PVP SIDDHARTHA INSTITUTE OF TEHNOLOGY, KANURU, VIJAYAWADA
(AUTONOMOUS)

INFORMATION TECHNOLOGY

COMPILER DESIGN

Course Code	19IT3601	**Year**	III	**Semester**	II
Course Category	PC	**Branch**	IT	**Course Type**	Theory
Credits	3	**L-T-P**	3-0-0	**Prerequisites**	Formal Language & Automata Theory
Continuous Internal Evaluation :	30	**Semester End Evaluation:**	70	**Total Marks:**	100

Course Outcomes		**Blooms Taxonomy Level**
Upon Successful completion of course, the student will be able to		
CO1	Understand about language processors and its phases.	L2
CO2	Demonstrate about scanning of tokens and perform the syntax analysis by using parsing techniques	L3
CO3	Perform Symantec analysis using attribute grammar and compare different memory management techniques in runtime environment	L3
CO4	Ascertain optimization techniques for intermediate code forms and code generation	L3

Contribution of Course Outcomes towards achievement of Program Outcomes & Strength of correlations (H:High, M: Medium, L:Low)

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	2											2	1
CO2	3	3	2	1	3							1	2	1
CO3	2	3		1	3							1	2	2
CO4	2	3	2		3							1	3	3

Syllabus		
Unit No	**Contents**	**Mapped CO**
I	**Overview of language processing:** preprocessors – compiler – assembler – Linkers & loaders, difference between compiler and interpreter-structure of a compiler –phases of a compiler. **Lexical Analysis:** Role of Lexical Analysis – Input Buffering – Specification of Tokens – Recognition of Token – The Lexical Analyzer Generator Lex.	**CO1**
II	**Syntax Analysis:** Role of a parser – Context Free Grammar – Top Down Parsing – Recursive Descent Parsing — Non recursive Predictive Parsing- FIRST and FOLLOW – LL(1) Grammar – Error Recovery in Predictive Parsing.	**CO1, CO2**
III	**Bottom up Parsing:** Reductions – Handle Pruning - Shift Reduce Parsing – Introduction to simple LR – Why LR Parsers – Model of an LR Parsers — Construction of SLR Tables. **More powerful LR parsers**: Construction of CLR (1) - LALR Parsing tables.	**CO1, CO3**
IV	**Runtime Environment:** Storage organization - Stack allocation – Static allocation – Heap management - Parameter passing mechanisms. **Intermediate code:** DAG - Three address code – Quadruples - Triples - Indirect Triples.	**CO1, CO3, CO4**
V	**Basic Blocks:** DAG representation of Block. Machine independent code optimization - Common sub expression elimination - Constant folding - Copy propagation -Dead code elimination - Strength reduction - Loop optimization. **Machine dependent code optimization:** Peephole optimization – Register allocation - Instruction scheduling - Inter Procedural Optimization - Garbage collection via reference counting.	**CO1, CO4**

Learning Resources
Text Books
1. Compilers: Principles, Techniques and Tools: 2nd Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, J effrey D. Ulman; 2nd Edition, Pearson Education. 2. Modern Compiler Implementation in C- Andrew N. Appel, Cambridge University Press, First edition.
References
1. lex &yacc – John R. Levine, Tony Mason, Doug Brown, O‟reilly, 2nd edition, 2017. 2. Modern Compiler Design- Dick Grune, Henry E. Bal, Cariel T. H. Jacobs, Wileyream tech, 2012. 3. Engineering a Compiler-Cooper & Linda, Elsevier, Third edition. 4. Compiler Construction, Louden, Thomson, First edition. 5. Principles of compiler design, V. Raghavan, 2nd edition, TMH, 2011.
E-Resources and other Digital Material
1. http://www.nptel.iitm.ac.in/downloads/106108052/