22CS1T3: FORMAL LANGUAGES AND AUTOMATA THEORY

Course Name	Formal Languages and Automata Theory		L	T	P	C	CIA	SEE	TM
Course Code	22CS1T3		4	0	0	4	30	70	100
Year of Introduction: 2018	Year of Offering: 2022	Year of Revision: 2022			Percentage of Revision: 20				
L-Lecture, T-Tutorial, **P**-Practical, **C**-Credits, **CIA**-Internal Marks, **SEE**-External Marks, **TM**-Total Marks									

Course Description and Purpose:
Formal Languages and Automata Theory deals with the concepts of *Automata*, *Formal Languages*, *Grammar*, *Algorithms*, *Computability*, *Decidability and Complexity*. It also helps to develop methods by which computer scientists can describe and analyze the dynamic behavior of *Discrete Systems*, in which signals are sampled periodically.

Course Objectives:
- ✓ To understand basic properties of *Deterministic* and *Nondeterministic Finite Automat.*
- ✓ To understand *Context Free Languages* and *Grammers*, and also *Normalising CFG*.
- ✓ To understand the concept of *Pushdown Automata Turing Machine* and its application.
- ✓ To understand Basic Structure of *Compiler Design*.
- ✓ To understand the concept of *Lex* and *Syntax Analysis.*

Course Learning Outcomes:

At the end of this course the students should be able to:

CO1: Understand basic properties of *Deterministic* and *Nondeterministic Finite Automat.*
CO2: Understand the *Context Free Languages* and *Grammers*, and also *Normalising CFG*.
CO3: Understand the concept of *Pushdown Automata Turing Machine* and its application.
CO4: Understand Basic Structure of *Compiler Design*.
CO5: Understand the concept of *Lex* and *Syntax Analysis.*

UNIT I (12 Hours)

Fundamentals: Strings, Alphabet, Language, Operations, Finite Automaton Model, Acceptance of Strings and Languages, Transition Table and Transition Diagrams.
Finite Automata: Deterministic Finite Automaton, Non deterministic Finite Automaton and NFA with ε Transitions, Significance, Equivalence between NFA with and without ε Transitions, NFA to DFA Conversion,Minimization of FSM, Equivalence between two FSMs, Finite Automata with Output-Moore and Mealy Machines.

UNIT II (12 Hours)

Regular Languages: Regular Sets, Regular Expressions, Identity Rules, Construction of Finite Automata(DFA) for a given Regular Expressions and its inter conversion using State Elimination and Ardens Theorem, Pumping Lemma of Regular Sets, Closure Properties of Regular Sets (Proofs not required).

UNIT III (12 Hours)

Context free grammar: Introduction, Derivation Trees, Ambiguity in Context Free Grammars. Minimization of Context Free Grammars. Chomsky Normal Form, Greibach Normal Form.
Push down Automata: Definition, Model, Design of PDA. The Language of PDA- Acceptance by Final State, Acceptance by Empty Stack ,Equivalence of CFL and PDA -Conversion of CFL to PDA and PDA to CFL

Turing Machine: Definition, Turing Machine Model, Types of Turing machine (problems not required) ,Types of Turing machine ,Recursively Enumerable Languages and Recursive Languages Chomsky Hierarchy of Languages and Post correspondence problem

UNIT IV (12 Hours)

Compiler: Introduction, Structure of a compiler, Design issues of compiler, Phases of Compiler, Lexical Analysis, Role of Lexical Analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens

UNIT V (12 Hours)

Lex (Lexical-Analyzer Generator): Uses of Lex, Structure of Lex Programs, Conflict Resolution in Lex , The Lookahead Operator.

Syntax Analysis: Top Down Parsing, Recursive-Descent Parsing, FIRST and FOLLOW, LL(l) Grammar, Nonrecursive Predictive Parsing, Error Recovery in Predictive Parsing.

Bottom-Up Parsing- Reductions, Handle Pruning, Shift-Reduce Parsing ,Conflicts During Shift-Reduce Parsing

Reference Text Books:

1. Hopcroft. H.E. and Ullman, Introduction to Automata Theory Languages and Computation, J. D. Pearson Education, January 2008.
2. Compilers-Principles, Techniques and Tools, Jeffery D.Ullman 2nd Edition, Pearson Education, January 2013.
3. John C Martin, Introduction to Languages and the Theory of Computation, Tata McGraw-Hill, 2003.

