Date:19/04/2022 Enrolment No.:_____

RK UNIVERSITY

B.TECH. / SEM-VI / REGULAR / APRIL-2022

CE621: THEORY OF COMPUTATION

Time: 10:30 AM TO 01:30 PM **Total Marks: 100**

Instructions:

1. **Attempt all questions.**
2. **Make suitable assumptions wherever necessary.**
3. **Figures to the right indicate full marks.**
4. **Programmable calculator is not permissible.**

SECTION – I

Q.1 (a) Select the most appropriate option: (Each of one mark) **06**

1. Which of the following is not true?
   a. Power of deterministic automata is equivalent to power of non-deterministic automata.
   b. The power of deterministic pushdown automata is equivalent to the power of non-deterministic pushdown automata.
   c. The power of the deterministic Turing machine is equivalent to the power of the non-deterministic Turing machine.
   d. All the above

2. Which of the following is false?
   a. The languages accepted by FA's are regular languages.
   b. Every DFA is an NFA.
   c. There are some NFA's for which no DFA can be constructed.
   d. If L is accepted by an NFA with e transition, then L is accepted by an NFA without e transition

3. $\sum = \{a, b\}$
   $r\square = a\ (a + b)^*$ and
   $r\square = b\ (a + b)^*$.
   a. $L(r\square) = L(r\square) = \sum^*$
   b. $L(r\square) \cap L(r\square) = \{\in\}$
   c. $L(r\square) \cup L(r\square) = \sum^*$
   d. $L(r\square) \cup L(r\square) \cup \{\in\} = \sum^*$

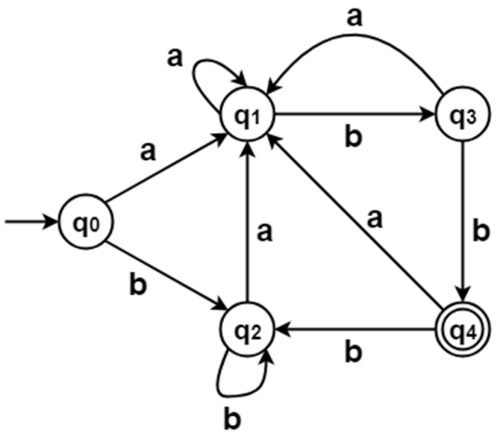
4. Which of the following regular expression corresponds to the language of all strings over the alphabet (a, b) that do not end with ab?
   a. $(a + b)^*\ (aa + ba + bb)$
   b. $(a + b)^*\ (aa + ba + bb) + a + b + \in$
   c. $b^*\ ab^*\ a$
   d. $b^*\ aa\ b^*$

5. What is regular expression corresponding to the language of strings of even lengths over the alphabet of (a, b)?
   a. (aa + bb + ba + ab)*
   b. (aa + bb)
   c. (ab + bb + ba)*
   d. a*b*a*b*
6. How many minimum a number of states will be there in the DFA accepting all strings (over the alphabet {a, b}) that do not contain two consecutive a's?
   a. 2
   b. 3
   c. 4
   d. 5

(b) Answer the following questions: (Each of two mark) **10**

1. If L(r) ={▪ ,x,xx,xxx,xxxx,xxxxx} what is r?
2. Create RE for L = { w ϵ {a,b} * | total no. of a's is divisible by 3 }
3. Write Regular Expression for L = {w ε {a,b}* | w does not contain 'bb' as substring. }
4. What is Kleen Clouser?
5. Design RE for L1={w Є (0,1)* | w end with 1 and does not contain substring '00'}

Q.2 (a) Minimize the Given DFA. **06**

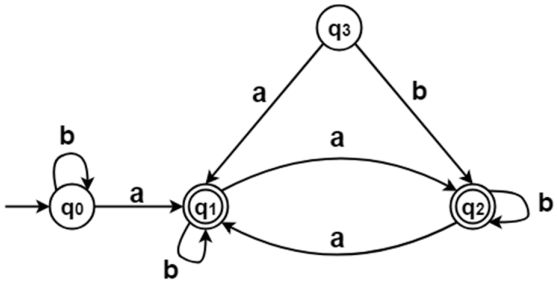


(b) Design DFA for string having 101 or 110 as a substring. **05**
W □ {0+1}*

(c) Design DFA for string where starting and ending symbols must be different. W □ {0+1} * **05**

