SCHEME OF COURSE WORK

Course Details:

Course Title	AUTOMATA and COMPILER DESIGN		
Course Code	13CS2110	**L T P C**	4 0 0 3
Programme:	M.Tech.		
Specialization:	Computer Science & Engineering		
Semester	II		
Prerequisites	Formal Languages and Automata Theory, Graph Theory.		
Courses to which it is a prerequisite	-		

Course Outcomes (CO):

At the end of the course, a student will be able to

1	Explain deterministic and non-deterministic machines.
2	Comprehend the hierarchy of problems arising in the computer sciences.
3	Design a deterministic finite-state machine to accept a specified language.
4	Explain how a compiler can be constructed for a simple context free language.
5	Determine a language's location in the Chomsky hierarchy (regular sets, context-free, context-sensitive, and recursively enumerable languages).

Programme Outcomes (PO):

A graduate of Computer Science &Engineering

1	Graduates will demonstrate knowledge in core subjects of Computer Science and Engineering and the ability to learn independently.
2	Graduates will demonstrate the ability to design a software application or process that meets desired specifications within the constraints.
3	Graduates will demonstrate the ability to solve problems relevant to industries and research organizations.
4	Graduates will develop innovative thinking capabilities to promote research in core and trans-disciplinary areas
5	Graduates will be familiar with modern engineering software tools and equipment to analyze computer science and engineering problems.
6	Graduates will demonstrate the ability to collaborate with engineers of other disciplines and work on projects requiring multidisciplinary skills.
7	Graduates will acquire project management and finance control abilities.
8	Graduates will be able to communicate effectively in both verbal and written forms.
9	Graduates will engage themselves in lifelong learning in the context of rapid technological changes in computer science and engineering
10	Graduates will demonstrate an appreciation of ethical and social responsibilities in professional and societal context.
11	Graduates will demonstrate the ability in carrying out tasks independently and by reflective Learning

Course Outcome versus **Program Outcomes:**

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO-1	S										M
CO-2			S	M					M		
CO-3		S									M
CO-4		M			M				M		
CO-5			S	M							

S - Strongly correlated, *M* - *Moderately* correlated, *Blank* - *No correlation*

Lesson Plan:

UNIT	Topic	No classes
Unit I	Formal Languages and Regular Expressions	1
	Languages, Definition	1
	Languages regular expressions	1
	Finite Automata – DFA, NFA	2
	Conversion of regular expression to NFA	2
	Conversion from NFA to DFA	1
	Applications of Finite Automata to lexical analysis	1
	lex tools	1
	No. of Classes	**10**
Unit II	Context Free grammars and parsing : Context free grammars	1
	derivation, parse trees	1
	ambiguity LL(K) grammars and LL(1) parsing	2
	Bottom up parsing, handle pruning	2
	LR Grammar Parsing, LALR parsing	2
	Parsing ambiguous grammars	1
	YACC programming specification	1
	No. of Classes	**10**
Unit III	Semantics: Syntax directed translation	1
	S-attributed and L-attributed grammars	2
	Intermediate code – abstract syntax tree	1
	translation of simple statements and control flow statements.	1
	Context Sensitive features – Chomsky hierarchy of languages and recognizers	2
	Type checking, type conversions,	1
	equivalence of type expressions,	1
	overloading of functions and operations.	1
	No. of Classes	**10**

Unit IV	Symbol table, Storage organization	1
	storage allocation strategies scope access to now local names, parameters	2
	language facilities for dynamics storage allocation	1
	Code optimization Principal sources of optimization, optimization of basic blocks	2 1
	peephole optimization	1
	flow graphs, optimization techniques.	2
	No. of Classes	**10**
Unit V	Code generation: Machine dependent code generation	2
	object code forms	2
	generic code generation algorithm	2
	Register allocation and assignment	2
	Using DAG representation of Block	2
	No. of Classes	**10**
	Total No. of Classes	**50**

