



JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA

KAKINADA – 533 003, Andhra Pradesh, India

DEPARTMENT OF INFORMATION TECHNOLOGY

II Year – II Semester		L	T	P	C
		3	0	0	3
AUTOMATA THEORY AND COMPILER DESIGN					

Course Objectives:

- To learn fundamentals of Regular and Context Free Grammars and Languages
- To understand the relation between Contexts free Languages, PDA and TM
- To study the various phases in the design of a compiler
- To understand the design of top-down and bottom-up parsers
- To understand syntax directed translation schemes
- To learn to develop algorithms to generate code for a target machine

Course Outcomes:

At the end of the course, the students will be able to:

- Ability to design, develop, and implement a compiler for any language
- Able to use LEX and YACC tools for developing a scanner and a parser
- Able to design and implement LL and LR parsers
- Able to design algorithms to perform code optimization in order to improve the performance of a program in terms of space and time complexity
- Ability to design algorithms to generate machine code

UNIT-I

Formal Language and Regular Expressions : Languages, Definition Languages regular expressions, Finite Automata – DFA, NFA. Conversion of regular expression to NFA, NFA to DFA. Applications of Finite Automata to lexical analysis, lex tools.

UNIT-II

Context Free grammars and parsing : Context free grammars, derivation, parse trees, ambiguity LL(K) grammars and LL(1) parsing

Bottom up parsing handle pruning LR Grammar Parsing, LALR parsing, parsing ambiguous grammars, YACC programming specification.

UNIT-III

Semantics: Syntax directed translation, S-attributed and L-attributed grammars, Intermediate code – abstract syntax tree, translation of simple statements and control flow statements.

Context Sensitive features – Chomsky hierarchy of languages and recognizers. Type checking, type conversions, equivalence of type expressions, overloading of functions and operations.

UNIT-IV

Run time storage : Storage organization, storage allocation strategies scope access to now local names, parameters, language facilities for dynamics storage allocation.

Code optimization : Principal sources of optimization, optimization of basic blocks, peephole optimization, flow graphs, Data flow analysis of flow graphs.

UNIT-V

Code generation : Machine dependent code generation, object code forms, generic code generation algorithm, Register allocation and assignment. Using DAG representation of Block.



JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA
KAKINADA – 533 003, Andhra Pradesh, India

DEPARTMENT OF INFORMATION TECHNOLOGY

TEXT BOOKS:

- 1) Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3rd Edition, Pearson, 2008.
- 2) Compilers Principles, Techniques and Tools Aho, Ullman, Ravisethi, Pearson Education.

REFERENCES:

- 1) Louden: “Compiler Construction, Principles & Practice”, 1st Edition, Thomson Press, 2006.
- 2) Tremblay J P, Sorenson G P: “The Theory & Practice of Compiler writing”, 1st Edition, BSP publication, 2010.
- 3) Theory of Computation, V. Kulkarni, Oxford University Press, 2013

e-Resources:

- 1) <https://nptel.ac.in/courses/106/104/106104028/>
- 2) <https://nptel.ac.in/courses/106/104/106104123/>