SWARNANDHRA
COLLEGE OF ENGINEERING & TECHNOLOGY
(AUTONOMOUS)
Accredited by National Board of Accreditation, AICTE, New Delhi,Accredited by NAAC with "A" Grade – 3.32 CGPA,Recognized under 2(f) & 12(B) of UGC Act 1956, Approved by AICTE, New Delhi,Permanent Affiliation to JNTUK, Kakinada
Seetharampuram, W.G.DT., Narsapur-534280, (Andhra Pradesh)

DEPARTMENT OF INFORMATION TECHNOLOGY
TEACHING PLAN

Course Code	Course Title	Semester/ Regulation	Branch	Contact Periods /Week	Academic Year	Date of commencement of Semester
23IT5T02	Automata Theory and Compiler Design	V/(R23)	IT	6	2025-2026	09/06/202

COURSE OUTCOMES

1	Analyze and design DFAs, NFAs, proving their equivalence, and applying regular expression
2	Understand CFGs and design PDAs, relating them to context-free languages
3	Perform the syntax analysis by using parsing techniques.
4	Implement parsers (LL/LR) and apply semantic rules and error detection/correction technique
5	Apply various techniques to improve program efficiency (space/time complexity).

UNIT	OutComes / Bloom's Level	Topics No.	Topics/ Activity	Text Book/ Reference	Contact Hour	Delivery Method
I	CO – 1		**UNIT – I: Regular Expressions, Languages and Finite Automata**			Chalk & Board Power poir presentatior Assignmen Test
		1.1	Formal Languages and the Chomsky Hierarchy	T1,T2	1	
		1.2	Regular Expressions and Regular Languages	T1,T2	1	
		1.3	Algebraic Laws for Regular Expressions	T1,T2	1	
		1.4	Applications of Regular Expressions	T1,T2	1	
		1.5	Abstract model of Finite Automaton	T1,R1	1	
		1.6	Transition Tables and Transition Graphs	T1,R1	1	
		1.7	Deterministic Finite Automata (DFA)	T1,R1	1	
		1.8	Nondeterministic Finite Automata (NFA)	T1,R1	1	
		1.9	Converting NFA to DFA	T1,R1	1	
		1.10	Finite Automata with ϵ transitions (NFA-ϵ)	T1,R1	1	
		1.11	Converting NFA-ϵ to NFA/DFA	T1,T2	1	

		1.12	Minimization of Finite Automata	T1,T2	1	
		1.13	Equivalence of FA and Regular Expressions	T1,T2	1	
Content beyond syllabus		1.14	Bootstrapping	R2	1	
				Total	**14**	
II	CO – 2		**UNIT-II: Context Free Grammars and Push Down Automata**			Chalk & Board; Power poir presentatior; Assignmen; Test
		2.1	Context Free Grammars(CFG)	T1,R2	1	
		2.2	Context Free Languages(CFL)	T1,R2	1	
		2.3	Design of CFGs	T1,R2	1	
		2.4	Leftmost and Rightmost Derivations	T1,R2	1	
		2.5	Parse Trees	T1,R2	1	
		2.6	Applications of CFGs	T1,R2	1	
		2.7	Ambiguity in Grammars and Languages	T1,R2	1	
		2.8	Push Down Automata (PDA),	T1,R2	1	
		2.9	The Language of a PDA	T1,R2	1	
			Equivalence of PDAs and CFGs	T1,R2	1	
Content beyond syllabus		2.10	Elimination of ambiguity	R1	1	
				Total	**10**	
III	CO – 3		**UNIT-III: Lexical Analysis and Top-Down Parsing**			Chalk & Board; Power poir presentatior; Assignmen Test
		3.1	The structure of a compiler	T1,R2	1	
		3.2	Role of lexical analyzer	T1,R2	1	
		3.3	Input Buffering	T1,R2	1	
		3.4	Specification of tokens	T1,R2	1	
		3.5	Recognition of tokens	T1,R2	1	
		3.6	The Lexical Analyser Generator –LEX	T1,R2	1	
		3.7	Introduction to Syntax Analysis	T1,R2	1	
		3.8	Eliminating ambiguity	T1,R2	1	
		3.9	left recursion from a CFG	T1,T2	1	
		3.10	Recursive Decent Parsing	T1,T2	1	
		3.11	LL(1) Grammars	T1,R2	1	
		3.12	Nonrecursive Predictive Parsing		1	
Content beyond syllabus		3.13	Operator precedence parsing	R2	1	
				Total	**13**	

