Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam – 603 110

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

Department of Information Technology

Continuous Assessment Test – I

Regulations – R2021

Degree B.E. / B. Tech.	B.Tech	**Branch**	IT
Semester	IV	**Date of CAT**	25-04-2023
Subject Code & Name	UIT2404 & Automata Theory and Compiler Design		
Time: 90 Minutes	**Answer All Questions**		**Maximum: 50 Marks**

(K1: Remembering, K2: Understanding, K3: Applying, K4: Analyzing, K5: Evaluating)

CO1:	Construct automata, regular expression for any pattern.
CO2:	Write Context free grammar for any construct.
CO3:	Build the different Phases of compiler and apply the various optimization techniques.
CO4:	Design Turing machine for a given language
CO5:	Explain decidability, semi-decidability, and undecidability

Part – A (4 × 2 = 8 Marks)

		KL	CO	PI
1.	Design DFA for the language L = {w ∈ Σ* | w ≠ ε and the first and last character of w are the same} over Σ = {a, b}.	K3	CO1	1.1.1 1.4.1 2.1.3
2.	Let the alphabet Σ_A = {a, b} and consider the following DFA A: A = (Q_A {0, 1, 2, 3}, Σ_A, δ_A, q_0 = 0, F_A = {1, 2}) δ_A = {((0, a), 1), ((0, b), 2), ((1, a), 0), ((1, b), 3), ((2, a), 3), ((2, b), 0), ((3, a), 2), ((3, b), 1)}. Calculate $\hat{\delta}_A(0, abba)$.	K2	CO1	1.3.1 1.4.1 2.1.3
3.	Is the following ambiguous? Justify. S → SAS | 0 A → ASA | 1	K3	CO2	1.3.1 1.4.1 2.1.3
4.	Consider the following context-free grammar G. Give two strings in L(G) and give two strings not in L(G). R → XRX | S S → aT b | bT a T → XTX | X | ε X → a | b	K2	CO2	1.1.1 1.3.1 1.4.1 2.1.3

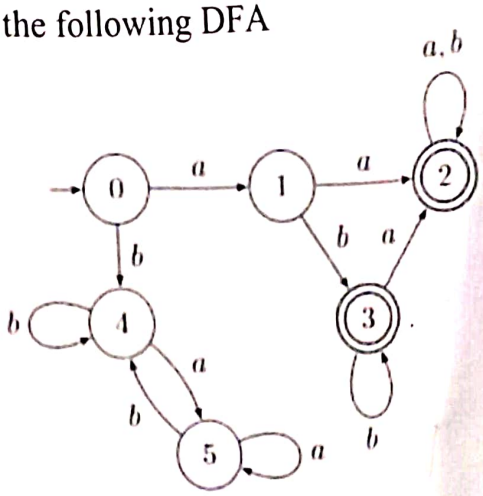
Part – B (3×6 = 18 Marks)

		KL	CO	PI
5.	In certain programming languages, comments appear between delimiters such as /# and #/. Let X be the language of all valid delimited comment strings. A member of X must begin with /# and end with #/ but have no intervening #/. Assume that the alphabet for X is Σ = {a, b, /, #}. a. Give a DFA that recognizes X. b. Give a regular expression that generates X. (3+3)	K3	CO1	1.1.1 1.4.1 2.1.3 13.1.1
6.	Let A be the language consisting of all strings over {0,1} containing a 1 in the third position from the end (e.g., 000100 is in A but 0011 is not). Design NFA for the language A and construct the equivalent DFA. (3+3)	K3	CO1	1.1.1 1.4.1 2.1.3 13.1.1
7.	Let G be a CFG with the following productions: S→AA A→AAA|a|bA|Ab Which strings of L(G) can be produced by derivations of four or fewer steps? Give at least four distinct derivations for the string babbab. (3+3)	K3	CO2	1.1.1 1.4.1 2.1.3 13.1.1

Part – C (2 × 12 = 24 Marks)

		KL	CO	PI
8.	Write regular expressions for the following languages over the alphabet Σ = {a, b} and systematically construct an NFA for each of the regular expression a) All strings that do not end with aa. b) All strings that contain an even number of b's. c) All strings which do not contain the substring **ba**. (4+4+4)	K3	CO1	1.1.1 1.4.1 2.1.3 13.1.1
	(Or)			
9.	a) Convert the following FA into regular expression $\rightarrow q_1$: q_1, q_2 $*q_2$: q_2, q_3 q_3 : q_3, q_1	K3	CO1	1.1.1 1.4.1 2.1.3 13.1.1

b) Minimize the following DFA



(6+6)

10.	Give context-free grammars generating the following languages. a. {w | w contains at least three 1s over Σ {0,1}} b. {w | the length of w is odd and its middle symbol is a 0 over Σ {0,1}} c. The set of strings over the alphabet {a, b} with more a's than b's d. A language of properly nested strings of parentheses, square brackets ([,]) and braces ({, }). **(3+3+3+3)**	K3	CO2	1.1.1 1.4.1 2.1.3 13.1.1
	(Or)			
11.	Consider the following grammar, with the terminals true, false, &&, and ||: T → true| false| T && T | T || T (a) Demonstrate that the grammar is ambiguous by showing at least two parse trees for the string true && false || true. (b) Reconstruct the grammar so that it is unambiguous, the && operator is right-associative, the || operator is left-associative, and && has higher precedence than ||. (c) Draw the parse tree for true && false || true in your refactored grammar. **(4+4+4)**	K3	CO2	1.1.1 1.4.1 2.1.3 13.1.1