TEACHING PLAN: Formal Language and Automata Theory

FACULTY NAME: Ms. SHRUTI GUPTA

<table>
<tr><td colspan="2">SCHOOL OF ENGINEERING AND TECHNOLOGY</td><td>ACADEMIC SESSION: 2022-2023</td><td>FOR STUDENTS BATCH: 2020-2024</td></tr>
<tr><td>1.</td><td>Course Code</td><td colspan="2">PEC-CSE-305</td></tr>
<tr><td>2.</td><td>Course Title</td><td colspan="2">THEORY OF COMPUTATION</td></tr>
<tr><td>3.</td><td>Credits</td><td colspan="2">4</td></tr>
<tr><td>4.</td><td>Learning Hours</td><td colspan="2"><table><tr><td>Contact Hours</td><td>3</td></tr><tr><td>Practical Teaching</td><td>0</td></tr><tr><td>Project, Tutorial and Assessment</td><td>1</td></tr><tr><td>Total Hours</td><td>4</td></tr></table></td></tr>
<tr><td>5.</td><td>Course Objectives</td><td colspan="2">Understand basic properties of formal languages and formal grammars. 2. Understand basic properties of deterministic and nondeterministic finite automata 3. Understand the relation between types of languages and types of finite automata 4. Understanding the Context free languages and grammers, and also Normalising CFG. 5. Understanding the minimization of deterministic and nondeterministic finite automata. 6. Understand basic properties of Turing machines and computing with Turing machines. 7. Understand the concept of Pushdown automata and its application. 8. Know the concepts of tractability and decidability, the concepts of NP-completeness and NP-hard problem. 9. Understand the challenges for Theoretical Computer Science and its contribution to other sciences.</td></tr>
<tr><td>6.</td><td>Course Outcomes</td><td colspan="2">1)Knowledge Acquire a full understanding and mentality of Automata Theory as the basis of all computer science languages design - Have a clear understanding of the Automata theory concepts such as RE's, DFA's, NFA's, Turing machines, Grammar, halting problem, computability and complexity. 2) Cognitive skills (thinking and analysis). - Be able to design FAs, NFAs, Grammars, languages modelling, small compilers basics - Be able to design sample automata - Be able to minimize FA's and Grammars of Context Free Languages. 3) Professional Skill - Perceive the power and</td></tr>
</table>

		limitation of a computer - Solve the problems using formal language 4) Attitude- Develop a view on the importance of computational theory.				
7.	**Outline syllabus**					
7.01	**Paper Code:** **PEC-CSE 305**		**Unit**	**Introduction**	**Reference Number**	**Teaching Methodology**
			I	**Sets, operations, relations, strings, transitive closure, count ability and diagonalisation, induction and proof methodspigeon-hole principle and simple applications – concept of language – grammars and production rules – Chomsky hierarchy.**		**Whiteboard, PPT slides, Tutorials, Demonstration**
			II	**Finite State machine, regular languages, deterministic finite automata, conversion to deterministic automata, Eclosures – regular expressions, finite automata, and minimization of automata, Moore and Mealy machine and their equivalence.**		**Whiteboard, PPT slides, Tutorials, Demonstration**
			III	**Pumping lemma for regular sets- closure properties of regular sets- decision properties for regular sets, equivalence between regular language and regular grammar. Context – free languages – parse trees and ambiguity, reduction of CFGS ,Chomsky and Griebach normal forms.**		**Whiteboard, PPT slides, Tutorials, Demonstration**

			IV	**Non Determinism – acceptance by two methods and their equivalence, conversion of PDA to CFG CFLs and PDAsclosure and decision properties of CFLs.**		**Whiteboard, PPT slides, Tutorials, Demonstration**	
			V	**Variants – recursively enumerable (r.e.) set – recursive sets ,TM as computer of function – decidability and solvability – Halting Problem – reductions – Post correspondence Problem (PCP) and un solvability of ambiguity problem of CFGs, Church‟s hypothesis, Introduction to recursive function theory – primitive recursive and partial recursive functions.**		**Whiteboard, PPT slides, Tutorials, Demonstration**	
8	**Course Evaluation**						
8.10	**CA: 20%**						
8.1	**Attendance**	**10%**					
8.12	**Homework**	**10%**					
8.13	**Quizzes**						
8.14	**Projects**						
8.15	**Presentation**						
8.16	**Any other**						
8.2	**MTE(IA)**	**20%**					

