

GOPALAN COLLEGE OF ENGINEERING AND MANAGEMENT

Department of Computer Science and Engineering

Academic Year: **2016-17**

Semester: **EVEN**

COURSE PLAN

Semester: **V**

Subject Code& Name: **10CS63 & Compiler Design**

Name of Subject Teacher: **ARVIND R**

Name of Subject Expert (Reviewer): **N S SARADHA DEVI**

For the Period: From: 13-02-17 to 02-06-17

Details of Book to be referred:

Text Books	T1: Alfred V Aho, Monica S. Lam ,Ravi Sethi, Jeffrey D Ullman: Compilers-Principles, Techniques and Tools,2nd Edition,Pearson Education,2007.
Reference Books	R1: Charles N. Fischer, Richard J. leBlanc, Jr. Crafting a compiler with C, Pearson Education, 1991. R2: Andrew W Apple: Modern Compiler Implementation in C, Cambridge University Press, 1997. R3: Kenneth C Loudon: Compiler Construction Principles and Practice, Cengage Learning, 1997.

Lecture NO	Topic Planned	Practical Applications & Brief objectives	Book referred with Pg No.	Planned Date	Executed Date	Deviation Reasons thereof	How Made Good / Reciprocate arrangement	Remarks by HOD
1.	Introduction to the subject Unit 1 : Introduction	Objective: Deals with Phases of compiler, application of compiler in various streams,		13-02-17				
2.	Language processors The Structure of a Compiler		T1: 1-4	15-02-17				
3.	The Structure of a Compiler		T1: 5-12	16-02-17				

4.	The Evolution of programming Languages -The Science of building a Compiler	specification and recognition of tokens Application: RISC and CISC MACHINES OUTCOME: Able to understand different phases of compiler, applications of compiler technology, lexical analysis	T1: 12-17	20-02-17				
5.	Applications of Compiler Technology		T1: 17-25	20-02-17				
6.	Programming Language Basics		T1: 25-35	22-02-17				
7.	Lexical analysis: The Role of Lexical Analyzer, Input Buffering		T1: 109-116	23-02-17				
8.	Specification of Tokens		T1: 116-125	27-02-17				
9.	Recognition of Tokens		T1: 128-136	27-02-17				
10.	Revision / Unit Test			01-03-17				
11.	UNIT-2:Syntax Analysis-1 Introduction		Objective: Understand the CFG and its derivations, ambiguity, left recursion and left factoring Application: Assemblers, System Softwares OUTCOME: Helps in eliminating ambiguity, left recursion, designing predictive parsers	T1: 191-196	02-03-17			
12.	Context Free Grammars			T1: 196-201	03-03-17			
13.	Context Free Grammars			T1: 202-206	04-03-17			
14.	Writing a Grammar	T1: 209-212		06-03-17				
15.	Top-Down Parsing	T1: 213-216		06-03-17				
16.	Top-Down Parsing contd..	T1: 217-226		08-03-17				
17.	Top-Down Parsing contd..	T1: 226-230		13-03-17				
18.	Revision / Unit Test			13-03-17				
19.	UNIT-3:Syntax-Analysis-2 Bottom-UP Parsing	Objective: Introduce the concept of Shift-reduce parser, error	T1: 233-235	15-03-17				
20.	Bottom-UP Parsing contd...		T1: 236-240	16-03-17				

