PSN College of Engineering and Technology (Autonomous)

Tirunelveli - 627152

COURSE PLAN

Academic year (2023-24)

(Regulation- 2018)

Subject Name	Compiler Design
Course Type	Core Paper
Programme	B. E , Computer Science and Engineering
Year/ Semester/ Section	III / V
Nature of Course / Credit	Theory / 3
Course Coordinator	Ms. I. Swetha
Course Code	503012

Course File Verification and Auditing

Part-I

(At the beginning of the semester)

Submission Date	Check List								Verified by HOD	Verified by Academic Auditor
	Vision and Mission	Course Description, Objective and Outcomes	CO-PO mapping	Course Plan and Target	Syllabus and Content beyond Syllabus	Assignments & additional resources	Course Delivery Plan	University Question Papers		

Part-II (After CAT - I)

Submission Date	Check List					Verified by HOD	Verified by Academic Auditor
	Syllabus Coverage	Notes and Other Materials	Performance Analysis	Feedback	Proof for Participatory Learning		

Part-III (After CAT - II)

Submission Date	Check List					Verified by HOD	Verified by Academic Auditor
	Syllabus Coverage	Notes and Other Materials	Performance Analysis	Question Papers and Keys	Proof for Participatory Learning		

Part-IV (After Model examination)

Submission Date	Check List					Verified by HOD	Verified by Academic Auditor
	Syllabus Coverage	Notes and Other Materials	Performance Analysis	Question Papers and Keys	Proof for Participatory Learning		

Semester Academic Audit

Audit Remarks:

Signature of the Auditor(s):

Signature of Director (Academics) Signature of Principal

Contents

Sl. No.	Description	Page No.
1	Pre-requisite	
2	Course Description	
3	Career Opportunities	
4	Syllabus	
5	Course Outcome (COs)	
6	Instructional Learning outcomes	
7	Program Educational Objectives (PEOs)	
8	Program Outcomes (POs)	
9	Program Specific Outcomes (PSOs)	
10	CO and PO mapping	
11	Text Books & Reference Books	
12	Web resources	
13	E – learning links	
14	Magazines & Journals	
15	Lesson Plan	
16	Class time table	
17	Course time table	
18	Content Delivery Methodologies	
19	Assignments	
20	Assignment Rubrics	
21	Mapping of CO to Assignment	
22	Assessment Methodologies	
23	Distribution of portions for assessment tests	
24	Mark Allotment for CO Assessment	
25	Lecture Notes	
26	Content beyond syllabus	
27	Question Bank	
28	End Semester questions papers	
29	Students' Name list	
30	Identification of fast and slow learners	
31	Remedial Action for slow learners	
32	Encouragements for fast learners	
33	Students' group list for topic discussion	
34	Course Review & Closure Report	

VISION AND MISSION OF THE INSTITUTE:

Institution Vision	Emerge as a pioneer institute inculcating engineering education and skills, research, values and ethics.	
Institution Mission	**IM-1**	To achieve greater heights of excellence in technical knowledge and skill development through innovative teaching and learning practices.
	IM-2	To develop the state of art infrastructure to meet the demands of technological revolution.
	IM-3	To improve and foster research in all dimensions for betterment of society.
	IM-4	To develop individual competencies to enhance innovation, employability and entrepreneurship among students.
	IM-5	To instill higher standards of discipline among students, inculcating ethical and moral values for societal harmony and peace.

VISION AND MISSION OF THE DEPARTMENT:

Department Vision	To emerge as a preeminence program to produce quality Computer Science and Engineering graduates.	
Department Mission	**DM-1**	To enhance professional and entrepreneurial skills through industry institute interaction to enable them in getting better placement.
	DM-2	To promote research and continuing education.
	DM-3	To train the students according to their discipline to meet dynamic needs of the society.