OR

Q.2 **(a)** Minimize the Given DFA. **06**



(b) Differentiate between DFA and NFA with suitable examples. **05**

(c) Prove given RE is equivalent or not. **05**

$(PQ)^* P = P(QP)^*$

Q.3 **(a)** Design DFA over Σ= {a,b} **06**

1] $(ab)^n$, n ≥ 0

2] $(ab)^n$, n ≥ 1

(b) Draw NFA-Ɛ diagram for given RE. **06**

a) $(a+b)^*$ abb

b) $a(a+b)^*b$

(c) Prove given RE is equivalent or not. **06**

$(0^* 1^{*)} * = (0+1)^*$

OR

Q.3 **(a)** Design DFA for language L = {x ε {a,b}* | x does not have 'aba' as a substring } **06**

(b) Design NFA for a given language. Also, convert NFA into DFA. **06**

L = {x □ {a,b}* | every string has 'abb' as substring }

(c) Draw NFA-Ɛ diagram for given RE. **06**

a) 00(11)*00

b) 1(01+10)*1

SECTION – II

Q.4 **(a)** **Select the most appropriate option: (Each of one mark)** **06**

1. Which of the following automata takes stack as auxiliary storage?

a. Finite automata
b. Push down automata
c. Turing machine
d. All of the mentioned

2. The following move of a PDA is on the basis of:

a. Present state
b. Input Symbol
c. Present state and Input Symbol
d. None of the mentioned

3. A string is accepted by a PDA when
   a. Stack is not empty
   b. Acceptance state
   c. All of the mentioned
   d. None of the mentioned

4. A Pushdown automaton employs _____ data structure.
   a. Queue
   b. Linked List
   c. Hash Table
   d. Stack

5. Push down automata accepts _____ languages.
   a. Type 3
   b. Type 2
   c. Type 1
   d. Type 0

6. Which of the operations are eligible in PDA?
   a. Push
   b. Delete
   c. Insert
   d. Add

(b) Answer the following questions: (Each of two mark) **10**

1. What is PDA?
2. Differentiate Finite Automata and Push Down Automata.
3. Why Stack is required in Push Down Automata?
4. Define Grammar.
5. Why every Regular Grammar is CFG but every CFG is not RG?

Q.5 **(a)** E→E+E | E*E | (E) |id **06**
Validate string " (id+(id*id)) "using LMD, RMD, Reduction, and Parse Tree.

(b) Design PDA to accept the language L={ $a^n . b^n \mid n >= 1$}. Assume any string related to language and prove it. **05**

(c) Check the given Grammar is Ambiguous or not. **05**
S→AA
A→AAA |a| bA | Ab
Consider the String = "babbab"

OR

Q.5 **(a)** E→E+E | E*E | (E) |id **06**
Validate string " ((id*id)+id) "using LMD, RMD, Reduction, and Parse Tree.

(b) Design PDA to accept the language L={ w ϵ $(a+b)^*$ | n_a (w) >= n_b(w | n>=1) }. The number of a's is greater than number of b's in all the strings. Assume any string related to language **05**

(c) Simplified the Given Grammar. **05**

S→AB
A→a
B→C|b
C→D
D→E | bC
E→ d | Ab

Q.6 **(a)** Consider the following grammar **06**

S→T00T
T→0T | 1T | Є

Derive the String "1000111" from the given grammar with Leftmost Derivation, Rightmost Derivation, Reduction, and Parse Tree.

(b) Design PDA to accept the language L={ w ϵ $(a+b)^*$ | $a^n . b^{2n}$ |n>=1}. Assume any string related to language **06**

(c) Design context-free grammar for **06**
Given Languages.
1. CFG for balanced parenthesis.
2. CFG for Palindrome string W □ $\{0+1\}^*$

OR

Q.6 **(a)** Consider following grammar **06**

S→aB | bA
A→a|aS|bAA
B→b|bS|aBB

Derive the String "aabbabba" from the given grammar with Leftmost Derivation, Rightmost Derivation, Reduction, and Parse Tree.

(b) Design PDA to accept the language L={ n_a (w) = n_b(w) | n >=1 }. The number of a's is equal to a number of b's in all the strings Assume any string related to language and prove it. **06**

(c) Simplified following CFG. **06**

S→ A | 0C1
A→B| 01 | 10
C→ ϵ | CD

****