Assessment Methods:	Assignment / Quiz / Mid-Test / End Exam

Teaching-Learning and Evaluation

Week	Topics/Contents	Course Outcomes	Sample Questions	Teaching - Learning Strategy	Assessment Method & Schedule
1	Formal Languages and Regular Expressions, Languages, Definition, Languages regular expressions	CO-1	1) Define Language and Regular expression	Lecture Discussion Solving Exercise	Quiz(Week-4) Assignment (Week 6- 8) Mid Test-1
2	Finite Automata – DFA, NFA, Conversion of regular expression to NFA	CO-1	1) Differentiate DFA and NFA. 2) Convert the following Regular Expression into NFA $(011+10)^*$	Lecture Discussion Solving Exercise	Quiz(Week-4) Assignment (Week 6- 8) Mid Test-1
3	Conversion from NFA to DFA, Applications of Finite ,Automata to lexical analysis, lex tools	CO-1	1) Design DFA for the language os strings ends with 11 and convert the NFA to DFA.	Lecture Discussion Solving Exercise	Quiz(Week-4) Assignment (Week 6- 8) Mid Test-1
4	Context Free grammars and parsing : Context free grammars, derivation, parse	CO-2	1) Define Context Free Grammar and generate the parse	Lecture Discussion Solving	Quiz(Week-4) Assignment (Week 6- 8) Mid Test-1

	trees ambiguity LL(K) grammars and LL(1) parsing		tree for the following grammar. S->AS/BS/^ A->aA/^ B->bB.	Exercise	
5	Bottom up parsing, handle pruning LR Grammar Parsing, LALR parsing	CO-2	1) Construct SLR parsing table for the following grammar. E->E+T/T T->T*F F->(E)/id	Lecture Discussion Solving Exercise	Quiz(Week-8) Assignment (Week 6- 8) Mid Test-1
6	Parsing ambiguous grammars YACC programming specification	CO-2	1) Explain about Yacc tool with an example.	Lecture Discussion Solving Exercise	Quiz(Week-8) Assignment (Week 6- 8) Mid Test-1
7	Semantics: Syntax directed translation S-attributed and L-attributed grammars	CO-3	1) Explain about S-attributed and L-attributed grammars.	Lecture Discussion	Quiz(Week-8) Assignment (Week 6- 8) Mid Test-1
8	Intermediate code – abstract syntax tree translation of simple statements and control flow statements.	CO-3	1) Represent the following expression in Triples, indirect Triples X=(a+b)*(c+d)	Lecture Discussion	Quiz(Week-8) Assignment (Week 6- 8) Mid Test-1
9	**MID TEST-1**			Revision Discussion	
10	Context Sensitive features – Chomsky hierarchy of languages and recognizers	CO-3	1) Explain about chomsky hierarchy languages with an examples.	Lecture Discussion	Quiz(Week-12) Assignment (Week 15- 16) Mid Test-2
11	Type checking, type conversions, equivalence of type expressions, overloading of functions and operations.	CO-3	1) Explain about Type checking and type conversions.	Lecture Discussion	Quiz(Week-12) Assignment (Week 15- 16) Mid Test-2
12	Symbol table, Storage organization storage allocation strategies scope access to now local names, parameters	CO-4	1) Explain about various storage allocation strategies of symbol table.	Lecture Discussion	Quiz(Week-12) Assignment (Week 15- 16) Mid Test-2
13	language facilities for dynamics storage allocation Code optimization Principal sources of optimization	CO-4	1) Explain about various code optimization techniques.	Lecture Discussion	Quiz(Week-15) Assignment (Week 15- 16) Mid Test-2
14	optimization of basic blocks peephole optimization flow graphs, optimization	CO-4	1) what is the importance of peephole	Lecture Discussion	Quiz(Week-15) Assignment (Week 15- 16) Mid Test-2

	techniques.		optimization and explain with an example.		
15	Code generation: Machine dependent code generation object code forms generic code generation algorithm	CO-5	1) Explain about various object forms in code generation.	Lecture Discussion	Quiz(Week-15) Assignment (Week 15- 16) Mid Test-2
16	Register allocation and assignment Using DAG representation of Block	CO-5	1) Briefly explain register allocation and assignment strategies in code generation.	Lecture Discussion	Quiz(Week-15) Assignment (Week 15- 16) Mid Test-2
17	**MID TEST-2**			Revision Discussion	
18	**END EXAM**				

Faculty Member
N.Durga Prasad
Assistant Professor
CSE Department