IV	CO – 4		**UNIT-IV: Bottom-Up Parsing and Syntax Directed Translation**			Chalk & Board Power poir presentatior Assignmen Test
		4.1	Shift-Reduce Parsing	T1,R2	1	
		4.2	Simple LR parsing	T1,R2	1	
		4.3	Canonical LR(1) Parsing	T1,R2	1	
		4.4	LALR Parsing	T1,R2	1	
		4.5	Parser Generators	T1,R2	1	
		4.6	Syntax Directed Definitions	T1,R2	1	
		4.7	Evaluation Orders for SDDs	T1,R2	1	
		4.8	Syntax Directed Translation Schemes	T1,R2	1	
Content beyond syllabus		4.15	Storage Allocation Strategies	R3	1	
				Total	**09**	
CO-5			**UNIT-V: Intermediate Code Generation, Code Optimization and Generation**			Chalk & Board Power poir presentatior Assignmen Test
		5.1	Three address code	T1,T2	1	
		5.2	Types and Declarations	T1,T2	1	
		5.3	Translation of Expressions	T1,T2	1	
		5.4	Type Checking	T1,T2	1	
		5.5	Control Flow	T1,T2	1	
		5.6	Basic Blocks and Flow Graphs	T1,T2	1	
		5.7	Optimization of Basic Blocks	T1,T2	1	
		5.8	Peephole Optimization	T1,T2	1	
		5.9	Issues in the design of a Code Generator	T1,T2	1	
		5.10	The Target Language	T1,T2	1	
		5.11	A simple Code Generator	T1,T2	1	
				Total	**11**	
				CUMULATIVE PROPOSED PERIODS	**57**	

Text Books:	
S.No	**AUTHORS, BOOK TITLE, EDITION, PUBLISHER, YEAR OF PUBLICATION**
1	J.E.Hopcroft, R.Motwani and J.D.Ullman, Introduction to Automata Theory, Languages and Computation, , 3rd Edition, Pearson, 2008.
2	Alfred V.Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Compilers Principles, Techniques and Tools, 2nd Edition, Pearson, 2022.
Reference Books:	
S.No.	**AUTHORS, BOOK TITLE, EDITION, PUBLISHER, YEAR OF PUBLICATION**
1	John C. Martin, McGraw Hill, Introduction to Languages and The Theory of Computation, 2020
2	K.L.P.Mishra and N.Chandrasekaran, Theory of Computer Science-Automata, Languages and Computation, , 3rd Edition, PHI, 2007
3	K.V.N. Sunitha, Compiler Construction, Pearson, 2013
4	SandeepSaxena, Rajkumar Singh Rathore, Compiler Design, , S.Chand publication, 2019

Web Details:	
1	https://www.geeksforgeeks.org/compiler-design-tutorials/
2	https://www.guru99.com/compiler-design-tutorial.html
3	https://www.tutorialspoint.com/compiler_design/compiler_design_overview.htm
4	https://www.greatlearning.in/academy/learn-for-free/courses/compiler-design

		Name	**Signature with Date**
i.	Faculty	Mr.K.Raja	K.Raj
ii.	Module Coordinator	Mr.K.Raja	K.Raj
iii.	Programme Coordinator	Dr. RVVSV Prasad	

Principal