Reg. No. : | | | | | | | | | | |

Question Paper Code: 45204

B.E. / B.Tech. DEGREE EXAMINATION, MAY 2022

Fifth Semester

Computer Science and Engineering

14UCS504 – THEORY OF COMPUTATION

(Regulation 2014)

Duration: Three hours Maximum: 100 Marks

Answer ALL Questions

PART A - (10 x 1 = 10 Marks)

1. Any NFA can be converted to a DFA

    (a) always (b) never
    (c) depending on the NFA (d) depending on the language of NFA

2. What is the minimum number of states in a DFA that recognizes the set of all binary strings which contains four consecutive 1's?

    (a) 6 (b) 5 (c) 4 (d) 3

3. The string 1101 does not belong to the set represented by

    (a) 110*(0+1) (b) 1(0+1)*101
    (c) (10)*(01)*(00+11) (d) ((11)*+01)*

4. The finite automata accept which of the following language.

    (a) context free language (b) regular language
    (c) context sensitive language (d) all the above

5. How many tuples are needed to represent an instantaneous description of a PDA?

    (a) 1 (b) 2 (c) 3 (d) 4

6. The language $L = \{0^m 1^m / m \geq 1\}$ is a

(a) regular language (b) context free language

(c) both (a) and (b) (d) none of these

7. The class of context free language is not closed under

(a) Concatenation (b) intersection

(c) Union (d) Repeated concatenation

8. Context free grammars are closed under

(a) union (b) kleene star (c) concatenation (d) all the above

9. What is the maximum number of codes is generated to encode a turing machine which consists of four transition function?

(a) 12 (b) 24 (c) 36 (d) 48

10. The diagonalization language L_d is

(a) recursive (b) not recursively enumerable

(c) recursively enumerable (d) both (a) and (c)

PART - B (5 x 2 = 10 Marks)

11. Differentiate DFA and NFA.

12. State the pumping lemma for regular languages.

13. Define the language generated by a PDA.

14. Design a turing machine for computing the function $f(x) = x + 1$.

15. Define the classes P and NP.

PART - C (5 x 16 = 80 Marks)

16. (a) (i) Prove that for every integer n≥0 the number $4^{2n+1} + 3^{n+2}$ is multiple of 13. (10)

(ii) Convert the given NFA to DFA. (6)

δ	0	1
→ q0	{ q0, q1}	q0
q1	q2	q1
q2	q3	q3
*q3	ϕ	q2

Or

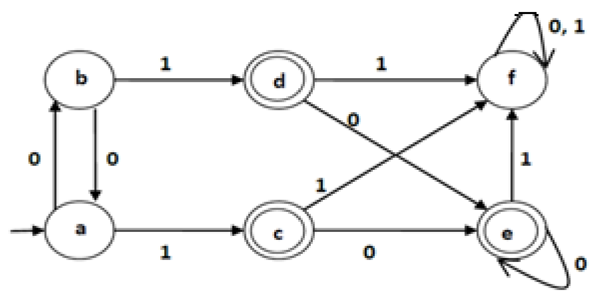
(b) (i) Consider the following Ɛ-NFA. Compute Ɛ-closure of each state and find its equivalent DFA. (10)

δ	Ɛ	a	b	c
→	ϕ	{p	{q	{r}
q	{p	{q	{r}	ϕ
*r	{q	{r}	ϕ	{p

(ii) Design a DFA which accepts odd number of 1's and any number 0's. (6)

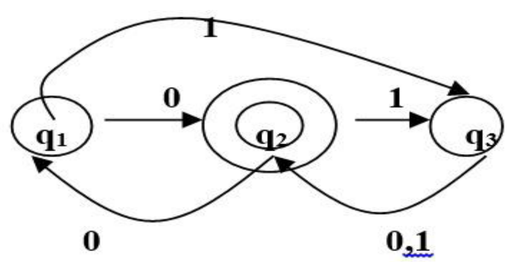
17\. (a) (i) Construct the Ɛ- NFA for the regular expression (1+0)*1(1+0). (6)

(ii) Find the minimized state DFA for the given DFA. (10)



Or

(b) Obtain the regular expression that denotes the language accepted by, using the recursive relation. (16)



18\. (a) (i) Let *S → aB/bA, A → aS/bAA/a, B → bS/aBB/b*. Show that *S ⇒ aaabbabbba* and construct a derivation tree whose yield is in *"aaabbabbba"*. (8)

(ii) Construct a PDA for the language $L = \left\{ \frac{a^n \; b^{2n}}{n \geq 1} \right\}$. (8)

Or

(b) (i) Convert the context free grammar $S \rightarrow aA,\ A \rightarrow aABC/bB/a,\ B \rightarrow b,\ C \rightarrow c$ into pushdown automata and process the string *"aaabc"*. (8)

(ii) Show that the following grammars are ambiguous.

$\{S \rightarrow aSbS/bSaS/\varepsilon\}$ and $\{S \rightarrow AB/aaB,\ A \rightarrow a/aA,\ B \rightarrow b\}$. (8)

19. (a) (i) Begin with grammar $S \text{ -> } 0A0/1B1/BB,\ A \text{ -> } C,\ B \text{ -> } S/A,\ C \text{ -> } S/\varepsilon$ and simplify using safe order

(1) eliminate ε production (2) eliminate unit production

(3) eliminate useless symbols (4) put the resultant grammar in CNF. (8)

(ii) Show that the language $L = \{a^i\ b^j\ c^i\ d^j \ /\ i \geq 1$ and $j \geq 1\}$ is not CFL. (8)

Or

(b) (i) Discuss the closure properties of CFL and prove any one of the property. (8)

(ii) Explain the programming techniques of Turing machine. (8)

20. (a) (i) State post correspondence problem. Let $\sum = \{a, b\}^*$. Let A and B be lists of three strings as given below

$A = \{b,\ bab^3,\ ba\}$ $B = \{b^3,\ ba,\ a)$. Does this instance of PCP have a solution? (6)

(ii) Prove that for two recursive language L_1 and L_2, their union and intersection is recursive. (10)

Or

(b) (i) Define universal language L_u. Prove that L_u is recursively enumerable. (8)

(ii) State halting problem. Show that it is undecidable. (8)