(Following Paper ID and Roll No. to be filled in your Answer Book)

PAPER ID : 1071 Roll No. | | | | | | | | | | |

B. Tech.

(SEM. IV) THEORY EXAMINATION 2010-11

THEORY OF AUTOMATA AND FORMAL LANGUAGES

Time : 3 Hours *Total Marks : 100*

Note :— (1) Attempt **ALL** questions.

(2) All questions carry equal marks.

(3) Notations/Symbols/Abbreviations used have usual meaning.

(4) Make suitable assumptions, wherever required.

1. Attempt any **two** parts of the following :

(a) Define Nondeterministic finite automata (NFA). Design a deterministic finite automata (DFA) over $\Sigma = \{a, b\}$ with minimum number of states which accepts all the strings that ends with babb.

(b) Define Mealy machine. Convert the following Moore machine into equivalent Mealy machine :

Present State	Next State Input 0	Next State Input 1	Output
$\rightarrow q_0$	q_0	q_1	Y
q_1	q_2	q_3	N
q_2	q_4	q_0	N
q_3	q_1	q_2	N
q_4	q_3	q_4	N

(c) Write the steps for minimizing the states in a DFA. Minimize the number of states in the following DFA :

Present State	Next State	
	Input 0	Input 1
$\rightarrow q_0$	q_1	q_3
q_1	q_0	q_3
q_2	q_1	q_4
q_3	q_5	q_5
q_4	q_3	q_3
q_5	q_5	q_5

Given that q_3 and q_5 are final states.

2. Attempt any **four** parts of the following :

(a) Write the regular expression for the following languages :

(i) The set of all strings of 0's and 1's in which every 0 is followed by 11.

(ii) The set of all strings of 0's and 1's in which the number of 0's is even.

(b) Obtain the NFA without epsilon transition corresponding to the following regular expression :

$00(0^* + 1^*)^*\ 11.$

(c) Obtain the regular expression for the following finite automata having q_0 and q_2 as final states :

Present State	Next State	
	Input a	Input b
$\rightarrow q_0$	q_0	q_1
q_1	q_0	q_2
q_2	q_0	q_1

(d) Prove that if L and M are regular languages then intersection of L and M is also regular langauge.

(e) Discuss the Chomsky hierarchy of the languages.

(f) Prove that every language defined by a regular expression is also accepted by some finite automata.

3. Attempt any **two** parts of the following :

(a) State the pumping lemma for regular expressions. Use the pumping lemma to prove that the language L is not regular. L is defined as follows :

$$L = \{0^n 1^{2n} \mid n \text{ is non-negative integers}\}.$$

(b) Convert the following grammar into Greibach Normal Form (GNF) :

$$S \rightarrow AA \mid 0$$

$$A \rightarrow SS \mid 1$$

(c) (i) What do you understand by ambiguous grammar ? Show that the following grammar is ambiguous :

$$S \rightarrow S + S \mid S * S \mid a$$

(ii) Simplify the following context free grammar to an equivalent context free grammar that do not have any useless symbol, null production and unit production :

$$S \rightarrow aSa \mid bSb \mid \in$$

$$A \rightarrow aBb \mid bBa$$

$$B \rightarrow aB \mid bB \mid \in$$

S is the start symbol.

4. Attempt any **two** parts of the following :

(a) Define Push Down Automata (PDA). Construct a PDA which accepts the language L given by :

$L = \{a^m b^n m^n \mid m \text{ and } n \text{ are non-negative integers}\}$.

(b) Obtain a context free grammar that generates the langauge accepted (by final state) by the NPDA with following transitions :

$$\delta(q_0, a, Z) = \{(q_0, AZ)\}$$

$$\delta(q_0, a, A) = \{(q_0, A)\}$$

$$\delta(q_0, b, A) = \{(q_1, \in)\}$$

$$\delta(q_1, \in, Z) = \{(q_2, \in)\}$$

q_0 is the initial state and q_2 is the final state.

(c) (i) Construct a Push Down Automata that accepts the language generated by the grammar with following productions :

$$S \rightarrow aSA \mid a$$

$$A \rightarrow bB$$

$$B \rightarrow b$$

(ii) Prove that context free languages are closed under star-closure.

5. Attempt any **two** parts of the following :

(a) Define Turing machine. Design a Turing machine that accepts the language L over {a, b, c} defined as follows :

$$L = \{wcw \mid w \in (a + b)^*\}.$$

(b) Discuss various variations of Turing machine.

(c) (i) Write short notes on the halting problem of Turing machine.

(ii) Differentiate between recursive language and recursively enumerable language.