Register No: | | | | | | | | | |

Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam – 603 110

(An Autonomous Institution, Affiliated to Anna University, Chennai)

B.E. / B.Tech. End Semester Theory Examinations, Nov / Dec 2021

Fifth Semester

COMPUTER SCIENCE AND ENGINEERING

UCS1503 Theory of Computation

(Regulations 2018)

Time: **Three Hours** **Answer ALL Questions** Maximum:**100 Marks**

K1: Remembering K2: Understanding K3: Applying K4 :Analyzing K5: Evaluating

PART – A (10 × 2 = 20 Marks)

01.	K2	Compare deterministic and non-deterministic finite automata.	CO1
02.	K2	Write the regular expression for following language over the alphabet {0,1}: The set of all strings with atmost one pair of consecutive 0's.	CO1
03.	K3	Derive the string "aaabbabbba" by left most derivation using the productions given: P: S→ aB | bA , A→ a | aS | bAA B→ b | bS | aBB	CO2
04.	K3	Remove the useless symbols in this grammar: E→AB | a A→ab B→aB	CO2
05.	K1	List the different ways of language acceptances by a Pushdown Automata and define them.	CO3
06.	K1	State the pumping lemma for Context Free Language.	CO3
07.	K1	Write the formal definition for Turing Machine.	CO4
08.	K3	Design a Turing Machine to perform addition of two integers.	CO4
09.	K1	What is undecidability? Give two examples for undecidable problems.	CO5
10.	K2	Show that the complement of a recursive language is recursive.	CO5

PART – B (5 × 6 = 30 Marks)

(No Sub-divisions in Part-B)

11.	K2	Show that the set L={0^P | P is prime } is not regular.	CO1

12.	K3	What is ambiguous grammar? Check whether the given grammar is ambiguous. S→A1B A→0A | λ B→ 0B | 1B | λ	CO2
13.	K3	Construct a PDA for the Language L = $\{a^n b^m c^n \mid w \text{ is in } \{a,b,c\}^*\}$.	CO3
14.	K3	Design a Turing Machine for the language L = $\{0^n 1^n \mid n \geq 1\}$.	CO4
15.	K3	Let Σ = {0,1}. Let A and B be the lists of three strings each, defined as List A: w_i = 110, 0011, 0110 List B: x_i = 110110, 00, 110 Find the instance of PCP.	CO5

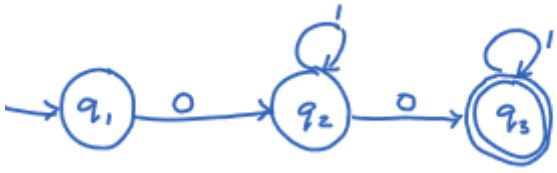
PART – C (5 × 10 = 50 Marks)

(No Sub-divisions in Part-C)

16.	K3	Construct a DFA equivalent of the NFA given below with **p** as initial state and **s** as final state.	CO1

states	0	1
→p	{q,s}	{q}
q	{r}	{q,r}
r	{s}	{p}
*s	-	{p}

OR

17.	K3	Find the regular expression for the given Finite Automata using $R_{ij}^{(k)}$ method:  	CO1
18.	K3	Write the procedure to convert CFG to CNF and apply it on the following CFG: S→ AAC A→ aAb | λ C → aC | λ	CO2

		OR	
19.	K3	Write the procedure to convert CFG to GNF and apply it on the following CFG: S→ AB A→ BS ∣ b B→ SA ∣ a	CO2
20.	K3	Construct a PDA for the Language L={wcw^R ∣ w is in {a,b}*} by empty stack. Verify whether the model accepts the string = abcba.	CO3
		OR	
21.	K3	Construct a Context Free Grammar G which accepts the PDA N(M), where M=({q0,q1},{a,b},{z0,z},δ,q0,z0,Φ) and where δ is given by a. δ(q0,b,z0)={(q0,zz0)} b. δ(q0, ε,z0)={(q0, ε)} c. δ(q0,b,z)={(q0,zz)} d. δ(q0,a,z)={(q1,z)} e. δ(q1,b,z)={(q1, ε)} f. δ(q1,a,z0)={(q0,z0)}	CO3
22.	K2	Explain the programming techniques for Turing machine construction.	CO4
		OR	
23.	K2	Explain how the Turing machine is used in computation with an example.	CO4
24.	K2	Prove the following: a. Halting problem is undecidable b. MPCP reduces to PCP	CO5
		OR	
25.	K2	Find whether the following languages are recursive or recursively enumerable and prove the same a. Union of two recursively enumerable languages. b. Diagnolization Language ($L_{d)}$ c. Universal Language ($L_{u)}$	CO5

CO1: Construct automata, regular expression for any given pattern (K3)
CO2: Identify the grammar type and the need of formal languages, and grammars (K3)
CO3: Model the pushdown automata for any CFL (K3)
CO4: Construct Turing Machines for the given languages (K3)
CO5: Explain the Decidability or Undecidability of various problems (K2)