

FIRST SEMESTER EXAMINATION

Paper Code: <BCA-102> Programming in C

Maximum Marks: 40

Time: 2 Hours

COURSE OUTCOMES:

CO1: Understand the basics of the C programming language and its application.

CO2: Implement decision making, branching, and looping structures.

CO3: Manage large amounts of data using arrays and structures.

CO4: Discuss pointer-based solutions.

CO5: Develop recursive functions for solving problems.

CO6: Discuss file management techniques in C.

Note: Attempt all questions. Internal choice is indicated. (Total: 40 Marks)

SECTION A

Q. No.	Questions	Marks	Unit No.	CO Mapped	Bloom's Level
Q1	Answer all parts (Short Notes / Conceptual):	10	1	CO1	2
(a)	Differentiate between <code>#define</code> macro substitution and <code>const</code> variable declaration with suitable code examples.	2	1	CO1	2
(b)	Explain short-circuit evaluation in logical AND (<code>&&</code>) and logical OR (<code> </code>) operators with an illustrative example.	2	1	CO1	2
(c)	What is a dangling else problem? How is it resolved in nested <code>if-else</code> constructs using curly braces?	2	2	CO2	2
(d)	Differentiate between <code>break</code> and <code>continue</code> statements with control flow diagrams or code snippets.	2	2	CO2	2
(e)	What is a NULL pointer and a void pointer (generic pointer) in C? State their primary applications.	2	4	CO4	1

SECTION B

Q. No.	Questions	Marks	Unit No.	CO Mapped	Bloom's Level
Q2.	(a) Describe bitwise operators in C (<code>&</code> , <code> </code> , <code>^</code> , <code>~</code> , <code><<</code> , <code>>></code>) with truth tables and practical bit manipulation examples. (b) Write a C program to check whether a given integer is even or odd using only bitwise operators (without using modulus %).	10	1	CO1	3
OR					
	(a) Explain the fundamental data types and their size/range modifiers (<code>short</code> , <code>long</code> , <code>signed</code> , <code>unsigned</code>) in C. (b) Write a C program that converts a given decimal number into its equivalent binary representation.	10	1	CO1	3
Q3.	(a) Explain recursion with its base condition and recursive step. Compare recursion with iteration in terms of overhead and memory stack. (b) Write a recursive C program to generate the Fibonacci series up to N terms and compute the GCD of two numbers.	10	2	CO5	3
OR					
	(a) Write a C program to search for an element in an array using the Binary Search technique. State the precondition for Binary Search. (b) Explain the working of standard string handling functions: <code>strlen()</code> , <code>strcpy()</code> , <code>strcat()</code> , and <code>strcmp()</code> without built-in libraries.	10	3	CO3	3
Q4.	(a) What is Dynamic Memory Allocation? Differentiate between <code>malloc()</code> and <code>calloc()</code> . Explain the role of <code>realloc()</code> and memory leaks. (b) Write a C program to dynamically allocate memory for N integers, compute their sum and average, and deallocate memory using <code>free()</code> .	10	4	CO4	3