1. **PRE REQUISITES:**
    - Data structures
    - Computer architecture

2. **COURSE DESCRIPTION:**

    A compiler is a fundamental software necessary to translate computer programs to a form that can be executed on intended machines. Designing a compiler involves learning several aspects of computer science: logic, formalism, mathematics, data structures, algorithms, programming, and so on. This course is intended as a primer to the various stages typical in the design of standard compilers, starting with the front-end stages of compilation, and giving a peek into the back-end and some recent advancements in the area. At the end of the course, students should be able to appreciate the underlying concepts in compiler design, and be motivated to learn the art of analyzing and transforming programs for performance.

3. **Career Opportunities:**
    - Computer Software Field

4. **SYLLABUS**

503012	**COMPILER DESIGN**			**L**	**T**	**P**	**C**
				3	**0**	**2**	**4**
Programme:	B.E (CSE)	**Sem:**	**5**	**Category:**			**CS**
Aim:	To obtain an in-depth knowledge of compiler design and optimization techniques						
Course Objectives:							
1	To understand the process of converting modern high-level programming language into executable code and the various phases of Compiler						
2	To evaluate the different parsers of compiler						
3	To understand about the intermediate representation of compiler						
4	To understand the code generation and code optimization techniques.						
5	To apply optimization techniques to attain better code for code generation.						

UNIT - I	**INTRODUCTION AND LEXICAL ANALYSIS**	**Hrs**
Introduction: Language Processors-Phases of a Compiler-The Science of Building Compiler Applications of Compiler Technology-Cousins of a compiler-Compiler Construction Tools, Lexical Analysis: The Role of the Lexical Analyzer-Specification of Tokens-Recognition of Tokens-Finite Automata		**12**
UNIT - II	**SYNTAX ANALYSIS**	**Hrs**

Syntax Analysis: Role of the Parser- Writing a Grammar - Top Down Parsing - Recursive Descent Parsing - Predictive Parsing - Bottom Up Parsing - Shift Reduces Parsing -LR Parsers-SLR Parser-CLR Parser		**12**
UNIT - III	**INTERMEDIATE CODE GENERATION**	**Hrs**
Syntax-Directed Translation: Syntax-Directed Definitions - Applications of Syntax - Directed Translation - Syntax-Directed Translation Schemes -Intermediate Code Generation - Types of Three Address Code - Type Checking -Control Flow Statements - Switch Statements – Procedures - Backpatching		**12**
UNIT - IV	**CODE GENERATION**	**Hrs**
Code Generation: Issues in the Design of a Code Generator - Basic Blocks and Flow Graphs - Optimization of Basic Blocks - Peephole Optimization - Register Allocation and Assignment Optimal Code Generation for Expressions - Dynamic Programming Code – Generation		**12**
UNIT - V	**CODE OPTIMIZATION**	**Hrs**
Introduction to Optimization: Background-Scope of Optimization-Local and Global Optimization - Inter-procedural Optimization. Dataflow Analysis: Iterative Data Flow Analysis - Static Single Assignment Form - Inter-procedural Analysis		**12**
Total:60 Periods		

List of Experiments:

1. Implementation of Symbol Table
2. Develop a lexical analyzer to recognize a few patterns in C.
3. Write a C program to test whether a given identifier is valid or not.
4. Write a C program to simulate lexical analyzer for validating operators
5. Program to recognize a valid arithmetic expression that uses operator +, –, * and /.
6. Program to recognize a valid variable which starts with a letter followed by any number of letters or digits.

Hardware / Software Requirements

Hardware

- LAN System (OR) Standalone PCs

Software

- OS – Windows 2000 Professional/XP
- Turbo C

5. COURSE OUTCOMES

After successful completion of this course, the students should be able to

CO'S	CO STATEMENTS	BLOOMS LEVEL	PO'S
CO 1	Understand the various phases in a compiler and to understand the knowledge of Lexical Analyzer	Understand	1,2
CO 2	Evaluate the parsers and experiment the knowledge of different parsers design	Evaluate	1,2
CO 3	Understand the knowledge of Intermediate code representations and generation	Understand	1,2
CO 4	Understand Conversion of source code into machine code for a novel computer	Understand	1,2,3
CO 5	Apply the various optimization techniques for dataflow analysis	Apply	1,2

