_6	q_2	2 1	2 1	2 1
q_4	q_5	q_7	2 2 √	4 3 —	4 3
q_5	q_6	q_2	2 1	2 1	2 1
q_6	q_4	q_6	2 2 √	2 2	2 2
q_7	q_2	q_6	1 2 √	1 2 —	1 2

$$\pi_0 = \underset{1}{\{q_2\}}\underset{2}{\{q_0, q_1, q_3, q_4, q_5, q_6, q_7\}}$$

$$\pi_1 = \underset{1}{\{q_2\}}\underset{2}{\{q_0, q_4, q_6\}}\underset{3}{\{q_1, q_7\}}\underset{4}{\{q_3, q_5\}}$$

$$\pi_2 = \underset{1}{\{q_2\}}\underset{2}{\{q_0, q_4\}}\underset{3}{\{q_1, q_7\}}\underset{4}{\{q_3, q_5\}}\underset{5}{\{q_6\}}$$

$$\pi_3 = \underset{1}{\{q_2\}}\underset{2}{\{q_6\}}\underset{3}{\{q_0, q_4\}}\underset{4}{\{q_1, q_7\}}\underset{5}{\{q_3, q_5\}}$$

	a	b
$\rightarrow [q_0, q_4]$	$[q_3, q_5]$	$[q_1, q_7]$
$[q_1, q_7]$	$[q_2]$	$[q_6]$
$[q_3, q_5]$	$[q_6]$	$[q_2]$
$[q_2]$	$[q_2]$	$[q_0, q_4]$
$[q_6]$	$[q_4, q_0]$	$[q_6]$