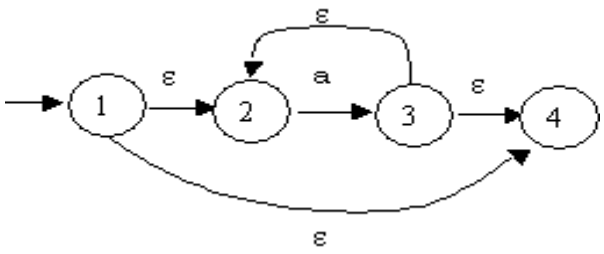
M.Tech (CSE) 1st Semester Examination -2019

Sub: Theory of Computation

Full Mark – 70 **Time: 3Hrs**

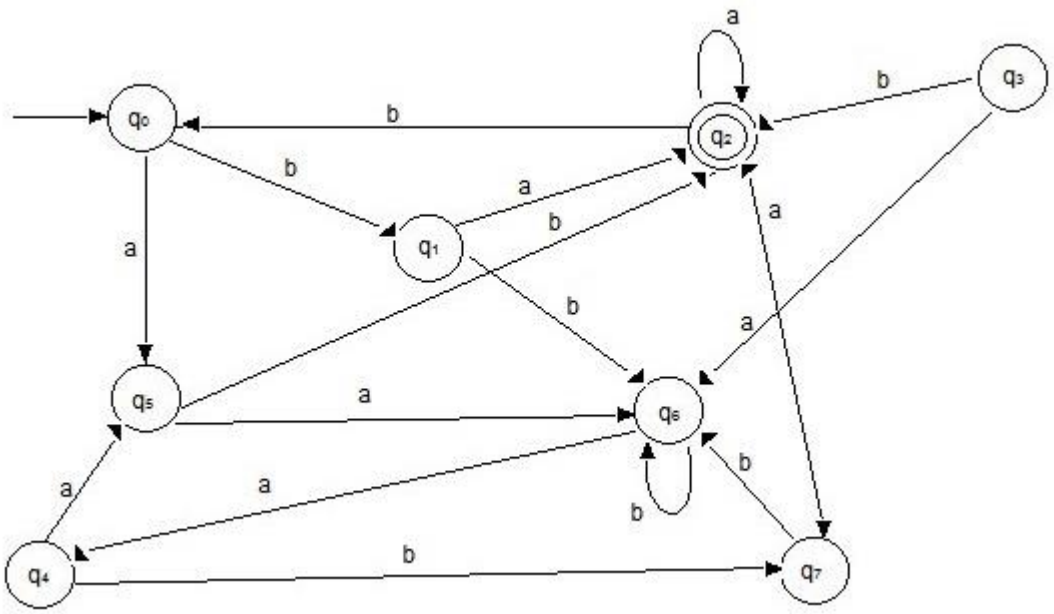
(Answer all questions. Figure in the right hand margin indicates marks.)

1. (a) Prove that if L is accepted by NFA, there exist a DFA which accepts L. (7)
   (b) Convert the ε-NFA to DFA. (7)



OR

(c) Design a DFA which accepts set of all strings divisible by 5 for binary alphabet. (7)
(d) Minimize the following DFA (7)



2. (a) Construct NFA for the RE $(a/ab)^* abbb$. Also convert it into its DFA form. (7)

(b) What is Pumping Lemma? Show that L= $\{0^n 1\ 0^{2n} / n \geq 0\}$ is regular or not. (7)

OR

(c) Construct NFA and Convert it to its equivalent DFA for the RE $1(1+10)^* + 10(0+01)^*$. (7)

(d) Prove that regular set are closed under union, Kleene closure and intersection. (7)

3. (a) Define CFG. Derive LMD and RMD for the following grammar for the string 00110101

S→0B/1A A→0/0S/1AA B→1/1S/0BB

Also check the grammar is ambiguous or not. (7)

(b) Design a PDA M to accept the language L= $\{wcw^R / w \in (a,b)^+\}$. (7)

OR

(c) Define the Pumping Lemma for CFL. Prove that CFL are closed under Substitution, Union, and Concatenation. (7)

(d) Convert the following grammar to CNF form

S→bA/aB A→bAA/aS/a B→aBB/bS/b (7)

4. (a) Design a TM over ∑={1} to accept the language L={1^m / m is odd} (7)

(b) Design a TM over ∑={0,1} to accept the language L={$\{0^n 1^m / m > 0\}$. (7)

OR

(c) Design TM to compute m+n where m and n are positive integers. (7)

(d) Design a TM that accepts all palindromes over the alphabet ∑={a, b}. (7)

5. (a)) Prove that the Complement of a recursive language is recursive and the union of two recursively enumerable language is recursively enumerable . (7)

(b) Prove that if L_1 and L_2 are two recursive language then their union and intersection is also recursive language. (7)

OR

(c) Show that PCP is undecidable if we limit the alphabet to ∑={0,1}. (7)

(d) Prove that the language L_{ne} is recursively enumerable and the language L_e is not recursive enumerable. (7)