U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

April 2025 Semester End Make-Up Examinations

Programme: B.E. | **Semester: III**

Branch: Artificial Intelligence and Machine Learning | **Duration: 3 hrs.**

Course Code: 23AM3AETFC | **Max Marks: 100**

Course: Theoretical Foundations of Computations

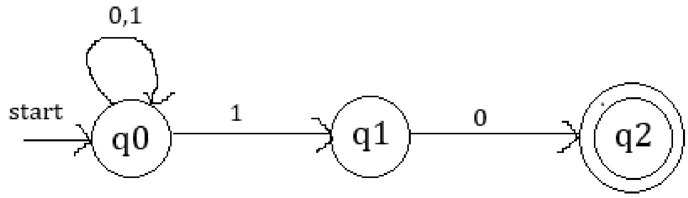
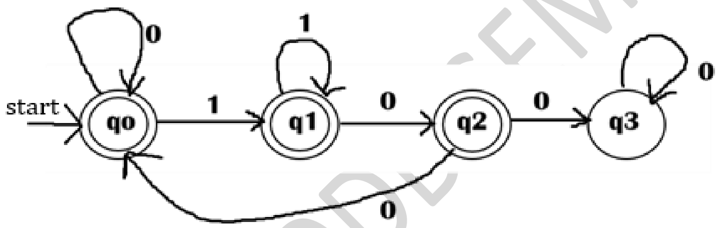
Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

Important Note: Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages. Revealing of identification, appeal to evaluator will be treated as malpractice.

		UNIT - I	*CO*	*PO*	**Marks**
1	a)	Illustrate the concept of Deterministic Finite Automata (DFA) along with its definition.	*CO1*	*PO2*	**05**
	b)	Design a Deterministic Finite Automata (DFA) that accepts the language over the alphabet Σ= {0,1} i. L= {w ∈ w is of even length and begins with 10}. ii. L= {w ∈ w is a set of all strings not consisting 1010 as a substring}.	*CO1*	*PO3*	**08**
	c)	Write transition diagram for the given transition table and convert the Non-Deterministic Finite Automata (NFA) to Deterministic Finite Automata (DFA) using the Lazy Evaluation Method.	*CO1*	*PO2*	**07**

	δ	a	b
→	q0	q0, q1	q0, q3
	q1	∅	q2
*	q2	∅	∅
	q3	q4	∅
*	q4	∅	∅

		OR			
2	a)	Define Alphabet, Language, String, Power of an alphabet, and Cardinality with an example.	*CO1*	*PO1*	**05**
	b)	Construct a Deterministic Finite Automata (DFA) that accepts the language L= {w is a binary number divisible by 5} over the alphabet Σ= {0,1}	*CO1*	*PO3*	**08**
	c)	i. Identify the language accepted by the given Non-Deterministic Finite Automata (NFA).	*CO1*	*PO2*	**07**

<table>
<tr><td></td><td></td><td></td><td>ii. Convert the Non- Deterministic Finite Automata (NFA) to its equivalent Deterministic Finite Automata (DFA) using the Subset Construction method.

</td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td>UNIT - II</td><td></td><td></td><td></td></tr>
<tr><td></td><td>3</td><td>a)</td><td>State Pumping Lemma for regular languages and prove that L = {$a^n b^n$: n is an integer} is not regular.</td><td>CO1</td><td>PO1</td><td>06</td></tr>
<tr><td></td><td></td><td>b)</td><td>i. Write Regular Expression to accept strings of a’s and b’s starting with ab.
ii. Construct ε-NFA for the Regular Expression obtained in (i).</td><td>CO1</td><td>PO2</td><td>06</td></tr>
<tr><td></td><td></td><td>c)</td><td>Obtain Regular Expression for the DFA using State Elimination Method.

</td><td>CO1</td><td>PO2</td><td>08</td></tr>
<tr><td></td><td></td><td></td><td>OR</td><td></td><td></td><td></td></tr>
<tr><td></td><td>4</td><td>a)</td><td>Design Regular Expressions for the languages over the alphabet Σ= {0,1}
i. The set of all strings that begin with 110
ii. The set of all strings that do not end with 01.
iii. The set of all strings whose length is divisible by 3.</td><td>CO1</td><td>PO1</td><td>06</td></tr>
<tr><td></td><td></td><td>b)</td><td>Apply Kleen’s Theorem to derive the Regular Expression of the DFA.
<table><tr><th>δ</th><th>0</th><th>1</th></tr><tr><td>→1</td><td>2</td><td>1</td></tr><tr><td>2</td><td>3</td><td>1</td></tr><tr><td>*3</td><td>3</td><td>2</td></tr></table></td><td>CO1</td><td>PO1</td><td>06</td></tr>
<tr><td></td><td></td><td>c)</td><td>Minimize the Deterministic Finite Automata (DFA) using the Myhill Nerode Method.</td><td>CO1</td><td>PO2</td><td>08</td></tr>
</table>

