SAHYADRI
COLLEGE OF ENGINEERING & MANAGEMENT
An Autonomous Institution
MANGALURU

SYSTEM SOFTWARE AND COMPILER DESIGN
(Effective from the Academic Year 2023 - 2024)
VI SEMESTER

Course Code	**21CS62**	CIA Marks	50
Number of Contact Hours/Week (L: T: P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40L + 20P	Exam Hours	03

CREDITS – 4

COURSE PREREQUISITES:

- Computer Orgainzation , Theory of Computation, Operating System, Knowledge of programming language (C, C++, java)

COURSE OBJECTIVES:

- Define System Software.
- Familiarize with source file, object file and executable file structures and libraries
- Describe the front-end and back-end phases of compiler and their importance to students

TEACHING - LEARNING STRATEGY:

Following are some sample strategies that can be incorporate for the Course Delivery

- Chalk and Talk Method/Blended Mode Method
- Power Point Presentation
- Expert Talk/Webinar/Seminar
- Video Streaming/Self-Study/Simulations
- Peer-to-Peer Activities
- Activity/Problem Based Learning
- Case Studies
- MOOC/NPTEL Courses
- Any other innovative initiatives with respect to the Course contents

COURSE CONTENTS

MODULE - I	
Introduction to System Software, Machine Architecture of SIC and SIC/XE. Assemblers: Basic assembler functions, machine dependent assembler features, machine independent assembler features, assembler design options. Basic Loader Functions	**8 Hours**
MODULE - II	
Introduction: Language Processors, The structure of a compiler, The evaluation of programming languages, The science of building a compiler, Applications of compiler technology. **Lexical Analysis**: The role of lexical analyser, Input buffering, Specifications of token, recognition of tokens.	**8 Hours**
MODULE - III	
Syntax Analysis: Introduction, Context Free Grammars, Writing a grammar, TopDown Parsers, Bottom-Up Parsers	**8 Hours**
MODULE - IV	
Syntax Directed Translation, Intermediate code generation, Code generation	**8 Hours**

MODULE – V	
Lex and Yacc –The Simplest Lex Program, Grammars, Parser-Lexer Communication, A YACC Parser, The Rules Section, Running LEX and YACC, LEX and Hand- Written Lexers, Using LEX - Regular Expression, Examples of Regular Expressions, A Word Counting Program, Using YACC – Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity.	**8 Hours**

COURSE OUTCOMES

Upon completion of this course, the students will be able to:

CO No.	Course Outcome Description	Bloom's Taxonomy Level
CO1	Apply the concepts of system software and make use of it to generate machine codes	CL3
CO2	Make use of the functionality of each phase involved in compilation process and construct the grammar for the given regular expressions	CL3
CO3	Apply the parsing techniques for the given programming construct described in context free grammar	CL3
CO4	Construct syntax directed tree and develop machine level codes	CL3
CO5	Make use of LEX and YACC tool to describe the concept of lexer and parser	CL3

LABORATORY COMPONENTS

Exp. No.	Experiment Description	CO No.	Bloom's Taxonomy Level
1.	Write a LEX program to recognize valid arithmetic expression. Identifiers in the expression could be only integers and operators could be + and *. Count the identifiers & operators present and print them separately.	CO2	CL3
2.	Develop, Implement and Execute a program using YACC tool to recognize all strings ending with b preceded by n a's using the grammar a n b (note: input n value)	CO3	CL3
3.	Design, develop and implement YACC/C program to construct Predictive / LL(1) Parsing Table for the grammar rules: A →aBa , B →bB | ε. Use this table to parse the sentence: abba$	CO3	CL3
4.	Design, develop and implement YACC/C program to demonstrate Shift Reduce Parsing technique for the grammar rules: E →E+T | T, T →T*F | F, F →(E) | id and parse the sentence: id + id * id.	CO3	CL3
5.	Design, develop and implement a C/Java program to generate the machine code using Triples for the statement A = -B * (C +D) whose intermediate code in three-address form: T1 = -B T2 = C + D T3 = T1 + T2 A = T3	CO1	CL3
6.	Write a LEX program to eliminate comment lines in a C program and copy the resulting program into a separate file.	CO4	CL3

CO-PO-PSO MAPPING

CO No.	Programme Outcomes (PO)												Programme Specific Outcome (PSO)		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO1	3	2		1									1		1
CO2	2	1		1											
CO3	2	2		1		1							1	1	1
CO4	3					1									1
CO5	2	2		1									1	1	

3: Substantial (High)	2: Moderate (Medium)	1: Poor (Low)

ASSESSMENT STRATEGY

Assessment will be both CIA and SEE. Students learning will be assessed using Direct and Indirect methods:

Sl. No.	Assessment Description	Weightage (%)	Max. Marks
1	**Continuous Internal Assessment (CIA)**	**100 %**	**50**
	Continuous Internal Evaluation (CIE)	60 %	30
	Practical Session (Laboratory Component)	40 %	20
2	**Semester End Examination (SEE)**	**100 %**	**50**

ASSESSMENT DETAILS

Continuous Internal Assessment (CIA) (50%)				Semester End Exam (SEE) (50%)
Continuous Internal Evaluation (CIE) (60%)			Practical Sessions (40%)	
I	II	III		
Syllabus Coverage			Syllabus Coverage	Syllabus Coverage
40%	30%	30%	100%	100%
MI			MI	MI
MII	MII		MII	MII
	MIII		MIII	MIII
		MIV	MIV	MIV
		MV	MV	MV

NOTE:

- Assessment will be both CIA and SEE.
- The practical sessions of the IPCC shall be for CIE only.
- The Theory component of the IPCC shall be for both CIA and SEE respectively.
- The questions from the practical sessions shall be included in Theory SEE.

Note: For Examinations (both CIE and SEE), the question papers shall contain the questions mapped to the appropriate Bloom's Level. Any COs mapped with higher cognitive Bloom's Level may also be assessed through the assignments.

SEE QUESTION PAPER PATTERN:

1. The question paper will have **TEN** full questions from **FIVE** Modules
2. There will be 2 full questions from each module. Every question will carry a maximum of 20 marks.
3. Each full question may have a maximum of four sub-questions covering all the topics under a module.
4. The students will have to answer FIVE full questions, selecting one full question from each module.

REFERENCE BOOKS:

1. System Software by Leland. L. Beck, D Manjula, 3rd edition, 2012
2. Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman , Compilers-Principles, Techniques and Tools, Pearson, 2nd edition, 2007
3. Doug Brown, John Levine, Tony Mason, lex & yacc, O'Reilly Media, October 2012.
4. Systems programming – Srimanta Pal , Oxford university press, 2016
5. System programming and Compiler Design, K C Louden, Cengage Learning
6. System software and operating system by D. M. Dhamdhere TMG
7. Compiler Design, K Muneeswaran, Oxford University Press 2013.

REFERENCE WEB LINKS AND VIDEO LECTURES (E - RESOURCES):

1. https://onlinecourses.nptel.ac.in/noc22_cs93/unit?unit=17&lesson=18
2. https://onlinecourses.nptel.ac.in/noc22_cs93/unit?unit=26&lesson=27
3. https://onlinecourses.nptel.ac.in/noc22_cs93/unit?unit=43&lesson=44
4. https://onlinecourses.nptel.ac.in/noc22_cs93/unit?unit=52&lesson=53
5. https://onlinecourses.nptel.ac.in/noc22_cs93/unit?unit=75&lesson=76