6. INSTRUCTIONAL LEARNING OUTCOMES

Unit	Assessment Procedure
I	The outcome will be assessed through assignment-1, tutorial-1, Class test -1, MCQ Test-1 and CAT-1.
II	The outcome will be assessed through assignment-2, tutorial-2, Class test -2, MCQ Test-2 and CAT-1&2.
III	The outcome will be assessed through assignment-3, tutorial-3, Class test -3, MCQ Test-3 and CAT-2.
IV	The outcome will be assessed through assignment-4, tutorial-4, Class test -4, MCQ Test-4 and CAT-3.
V	The outcome will be assessed through assignment-5, tutorial-5, Class test -5, MCQ Test-5 and CAT – 3

7. PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

S.No	Topic	PEOs
PEO1	Fundamental Knowledge	Graduates will be able to perform in technical and managerial roles ranging from design, development and problem solving to suit to the industrial needs
PEO2	Career Development	Graduates will be able to successfully pursue higher education and also Graduates will have the ability to adapt, contribute and innovate new technologies in different domains of computer science & Engineering

PEO3	Social Identity	Graduates will be ethically and socially responsible engineers in computer science & Engineering disciplines

8. PROGRAM OUTCOMES (POs)

PO's No	KNOWLEDGE	STATEMENTS
1	Engineering Knowledge	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2	Problem Analysis	Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3	Design / Development of Solutions	Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4	Conduct Investigations of Complex Problems	Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5	Modern Tool usage:	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an UN 2 of the limitations.
6	The Engineer and Society	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7	Environment and Sustainability	Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8	Ethics	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9	Individual and Team Work	Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10	Communication	Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11	Project Management and Finance	Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12	Life-long Learning	Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

9. PROGRAM SPECIFIC OBJECTIVE (PSOs)

PSO1	Proficient and Innovative with a strong cognizance in the IOT, through the application of acquired knowledge and skills.
PSO2	Design and Implement IOT based solutions for improving operational efficiency by investigating existing industrial environment.

10. CO ,PO and PSO MAPPING

Course Outcomes	**Program Outcomes**												**Program Specific Outcomes**	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO 1	3		1		1					1	2	3		
CO 2		3	3	3	1							3		
CO 3	3		2									3		
CO 4	3	3										2		
CO 5	3			2								3		

Enter correlation levels 1, 2 & 3 define as: 1-low, 2- medium, & 3-high

11. **TEXT BOOK & REFERENCE BOOK LIST**

S.No.	Name of the Book	Author/Publisher/Year/Edition	Web link
1	Compilers Principles, Techniques, Tools	Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman / Pearson Education, Second Edition, 2014.	https://www.pdfdrive.com/compilers-principles-techniques-tools-e158432002.html
2	Engineering a Compiler	Keith D. Cooper & Linda Torczon/ Morgan Kaufmann, Second Edition, 2012.	https://vdocuments.mx/engineering-a-compiler-second-edition.html
3	Modern Compiler Design	David Galles / Pearson Education Asia, First Edition, 2009	https://www.google.co.in/books/edition/Modern_Compiler_Design/RcucHgMtSPAC?hl=en&gbpv=1&dq=%E2%80%9CModern+Compiler+Design%E2%80%9D&printsec=frontcover

12. **Web Resources:**

Sl. No	Topic	Web link
1.	Compiler design	https://www.tutorialspoint.com/compiler_design/index.htm

13. E – learning videos/NPTEL

Video	https://www.youtube.com/playlist?list=PLEbnTDJUr_IcPtUXFy2b1sGRPsLFMghhS
Lecture Notes	https://www.brainkart.com/subject/Compiler-Design_386/

14. **Magazines & Journals**

Magazine	https://durofy.com/phases-of-compiler-design
Journals	https://ijcsmc.com/docs/papers/October2014/V3I10201482.pdf

