VIT BHOPAL www.vitbhopal.ac.in	Theory of Computation and Compiler Design	Course Type	LT
Course Code:	**CSE2004**	**Credits**	**4**
Prerequisite:			

Course Outcomes (CO):

Students will be able to

CO1. Design finite automaton for different regular expressions and languages and its applications in lexical analysis [KL3]

CO2. Build a simplified context-free grammar for a context-free language to recognize by a Pushdown automation [KL3]

CO3. Demonstrate the syntax analysis process using a top-down and bottom-up parser [KL3]

CO4. Develop a Computational model using Turing machine to test decidability of a problem [KL3]

CO5. Develop the intermediate code representations and optimize them for code generation [KL3]

CO	Topics to be discussed	Hrs
CO1	Basic concepts – Theorem proving – Finite automata: NFA, DFA, € - NFA, Regular expressions - Equivalence between FA and RE – Minimization – Decision properties – Pumping lemma for Regular Languages. Specification of tokens – FA and RE to represent token formats – LEX. Problems: Design of FA – Inter-conversion between RE and FA – Proving languages to be not regular, Design approach of Lexical Analyzer for a given token – LEX program to recognize tokens	**16**
CO2	Context Free Grammar – Derivations – Parse trees – Ambiguity – Chomsky Normal Form – Griebach Normal Form – Pushdown Automata – DPDA & NPDA – Decision properties – Pumping lemma for CFL. Problems: Design of CFG – Conversion from CFG to CNF, GNF – Design of PDA – Inter-conversion between PDA & CFG – Proving languages to be not context-free	**14**
CO3	Parsing – Top-down Parsing – Predictive Parsing - Bottom up parsing – SLR, CLR and LALR Parsing – YACC. Problems: Design of Top-down parser and bottom-up parser to illustrate syntax validation of an input string	**12**
CO4	Turing machines – TM as a computation model – TM as a recognizer – TM with multiple tapes – Other models of TM – Linear Bounded Automata – Chomsky Hierarchy of languages – Undecidability – Recursive and non – recursive languages – Examples Problems: Design of TM – Design of LBA – Identification of Undecidability	**10**
CO5	Three Address Codes – Code optimization techniques – Code generation. Problems: Conversion from parse tree to TAC – optimization techniques – Code generation	**08**

	Total Lectures:	**60**
Text books:		
1	John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, “Introduction to Automata Theory, Languages and Computation”, 3rd Edition, Pearson Education, 2014.	
2	Alfred V. Aho, Monica S Lam, Ravi Sethi, Jeffery D Ullman, “ Compilers: Principles, Techniques, and Tools”, 2nd Edition, Pearson Education, 2015.	
Reference Books, Web reference:		
1	Michael Sipser, “Introduction to the Theory of Computation”, 2nd Edition, Wadsworth Publishing Co Inc, 3rd Edition, 2012.	
Recommendation by the Board of Studies on		
Approval by Academic council on:		
Compiled by:		

*KL – Revised Blooms Knowledge Level (Cognitive Domain)