8.3	**End Term Examination**	**60%**
9	**Text book and References**	**Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley.** **Introduction to languages and the Theory of Computation ,John C Martin, TMH**
9.1	**Text books**	
9.2	**References**	
9.3	**Video References**	

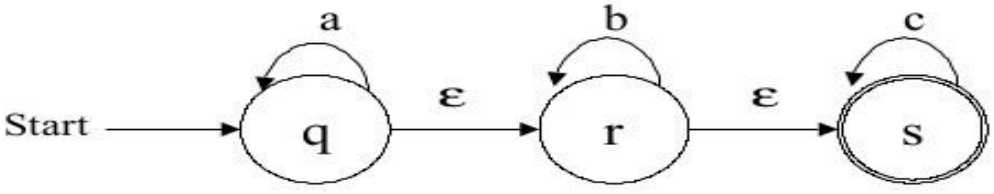
Mapping of Outcomes vs Topics

Course Outcome	Program Outcome												PSO			
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3	4
CO1	3	3	2	3	1	2	1		2	2	1	2	3	3	2	2
CO2	3	3	3	3	2	2	1		1	1	1	1	3	3	2	2
CO3	3	3	3	3	3	2	2		2	2	2	2	3	3	2	2
CO4	3	3	3	3	3	2	2		2	1	1	1	3	2	2	1
CO5	3	2	1	1	3	3	2		1	1	1	2	3	2	3	2

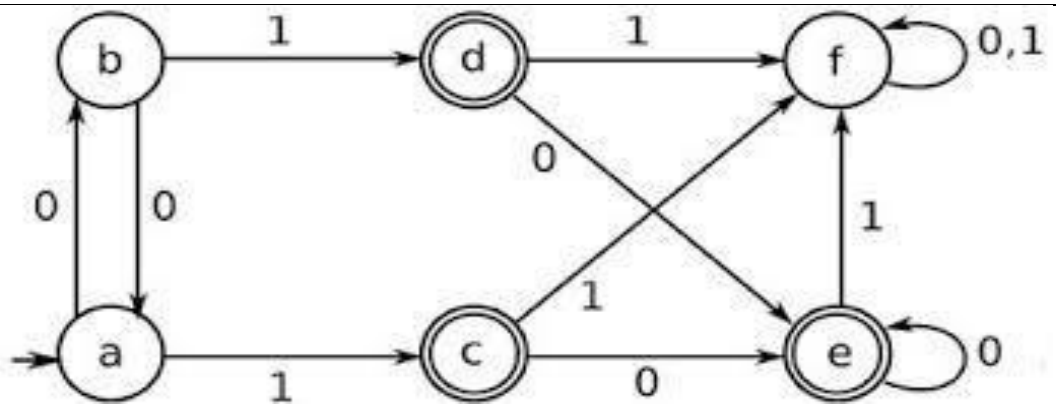
QUESTION BANK

1. **Define NFA. What are the differences between DFA & NFA?**

2. **Convert the following NFA with ε moves to DFA without ε moves.**



3. **Minimize the following finite automata.**



4. Convert the following Mealy machine into its equivalent Moore machine.

Present State	I/P=0		I/P=1	
	Next State	O/P	Next State	O/P
→ A	C	0	B	0
B	A	1	D	0
C	B	1	A	1
D	D	1	C	0

5. A) Write about relations on sets.

B) Define Grammar? What are the tuples?

C) Define Finite Automaton.

D) Show that (0*1*)* = (0+1)*.

E) Define Mealy machine and Moore machine.

6. a) Discuss Chomsky's Hierarchy of formal languages.

b) Explain briefly about DFA and NFA?

7. a) Define Moore machine? Construct Mealy machine corresponding to Moore machine?