21.	Bottom-UP Parsing contd...	recovery strategies, SLR parsers Application: Programming languages and databases Outcome: Bottom-Up parsing, LR parsers :SLR parser	T1: 236-240	17-03-17				
22.	Introduction to LR Parsing		T1: 241-245	20-03-17				
23.	Simple LR contd...		T1: 246-252	20-03-17				
24.	Simple LR contd...		T1: 253-257	22-03-17				
25.	Revision/Class Test			23-03-17				
26.	Unit-4:Syntax-Analysis-3 More Powerful Parsers	Objective: Understand LR parsers, CLR,LALR Parsers, YACC parser generators Application: Operating system,Assemblers OUTCOME: helps in Design of LR,CLR,LALR parsers, LR(1) items, Yacc generator	T1:259-263	27-03-17				
27.	More Powerful Parsers Contd...		T1: 264-266	27-03-17				
28.	More Powerful Parsers contd...		T1: 268	30-03-17				
29.	More Powerful Parsers contd...		T1: 269-270	31-03-17				
30.	Using ambiguous grammars		T1: 278-284	03-04-17				
31.	Parser Generators		T1: 287-296	03-04-17				
32.	Revision / Unit Test			05-04-17				
33.	Unit-5:Syntax Directed Translation: Syntax-directed definitions	Objective: Introduce the Concept of SDD, attributes, S & L- attributes, SDT schemes	T1: 303-306	06-04-17				
34.	Syntax-directed definitions contd...		T1: 306-309	07-04-17				
35.	Evaluation orders for SDDs		T1: 310-314	10-04-17				
36.	Evaluation orders for SDDs contd...		T1: 315-317	10-04-17				

37.	Applications of Syntax-directed translation	Application: Programming OUTCOME: Helps in implementing SDDs, Annotated parse Tree, Types of attributes, Structure of C Types, different schemes of SDT	T1: 318-320	12-04-17				
38.	Applications of Syntax-directed translation		T1: 320-323	13-04-17				
39.	Syntax-directed translation schemes		T1: 324-325	20-04-17				
40.	Syntax-directed translation schemes		T1: 326-336	21-04-17				
41.	Revision / Unit Test			24-04-17				
42.	Unit 6:Intermediate Code Generation : -Variants of syntax tree	Objective: Deals with DAGs, Triples, quadruples, conditional statement backpatching Application: Implementing c,c++ programs,functions OUTCOME: Designing DAG,3AC,Type and declaration, ,Backpatching	T1: 357-362	24-04-17				
43.	Three-address code		T1: 363-370	26-04-17				
44.	Translation of expressions		T1: 378-384	27-04-17				
45.	Control Flow		T1: 399-408	28-04-17				
46.	Back-patching		T1: 410-417	03-05-17				
47.	Switch statements procedures		T1: 418-423	04-05-17				
48.	Revision / Unit Test			05-05-17				
49.	UNIT -7:Run-Time Environments: Storage organization -Stack allocation of space		Objective: Deals with concept of Storage Organization, Accessing Non-local data Application: Operating system processors	T1: 427-433	08-05-17			
50.	Stack allocation of space	T1: 434-440		08-05-17				
51.	Access to non-local data on the stack	T1: 441-445		10-05-17				

52.	Access to non-local data on the stack	OUTCOME: Understands static versus dynamic allocation, activation records, heap management using fragmentation techniques, garbage collector	T1: 445-451	11-05-17				
53.	Heap Management		T1: 452-463	12-05-17				
54.	Introduction to garbage Collection		T1: 463-468	18-05-17				
55.	Revision / Unit Test			19-05-17				
56.	Unit 8:Code Generation Issues in the design of code generator	Objective: Design of code generator, Design of basic blocks &flow graphs Application: Micro Processors Macro Processors OUTCOME: Understands Issues in code generation, implementing basic blocks and flow graphs, designing optimized DAG	T1: 505-511	22-05-17				
57.	The target language Addresses in the target code		T1: 512-524	22-05-17				
58.	Basic blocks and Flow graphs		T1: 525-531	24-05-17				
59.	Optimization of basic blocks		T1: 533-541	25-05-17				
60.	A Simple Code Generator		T1: 542-548	26-05-17				
61.	Revision / Unit Test			29-05-17				
62.	Revision	Solving VTU Question Paper		29-05-17				
63.	Revision			30-05-17				
64.	Revision			01-06-17				
65.	Revision			02-06-17				

Prepared By: **ARVIND R**
(Faculty)
Date & Sign _____

Reviewed by: **N S SARADHA DEVI**
(Sub. expert)
Date & Sign _____

Approved by: _____
(HOD)
Date & Sign _____

Approved by: _____
(Principal/ Acad. Co)
Date & Sign _____