δ	0	1
-> A	B	E
B	C	F
*C	D	H
D	E	H
E	F	I
*F	G	B
G	H	B
H	I	C
*I	A	E

			UNIT - III			
	5	a)	i. Define ambiguous and unambiguous grammar. ii. Consider the Grammar S→ S*S | S+S | S-S | b | c. Show that the grammar is ambiguous for the string w=b+c*b-c.	*CO2*	*PO1*	**06**
		b)	Obtain Context Free grammar to generate strings belonging to the given languages i. L= {w : na(w)>=1, w ∈ {a,b}*} ii. L={w: na(w)<=3, w ∈ {a,b}*} iii. L={w=0^n 1^m | n>=0,m>=0 }	*CO2*	*PO2*	**06**
		c)	Write Left most derivation(LMD), Right most derivation(RMD), and parse tree for the string w=+*-xyxy using the grammar E-> +EE | *EE | -EE| x|y	*CO2*	*PO1*	**08**
			OR			
	6	a)	Convert the given grammar to Chomsky Normal Form (CNF) E->T*E / T-E / T T-> T+F /F F-> a / (E)	*CO2*	*PO2*	**10**
		b)	i. Is the language L = {$x^ny^nz^n$ | n ≥ 1} context free? Justify. ii. Elaborate on the closure properties of Context-Free Language (CFL).	*CO2*	*PO2*	**10**
			UNIT - IV			
	7	a)	i. Construct Pushdown Automata (PDA) to accept the language. L= {a^nb^{2n}; n>=1}. ii. Draw the transition diagram for the constructed PDA. iii. Provide the instantaneous description for the string aabbbb.	*CO2*	*PO3*	**10**
		b)	i. Convert the given Context Free Grammar (CFG) to Pushdown Automata (PDA) S→aABC A→aB |a B→bA|b C→a ii. Provide the moves for the string aabba for the PDA in (i).	*CO2*	*PO2*	**10**
			OR			

	8	a)	i. Design a Pushdown Automata (PDA) for the language $L=\{wcw^R, w \in \{a,b\}^*\}$.Is it deterministic? Justify. ii. Show the moves for the string abaaba.	*CO2*	*PO2*	**10**
		b)	Convert the given Pushdown Automata (PDA) M=({q}, {0,1},{Z,A,B}, δ, q, Z) to its equivalent Context Free Grammar(CFG). The transition functions are as follows: δ(q,0,B)= (q, ε) δ(q,1,A)= (q, ε) δ(q,1,Z)= (q, ε) δ(q,0,Z)= (q, AZ) δ(q,1,Z)= (q, BZ) δ(q,0,A)= (q, AA) δ(q,1,B)= (q, BB)	*CO2*	*PO2*	**10**
			UNIT - V			
	9	a)	Illustrate the Turing machine (TM) using its 7-tuple representation.	*CO3*	*PO2*	**05**
		b)	Differentiate between Turing machine (TM) as an acceptor and TM as a transducer.	*CO3*	*PO1*	**05**
		c)	Provide a formal description of the Turing Machine (TM) that accepts the language $L= \{w=0^n1^n2^n \mid n>=0\}$. Prove the acceptance of the string 001122 using an instantaneous description.	*CO3*	*PO2*	**10**
			OR			
	10	a)	Design a Turing Machine (TM) to accept the language containing strings of 0's and 1's with a substring 001.	*CO3*	*PO3*	**07**
		b)	Consider 2 integer numbers x and y in unary representation of 1's only. Design a Turing Machine in the role of transducers to perform the addition operation x+y.	*CO3*	*PO1*	**07**
		c)	Is there a solution to the following Post Correspondence Problem (PCP)? <table><tr><th>i</th><th>A</th><th>B</th></tr><tr><td>1</td><td>100</td><td>1</td></tr><tr><td>2</td><td>0</td><td>100</td></tr><tr><td>3</td><td>1</td><td>00</td></tr></table>	*CO3*	*PO1*	**06**

******