15. Lesson Plan & Content Delivery Methodologies

S. No.	Unit	Topic to be covered	Hours Needed	Mode of Teaching	Text/ Ref. Book	Page No.
INTRODUCTION AND LEXICAL ANALYSIS						
1	I	Introduction: Language Processors	1	BB	T1	1
2		Phases of a Compiler	1	BB	T1	4
3		The Science of Building Compiler	1	BB	T1	15
4		Applications of Compiler Technology	1	BB	T1	17
5		Cousins of a compiler-Compiler Construction Tools	1	BB	T1 WEB	12
6		Lexical Analysis: The Role of the Lexical Analyzer	1	BB	T1	109
7		Specification of Tokens	2	BB	T1	116
8		Recognition of Tokens	2	BB	T1	128
		Lexical-Analyzer Generator Lex*	1	Web		
9		Finite Automata	2	BB	T1	147
		Implementation of Symbol Table	3			
SYNTAX ANALYSIS						
10	II	Syntax Analysis: Role of the Parser	1	BB	T1	191
11		Writing a Grammar	2	BB	T1	209
12		Top Down Parsing - Recursive Descent Parsing	1	BB	T1	217
13		Predictive Parsing	1	BB	T1	226
14		Bottom Up Parsing - Shift Reduces Parsing	1	BB	T1	233
15		LR Parsers-SLR Parser	1	BB	T1	241
16		CLR Parser	1	BB	T1	259
		Develop a lexical analyzer to recognize a few patterns in C	3			
INTERMEDIATE CODE GENERATION						
17		Syntax-Directed Translation: Syntax-Directed Definitions	1	BB	T1	304
18		Applications of Syntax - Directed Translation	1	BB	T1	318
19		Syntax-Directed Translation Schemes	1	BB	T1	324
20		Intermediate Code Generation: Types of Three Address Code	2	BB	T1	363
21		Type Checking	2	BB	T1	386
22		Control Flow Statements	2	BB	T1	399
23		Switch Statements	1	BB	T1	418
24		Procedures	1	BB	Web	
25		Backpatching	1	BB	T1	410
		Write a C program to test whether a given identifier is valid or not	3			

S. No.	Unit	Topic to be covered	Hours Needed	Mode of Teaching	Text/ Ref. Book	Page No.
CODE GENERATION						
.26	**IV**	Code Generation: Issues in the Design of a Code Generator	2	BB	T1	506
27		Basic Blocks and Flow Graphs	2	BB	T1	525
28		Optimization of Basic Blocks	2	BB	T1	533
29		Peephole Optimization	2	BB	T1	549
30		Register Allocation and Assignment	2	BB	T1	553
31		Optimal Code Generation for Expressions	2	BB	T1	567
32		Dynamic Programming Code – Generation	2	BB	T1	573
		Write a C program to simulate lexical analyzer for validating operators	3			
		Program to recognize a valid arithmetic expression that uses operator +, –, * and /	3			
CODE OPTIMIZATION						
33	**V**	Introduction to Optimization: Background	1	BB	R1	405,407
34		Scope of Optimization	1	BB	R1	417
35		Local Optimization	2	BB	R1	420
		Regional Optimization*	1	Web		
36		Global Optimization	2	BB	R1	445
37		Inter-procedural Optimization	1	BB	R1	457
38		Dataflow Analysis: Iterative Data Flow Analysis	2	BB	R1	477
39		Static Single Assignment Form	2	BB	R1	495
40		Inter-procedural Analysis	1	BB	R1	519
		Program to recognize a valid variable which starts with a letter followed by any number of letters or digits	3			
Total Hours Needed: 45(L) +15(T)+ 3* = 63 Hour						

Signature	Prepared by:	Approved by:	
Name :	**Ms.I.Swetha**	**Dr.M.Vargheese**	**Dr.V.Manikandan**
Designation:	Asst. Professor	HoD / CSE	Principal
Date:			