20MA211E					Formal Languages and Automata Theory					
Teaching Scheme					Examination Scheme					
L	T	P	C	Hrs. /Week	Theory			Practical		Total Marks
					MS	ES	IA	LW	LE/Viva	
3	0	0	3	3	25	50	25	--	--	100

COURSE OBJECTIVES

- To identify the hierarchy of formal languages, grammars and machines.
- To understand equivalence of languages accepted by Push Down Automata and languages generated by context free grammars.
- To design finite automata and to accept a set of strings of a language.
- To design context free grammars to generate strings from a context free language.
- To develop a formal notation for strings, languages and machines.

UNIT 1 REGULAR LANGUAGES AND ITS APPLICATIONS **12 Hrs.**

Alphabets, Language, Operations on Languages. Regular Expression, Regular Languages. Finite State Machine, Deterministic and Non-deterministic Finite State machines, Regular Grammar, Finite State Machine with output - Moore machine and Melay Machine, Conversion of Moore machine to Melay Machine and Vice-Versa. Conversion of DFA to Regular Expression, Pumping Lemma, Properties and Limitations of Finite state machine, Decision properties of Regular Languages, Application of Finite Automata.

UNIT 2 CONTEXT FREE GRAMMAR (CFGs) **10 Hrs.**

Context Free Grammar, Derivation tree and Ambiguity, Application of Context free Grammars, Chomsky and Greibach Normal form, Properties of Context Free Grammar, CKY Algorithm, Decidable properties of Context free Grammar, Pumping Lemma for Context free grammar.

UNIT 3 PUSH DOWN AUTOMATA (PDAs) **10 Hrs.**

Definition of Push down automation, The languages of a PDA, Equivalence of PDAs and CFGs, Stack Machine, Design of Deterministic and Non -deterministic Push-down stack.

UNIT 4 TURING MACHINE **08 Hrs.**

Turing machine, Definition and design of Turing Machine, Church-Turing Thesis, Variations of Turing Machines, Universal Turing Machine, Post Machine, Chomsky Hierarchy, Post correspondence problem.

40 Hrs.

COURSE OUTCOMES

On completion of the course, student will be able to

CO1 – Understand the equivalence between Context-Free Grammars and Pushdown automata.
CO2 – Explain the theory of finite automata, as the first step towards learning advanced topics, such as compiler design.
CO3 – Design context free grammars to generate strings of context free language.
CO4 – Design finite automata to accept a set of strings of a language.
CO5 – Develop an understanding of computation through Turing Machines.
CO6 – Develop a clear understanding of the Chomsky hierarchy for language classes.

TEXT/REFERENCE BOOKS

1. E. J. Hopcroft, D. J. Ullman and R. Motwani, Introduction to Automata Theory, Languages and Computation, 3rd ed., Pearson Education, 2007.
2. C.J. Martin, Introduction to Languages and the Theory of Computation, 4th ed., McGraw-Hill Higher Education, 2011.
3. R.H. Lewis and H.C. Papadimitriou, Elements of the Theory of Computation, 2nd ed., Prentice Hall, 1998.
4. A. I. D. Cohen, Introduction to Computer Theory, 2nd ed., Wiley, 1997.
5. M. Sipser, Introduction to the Theory of Computation, 3rd ed., Cengage Learning, 2013.