

FACULTY OF COMPUTER APPLICATIONS

Programme	Master of Computer Applications					Branch/Spec.	Computer Application		
Semester	I – Bridge Course					Version	1.0.0.0		
Effective from Academic Year			2026-27			Effective for the batch Admitted in		July 2026	
Subject Code	P11E8ALD		Subject Name			Algorithms and Logic Development			
Teaching scheme						Examination scheme (Marks)			
(Per week)	Lecture (DT)		Practical (Lab.)		Total		CE	SEE	Total
	L	TU	P	TW					
Credit	-	-	-	-	-	Theory	40	-	40
Hours	20	-	-	-	20	Practical	-	-	-

Objective:

To develop logical thinking and problem-solving skills through algorithmic design, flowcharting, pseudocode formulation, and foundational C programming techniques.

Pre-requisites:

Basic mathematical aptitude, analytical thinking, and fundamental computer operational skills.

Course Outcomes:

Name of CO	Description
CO1	Formulate clear algorithmic steps and visual flowcharts for problem analysis, design, and logic development.
CO2	Apply structured programming concepts using top-down/bottom-up design, pseudocode, and modular flow constructs.
CO3	Implement core C programming components, including data types, operators, and input/output control statements.
CO4	Identify, debug, and resolve common programming errors using trace tables, code execution logic, and systematic testing.

Mapping of CO and PO

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	2	-	-	-	2
CO2	3	2	2	1	-	-	-	2
CO3	3	2	2	2	-	-	-	2
CO4	3	3	2	2	-	-	-	3

Content:		
Unit	SECTION – I	Hrs
1	Introduction to Problem Solving & Program Design Problem Analysis: Concept of problem solving, defining inputs/outputs, structured program design. Development Lifecycle: Program execution flow, documentation techniques, writing clear internal comments. Debugging Basics: Introduction to program debugging, understanding syntax, runtime, and logical errors.	05
2	Fundamentals of Algorithms & Flowcharts Logic Formulation: Writing algorithms, properties of effective step-by-step logic. Flowcharting: Standard flowchart symbols, drawing logic paths, visual problem representation. Structured Logic: Writing pseudocode, applying top-down and bottom-up design methodologies.	05
SECTION – II		
3	Basics of C Programming & Language Structure C Basics: Structure of a C program, compilation, linking, and execution process. Tokens & Variables: Keywords, identifiers, literals, constants, data types (integer, float, char), and strings. Console Operations: Basic input and output statements	05
4	Operators, Control Structures & Decision Statements Operators: Arithmetic, relational, logical, and assignment operators. Decision Making: Selection statements (if, if-else, nested if, switch-case). Looping & Control: Iteration statements (while, for, do-while), jumping statements (break, continue).	05
Practical Content:		
List of programs specified by the subject teacher based on above mentioned topics		
Text/Reference Books:		
1.	Griffiths, D., & Griffiths, D. <i>Head First C: A Brain-Friendly Guide</i> , 1st Edition, O'Reilly Media	
2.	Pilone, D., & Miles, R. <i>Head First Software Development</i> , 1st Edition, O'Reilly Media.	
3.	Kanetkar, Y. <i>Let Us C</i> , 19th Edition, BPB Publications.	
4.	Garey, T. <i>Programming Logic and Design, Comprehensive</i> , 9th Edition, Cengage Learning.	
Web References /Online Learning Resources		
1.	RAPTOR (Flowchart Interpreter Tool): https://raptor.martincarlisle.com/	
2.	Raptor Flowchart Tutorial: https://programming-steps.blogspot.com/2013/10/raptor-flowchart.html	
3.	W3Schools (C Programming Tutorial): https://www.w3schools.in/c-tutorial/	

4.	TutorialsPoint (C Programming): https://www.tutorialspoint.com/cprogramming/index.htm