FACULTY OF ENGINEERING & TECHNOLOGY

Effective from Academic Batch: 2020-21

Programme: **Bachelor of Engineering (Information Technology)**

Semester: **I**

Course Code: **102000110**

Course Title: **Computer Programming with C**

Course Group: **Engineering Science Courses**

Course Objectives: Students will gain understanding of basics of computer, hardware, software, and programming language. Students will learn problem solving skills through C programming language.

Teaching & Examination Scheme:

Contact hours per week			Course Credits	Examination Marks (Maximum / Passing)				
				Theory		J/V/P*		Total
Lecture	Tutorial	Practical		Internal	External	Internal	External	
03	00	02	04	40 / 14	60 / 21	20 / 07	30 / 10	150 / 52

* **J**: Jury; **V**: Viva; **P**: Practical

Detailed Syllabus:

Sr.	Contents	Hours
1	**Introduction to Computers and Programming:** Introduction to computer: Basic block diagram, Functions of various components of computer, Concepts of Hardware and software, Types of software Computer languages and programming: Concepts of Machine level, Assembly level and high-level languages, Compiler and interpreter, Flowcharts and Algorithms	05
2	**Fundamentals of C:** Features of C language, structure of C Program, comments, header files, data types, constants and variables, operators, expressions, evaluation of expressions, type conversion, precedence and associativity, I/O functions	06
3	**Control structure in C:** Decision making and Branching: Simple if, if-Else, Nesting of if-else, Else If ladder, Switch statement, The ? operator, goto statement Decision making and Looping: while statement, do statement, for statement, Jumps in loop, break and continue, Nesting of control structures	08
4	**Array and String**: Concepts of array: One- and two-dimensional arrays, declaration and initialization, operation on array, multidimensional arrays Character array and string: declaration and initialization, operations on string, Built-in string functions, table of strings	07

5	**Functions and Recursion:** Concepts of user defined functions: function declaration, function definition, function call, passing parameters, nesting of functions, Introduction to Recursion as a way of solving problems and examples	**06**
6	**Structures and Unions:** Basics of structure, structure members, accessing structure members, nested structures, array of structures, structure and functions, Introduction to Unions	**04**
7	**Pointers and File Management:** Basics of pointers, pointer to pointer, pointer and array, pointer to array, array to pointer, function returning pointer, structures, and pointers, Introduction to file management and its functions	**04**
	TOTAL	**40**

List of Practicals / Tutorials:

1	• Write a program to understand concepts of structure of C Program, scanf and printf. • Write a program to declare, assign, read and print values of variables of different datatypes. • Write a program to that performs as calculator (addition, multiplication, division, subtraction).
2	• Write a program to understand concepts of other operators (bitwise, increment/decrement, conditional, etc.). • Write a program to find area of square, rectangle, triangle, and circle. • Write a program to calculate simple interest (i = (p*r*n)/100). Where i = Simple interest, p = Principal amount, r = Rate of interest, n = Number of years
3	• Write a program to enter a distance in to kilometer and convert it in to meter, feet, inches, and centimeter. • Write a program to compute Fahrenheit from centigrade (f=1.8*c +32). • Write a program to read a number and check it is even or odd.
4	• Write a program to find that the accepted number is Negative, or Positive or Zero. • Write a program to read three numbers from keyboard and find out maximum out of these three (Nested if else). • Write a program to check whether the entered character is capital, small letter, digit or any special character.
5	• Write a program to read marks from keyboard and your program should display equivalent grade according to following table (if else ladder); Marks — Grade 100 - 80 — Distinction 79 - 60 — First Class 59 - 40 — Second Class < 40 — Fail • Write a program demonstrate functionality of calculator using switch-case. • Write a program to find factorial of a given number.
6	• Write a program to reverse a number. • Write a program to generate first n number of Fibonacci series. • Write a program to find the sum and average of different numbers which are accepted by user as many as user wants. • Write a program to check whether the given number is prime or not.