KRISHNA UNIVERSITY, MACHILIPATNAM, A.P., INDIA.
M.Sc.,(COMPUTER SCIENCE) DEGREE EXAMINATIONS
FIRST SEMESTER

FORMAL LANGUAGES AND AUTOMATA THEORY
SYLLABUS W.E.F 2022-2022 (R22)

Time 3 Hours **Max.Marks: 70**

SECTION-A
Answer any five questions. 5 × 4 = 20 Marks

1. (a) What is NFA with example?(CO1,L1)
(or)
(b) Define Mealy Machine with example (CO1,L1)

2. (a) Define regular set .What are the closure properties of regular sets? (CO2,L1)
(or)
(b) Define Expression. What are the different identity rules used in regular expression. (CO2,L1)

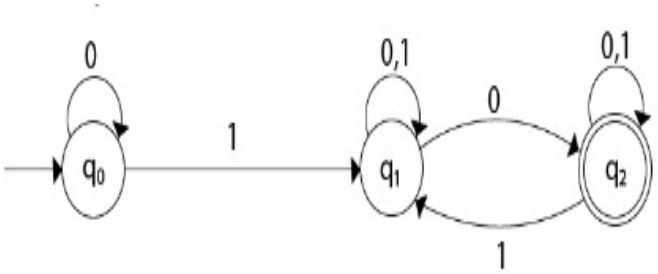
3. (a) Explain Ambiguity in context free grammars with example.(CO3,L2)
(or)
(b) Explain LMD and RMD with example(CO3,L2)

4. (a) Define input buffering with example. (CO4,L1)
(or)
(b) What is the role of Lexical Analyzer. (CO4,L1)

5. (a) Explain Conflict Resolution in Lex. (CO5,L2)
(or)
(b) Explain error recovery in predictive parsing. (CO5,L2)

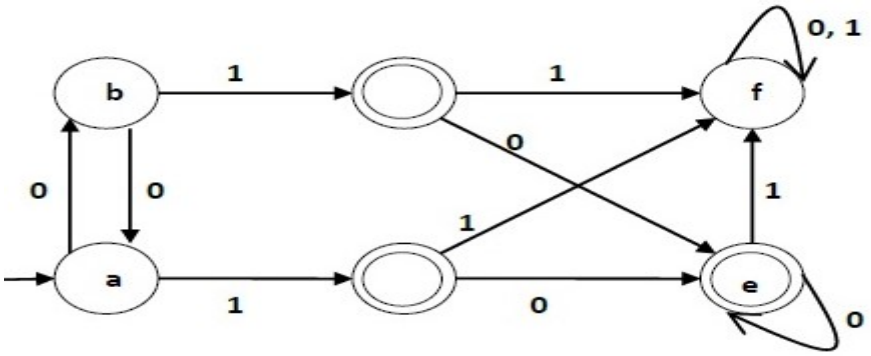
SECTION-B
Answer all questions. 5 × 10 = 50 Marks

6. (a) Construct the given NFA to DFA. (CO1,L3) 10 Marks

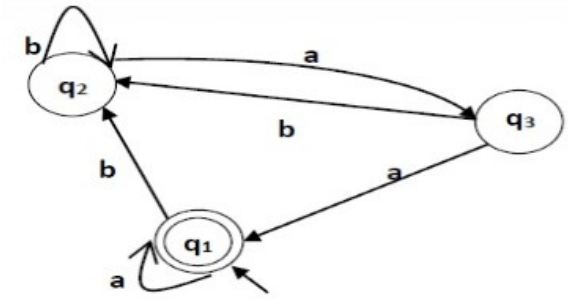


(or)

(b) Construct the given DFA into minimized DFA (CO1,L3)



7. (a) Construct a regular expression corresponding to the automata given below (CO2,L3)



(or)

(b) Solve the given Language $L= \{0^n 1^n \mid n \geq 1\}$ is not a regular language using pumping lemma (CO2,L3)

8. (a) Translate the given grammar to CNF (CO3,L2)

S → aAD
A → aB / bAB
B → b
D → d

(or)

(b) Explain Chomsky Hierarchy of Languages and Post correspondence problem with example.(CO3,L2)

9. (a) what are the design issues of compiler? (CO4,L1)

(or)

(b) What are the different phases used in Complier Design with diagram 10 Marks (CO4,L1)

10. (a) Define Lex. explain structure of Lex program and its uses (CO5,BTL2)

(or)

(b) Explain top down and bottom up parsing with example(CO5,L2)