Roll No. | | | | | | | | |

B.E / B.Tech (Full Time) DEGREE END SEMESTER EXAMINATIONS, APRIL / MAY 2014

COMPUTER SCIENCE AND ENGINEERING

Fifth Semester

CS 282 / CS 9302 – THEORY OF COMPUTATION

(Regulations 2004/2008)

Time : 3 Hours | Answer ALL Questions | Max. Marks 100

PART-A (10 x 2 = 20 Marks)

1. Convert the following $\in$-NFA to DFA.

δ	$\in$	a	b
$\rightarrow p$	{r}	{q}	{p, r}
q	ϕ	{p}	ϕ
* r	{p, q}	{r}	{p}

2. Design a nondeterministic finite automaton to accept the set of all strings containing either 101 or 110 as substring, over the alphabet {0,1}*.
3. Convert the following regular expression to NFA with $\in$- transitions: (01+1)*.
4. Write a regular expression that recognizes the set of all strings that do not contain the substring 1011 over the alphabet (0+1)*.
5. Show that the following grammar is ambiguous: $S \rightarrow SbS \mid a$
6. Consider the context-free grammar given below. Give the leftmost derivation for the string *bbaa* using the grammar.

$$S \rightarrow bS \mid aT \mid \in$$
$$T \rightarrow aT \mid bU \mid \in$$
$$U \rightarrow aT \mid \in$$

7. If L is a context-free language, then prove that L^R is also a CFL.
8. Consider the Turing machine M = ({q_0, q_1, q_2, q_f }, {0, 1}, {0, 1, B}, δ, q_0, B, { q_f }. Describe the language $L(M)$ if δ consists of the following set of rules:

$\delta(q_0,0) = (q_1, 1, R)$
$\delta(q_1,1) = (q_0, 0, R)$
$\delta(q_1,B) = (q_f, B, R)$

9. Prove that if a language L and its complement are recursively enumerable, then L is recursive.

10. Give the binary code for the Turing machine $M = (\{q_0, q_1, q_2, q_3, q_4\}, \{0, 1\}, \{0, 1, B\}, \delta, q_0, B, q_4)$ where δ is given as

$\delta(q_0,0) = (q_1, 0, R)$
$\delta(q_1,0) = (q_2, 0, R)$
$\delta(q_2,0) = (q_3, 1, R)$
$\delta(q_2,1) = (q_3, 1, L)$
$\delta(q_2,B) = (q_3, 1, R)$
$\delta(q_3,0) = (q_2, 1, L)$
$\delta(q_3,1) = (q_4, 1, R)$
$\delta(q_3,B) = (q_2, 1, L)$

Part – B (5 x 16 = 80 marks)

11. (i) Prove using structural induction that every tree has one more node than it has edges. (4)
(ii) Construct a DFA that will accept the set of all strings which do not contain 001 over the alphabet {0,1}. (6)
(iii) If $D = (Q_D, \Sigma, \delta_D, \{q_0\}, F_D)$ is the DFA constructed from NFA $N = (Q_N, \Sigma, \delta_N, q_0, F_N)$ by the subset construction, then prove that $L(D) = L(N)$. (6)

12. a) (i) Prove using Pumping lemma that the language $\{0^n \mid n \text{ is a perfect square}\}$ is not regular. (8)
(ii) Prove that regular languages are closed under union and intersection. (8)

OR

b) (i) Consider the DFA given by the following transition table. Construct the minimum state equivalent DFA using the table filling algorithm. (8)

δ	0	1
$\rightarrow q_1$	q_2	q_3
q_2	q_3	q_5
$* q_3$	q_4	q_3
q_4	q_3	q_5
$* q_5$	q_2	q_5

(ii)Consider the DFA given below: Give all the regular expressions $R_{ij}^{(2)}$. Simplify the expressions as much as possible. (8)

δ	0	1
$\rightarrow q_1$	q_2	q_1
q_2	q_2	q_3
$* q_3$	q_3	q_2

13. a) (i) Find a context-free grammar generating the language $\{a^i b^j c^k \mid i = j+k\}$ (10)
(ii) Convert the grammar

$S \rightarrow S0S1S0S \mid S0S0S1S \mid S1S0S0S \mid \epsilon$

to a PDA that accepts the same language by empty stack (6)

OR

b) (i) If $L = N(P_N)$ for some PDA $P_N = (Q, \Sigma, \Gamma, \delta_N, q_0, Z_0)$ that accepts by empty stack, then prove that there is a PDA P_F that accepts by final state such that $L = L(P_F)$.

(6)

(ii) Design a push down automaton to accept the language $\{0^n 1^m 0^m 1^n \mid m, n \geq 1\}$ over the alphabet $\{0,1\}^*$. (10)

14. a) (i) Simplify the following grammar and convert to Chomsky Normal Form. (8)

$S \rightarrow ABC \mid BaB$
$A \rightarrow aA \mid BaC \mid aaa$
$B \rightarrow bBb \mid a \mid D$
$C \rightarrow CA \mid AC$
$D \rightarrow \epsilon$

(ii) If a context free grammar G_1 is constructed after eliminating ϵ–productions from the context free grammar G, prove that $L(G_1) = L(G) - \epsilon$. (8)

OR

b) (i) Design a Turing machine that will accept the language $\{ww^R \mid w$ is any string of 0s and 1s$\}$. (8)

(ii) Design a Turing machine that will increment the binary number given as input by one. (8)

15. a) (i) Define L_d the diagonal language. Prove that L_d is not recursively enumerable. (6)

(ii)Prove that L_u, the universal language is recursively enumerable but not recursive. (10)

OR

b) (i) If L is a recursive language, prove that the complement of L is also recursive (5)

(ii) Define L_{ne}, the nonempty language. Prove that L_{ne} is recursively enumerable. (6)

(iii) Define Post's Correspondence Problem. Explain with an example. (5)