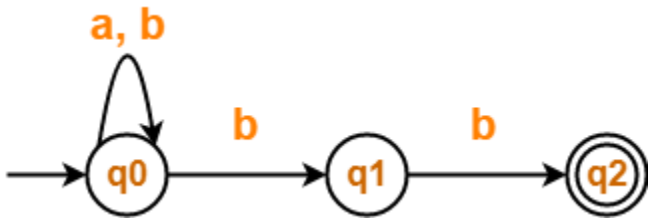
States (Q)	Next States		Output
	I/P=0	I/P=1	
→q1	q1	q2	0
q2	q1	q3	0
q3	q1	q3	1

8. Obtain DFAs to accept strings of a's and b's having exactly one a.

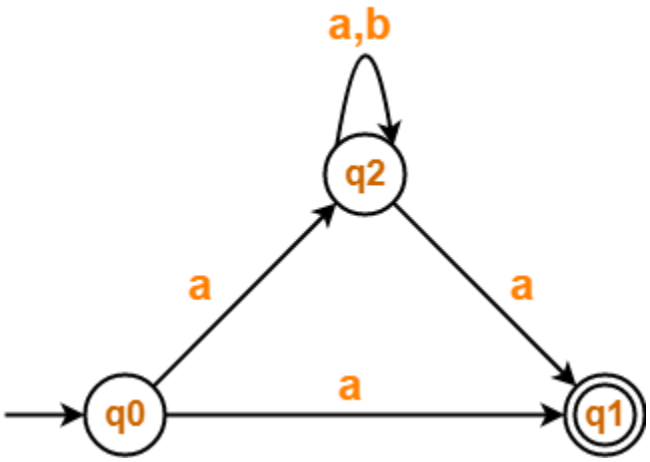
9. Define DFA, NFA & Language?

10. Draw a DFA to accept string of 0's and 1's ending with the string 011.

11. Convert the following Non-Deterministic Finite Automata (NFA) to Deterministic Finite Automata (DFA)-



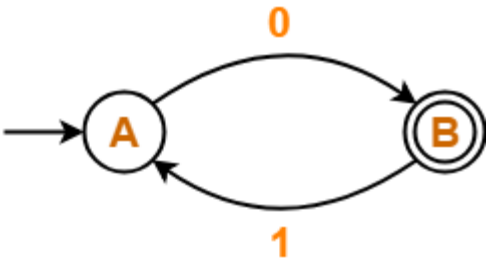
12. Convert the following Non-Deterministic Finite Automata (NFA) to Deterministic Finite Automata (DFA)-



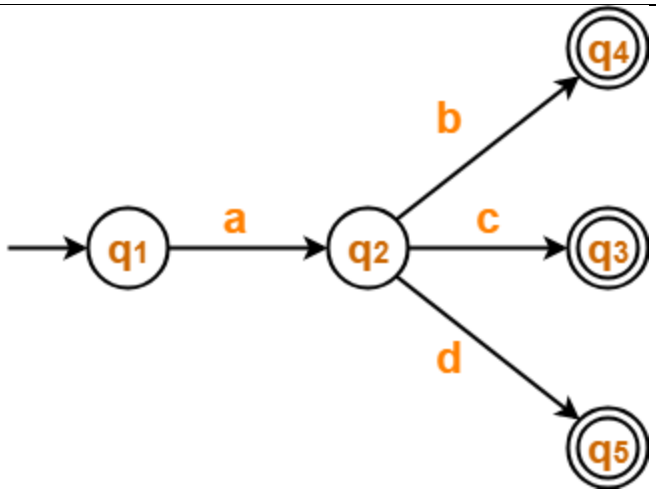
13. **Draw a DFA for the language accepting strings starting with '101' over input alphabets ∑ = {0, 1}**

14. Define finite automata? Explain detail about the tuples.

15. Find regular expression for the following DFA-



16. Find regular expression for the following DFA-



17. **List out the identities of Regular expression.**

18. **Construct an equivalent FA for the given regular expression (0+1)*(00+11)(0+1)***

19. **Draw a DFA for the language accepting strings starting with '101' over input alphabets**

$\sum$ = {0, 1}.

20. **Draw a DFA that accepts a language L over input alphabets $\sum$ = {0, 1} such that L is the set of all strings starting with '00'.**