7	• Write a program to evaluate the series 1^2+2^2+3^2+……+n^2
	• Write a program to find 1+1/2!+1/3!+1/4!+.....+1/n!.
	• Write a program to display following patterns using asterisk (*).
	`*` `* *` `* * *` `* * * *`
	`*` `* *` `* * *` `* * * *`
	`* * * *` `* * *` `* *` `*`
	• Write a C program to display following patterns.
	`1 2 3 4 5` `A A A A A` `1`
	`2 3 4 5` `B B B B` `0 1`
	`3 4 5` `C C C` `1 0 1`
	`4 5` `D D` `0 1 0 1`
	`5` `E` `1 0 1 0 1`

8	• Write a program to read array of integers and print it in reverse order. • Write a program that adds two 1-dimensional array & store into third array. • Write a program to insert and delete an element to/from desired position in an array. • Write a program to sort a given array in ascending order (Use Bubble Sort algorithm).
9	• Write a program for multiplication of two matrices. • Write a program to find length of string without using library function. • Write a program to concatenate two strings without using library function.
10	• Write a program that reads a string and counts occurrences of a given character. • Write a program convert character into Toggle character. • Write a program that checks whether the string is palindrome or not using string library function.
11	• Write a program to demonstrate the use of inbuilt string functions. • Write a function power that computes x raised to the power y for integer x and y and returns double type value. • Write a calculator program (add, subtract, multiply, divide). Prepare user defined function for each functionality.
12	• Write a program to find sum of elements of 1-D Array using Function. • Write a program that use user defined function swap() to interchange the value of two variable. • Write a program to find factorial of a number using recursion. • Write a program to generate Fibonacci series using recursion.
13	• Write a function which takes a two integer array as argument and give sum of these arrays. • Define a structure to enter enrolment number, name of student and marks of the student in three subjects. Enter data for 5 students. Display grade cards of all students. Display student who has top rank in the class. • Define a structure called cricket that will describe the following information: Player name, Team name, Batting average Declare an array player. Write a program to print name & team of those players whose batting average is greater than given value.
14	• Write a program to demonstrate the concept of union. • Write a program using pointer and function to determine the length of string. • Write a program to demonstrate the concept of pointer. • Write a program to add elements of array using pointer.

15	• Write a program to copy the content one file into another file. • Write a program to demonstrate ftell() and fseek() for file handling. • Write a program that compares two files and returns 0 if they are equal and 1 if they are not.

Reference Books:

1	Programming in ANSI C, Eighth Edition by E. Balagurusamy, McGraw Hill Education
2	Let us C, by Yashavant Kanetkar, BPB Publications
3	Fundamentals of Computing and Programming in C, by Pradip Dey, Manas Ghosh, Oxford University Press
4	How to Solve it by Computer, by R. G. Dromey, Pearson Education

Supplementary learning Material:

1	NPTEL course / tutorials
2	Vlabs.iitb.ac.in
3	Open online courses from www.coursera.org, www.udacity.com, etc.

Pedagogy:

- Direct classroom teaching
- Assignments/Quiz
- Continuous assessment
- Seminar/Poster Presentation
- Course Projects

Suggested Specification table with Marks (Theory) (Revised Bloom's Taxonomy):

Distribution of Theory Marks in %						**R**: Remembering; **U**: Understanding; **A**: Applying; **N**: Analyzing; **E**: Evaluating; **C**: Creating
R	**U**	**A**	**N**	**E**	**C**	
20%	30%	30%	20%	-	-	

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.

Course Outcomes (CO):

Sr.	Course Outcome Statements	%weightage
CO-1	Formulate algorithm and/or flowchart for a given problem.	**10**
CO-2	Translate algorithm and/or flowchart into C program using correct syntax and execute it.	**10**
CO-3	Write programs using control structures, arrays, functions, structures.	**40**
CO-4	Decompose a problem and formulate solutions using functions.	**20**
CO-5	Apply concepts of array, pointer, structure, functions, recursion and file management to solve engineering and/or scientific problems.	**20**

Curriculum Revision:	
Version:	1.0
Drafted on (Month-Year):	June-2020
Last Reviewed on (Month-Year):	-
Next Review on (Month-Year):	June-2025