Integrated MCA I Semester 1

Subject Name: Problem Solving Using C **Subject Code:** IMCA13103

Type of course: Professional Core Course

Prerequisite (if any):
- Programming Language with C
- Mathematical Fundamentals

List of Courses where this course will be prerequisite:
- Object Oriented Programming
- Backend Programming
- Machine Learning

Rationale: Problem Solving Using C helps understanding the nature of the problem at a deeper level through logical thinking. This helps the students to achieve proficiency in writing efficient code while developing solutions to the problems in computer science.

Teaching and Examination Scheme:

TEACHING SCHEME				Theory Marks		Practical Marks		Total
L	T	P	C	TEE	CAT	TEP	CAP	
4	0	0	4	60	40	-	-	100

CAT: Continuous Assessment Theory comprised of CA1 and CA2 **CA1:** Continuous Assessment (assignments/projects/open book tests/closed book tests **CA2:** Sincerity in attending classes/class tests/ timely submissions of assignments/self-learning attitude/solving advanced problems **TEE:** Term End Examination **TEP:** Term End Practical Exam (Performance and viva on practical skills learned in course) **CAP:** Regular submission of Lab work/Quality of work submitted/Active participation in lab sessions/viva on practical skills learned in course

Content:

Sr. No.	Content	Teaching Hrs.	Module Weightage
1	**Introduction to programming:** Generation of languages, Introduction to Programming, Introduction to components of a computer system (disks, memory, processor, where a program is stored and executed, operating system, Compilers, Interpreters), Process of compilation and execution. Concepts of Algorithm and Flowcharts: steps to solve logical and numerical problems, Representation of Algorithm using Flowchart/Pseudo code with examples.	06	10%
2	**Basics of C:** Basic features of C Language, The C Character Set, Data Types and Sizes, Constants, Variables, Keywords, Types of C Variables, Rules for constructing Constants and Variables. Arithmetic Operators, Relational and Logical Operators, Type Conversions, Increment and Decrement Operators, Bitwise Operators, Assignment Operators and Expressions, Conditional Expressions, Precedence and Order of Evaluation (Hierarchy of operations and Associativity of operators).	06	10%
3	**Control Flow:** Statements and Blocks. Test Conditions, Conditional execution and selection using Flowchart, If-else, Else-if, Switch, Nested if. Iteration and Repetitive Executions using Flowchart, Loops – While and For, Loops – Do-While, Nested loops. Break and Continue, Goto and labels.	14	23%

4	**Arrays and Pointers:** Introduction to contiguous data types. One dimensional arrays, multidimensional arrays, Array as strings, multidimensional character arrays. Operations on strings. Basics of Pointers, Pointer Notation, Pointer Arithmetic, Pointer and Array, Pointer and String, Array of Pointers, Pointer as a function argument.	08	13%
5	**Functions:** Basics of Functions, What is a Function? Why use Functions? Passing values between Functions, Order of passing Arguments, Return type of Functions, Scope rules of Functions, External variables, Recursion.	06	10%
6	**User Defined Data Types:** Basics of Structures, usage of structure, declaring structure, accessing structure elements, nested structures, Arrays of Structures, Structures and Functions, Self-referential Structures, Typedef, Unions, Enumeration types, Bit-fields.	10	17%
7	**Data Types Revisited:** Integers – long and short, signed, unsigned, Chars – signed and unsigned, Floats and Doubles. Storage Classes – Automatic Storage Class, Register Storage Class, Static Storage Class, And External Storage Class.	04	7%
8	**The C Preprocessor:** Features of C Preprocessor, Macros with Arguments, Macros versus Functions, Conditional Compilation - #if and #elif Directives, Miscellaneous Directives - #undef and #pragma Directives.	06	10%

Suggested Specification table with Marks (Theory):

Distribution of Theory Marks					
R Level	**U Level**	**A Level**	**N Level**	**E Level**	**C Level**
20	20	15	15	15	15

Legends: R: Remembrance; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create and above Levels (Revised Bloom's Taxonomy)

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table.

Reference Books:

Sr. no.	Title of book /article	Author(s)	Publisher and details like ISBN	Year of publication	Publication Edition
1	Let Us C	Yashavant Kanetkar	BPB Publication ISBN: 9387284573	2016	15th Edition
2	The C Programming Language	Brian W. Kernighan and Dennis M. Ritchie	Prentice Hall of India ISBN: 9332549443	2015	2nd Edition
3	Programming in ANSI C	E. Balagurusamy	Tata McGraw Hill ISBN: 935316513X	2019	8th Edition
4	Outline of Programming with C	Byron Gottfried, Schaum	McGraw-Hill ISBN: 9789353160272	2018	4th Edition
5	AICTE Prescribed Textbook – Programming for Problem Solving (with Lab Manual)	R.S. Salaria	Khanna Book Publishing Co. ISBN: 939150521X	2022	1st Edition
6	C: The Complete Reference.	Herbert Schildt	Tata McGraw Hill ISBN: 9332585482	2017	4th Edition
7	As per the latest AICTE Syllabus Programming in C	Pradip Dey & Manas Ghosh	Oxford ISBN: 9780199491476	2018	-

Course Outcomes:

Sr. No.	CO Statement After learning this subject, students will be able to	Marks % weightage
CO-1	Learn fundamentals of computer and to analyze algorithms and to understand the various steps in program development.	10%
CO-2	Ability to understand fundamentals of C especially the syntax and semantics of C programming language.	10%
CO-3	Learn the usage of structured programming approach in solving problems.	23%
CO-4	Understand the contiguous and homogeneous data values using arrays, string manipulation and learn dynamic memory allocation and access using pointers.	13%
CO-5	Learn the modular programming approach for reusability.	10%
CO-6	Acquire the program readability and efficiency by organizing data values using structure, union concepts.	17%
CO-7	Determine visibility, lifetime and memory allocation and functions using storage classes.	7%
CO-8	Understand preprocessing before compilation using text substitution.	10%

Mapping with POs:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PO 13
CO-1	3	1	3	3	2	3	1	2	1	0	2	1	3
CO-2	3	2	3	3	1	3	1	2	0	0	2	1	3
CO-3	3	2	3	3	1	2	0	2	0	0	2	1	3
CO-4	3	1	3	3	3	2	0	2	0	0	2	1	3
CO-5	3	2	3	3	1	2	0	2	0	0	2	1	3
CO-6	3	1	3	3	1	2	0	2	0	0	2	1	3
CO-7	3	1	3	3	1	2	0	2	0	0	2	1	3
CO-8	3	1	3	3	1	2	0	2	0	0	2	1	3
Rationale*													

Rationale*: Explaining why it is matching this particular program outcome

List of Open learning website:

URL: https://onlinecourses.nptel.ac.in/
NPTEL Course on

1. Introduction to Programming in C by Prof. Satyadev Nandakumar (IIT Kanpur),
2. Problem Solving Through Programming in C by Prof. Anupam Basu (Iit Kharagpur))

List of Open Source Software:

- Inbuilt compiler GCC available on UBUNTU
- Clang
- TinyCC

Major Equipment Needed: NA