COURSE STRUCTURE

Course Code	**CET1041B**			
Course Category	**Engineering Science**			
Course Title	**Programming Foundations**			
Total Teaching Hrs and Credits	**Lectures**	**Tutorial**	**Laboratory**	**Credits**
	2hrs/wk	**--**	**4hrs/wk**	**2+2=4**

Pre-requisites:

- Introductory Knowledge of Computers.

Course Objectives:

1. Knowledge:

1. Learn the programming skills and programming language construct.

2. Skills:

1. Understand the functions,arrays and structures using C language.
2. Understand file handling and pointers using C language.

3.Attitude:

1. Learn to apply programming skills for solving real world problems.

Course Outcomes:

After completion of the course the students will be able to: -

1. Develop efficient logic and algorithms for solving a problem.
2. Analyze the given problem and solve it using suitable programming constructs.
3. Apply programming skills for solving real world problems.

Course Contents:

1. Introduction of Computer System and Problem Solving
2. Fundamentals of C
3. Data types and Functions in C
4. Pointers and File handling in C
5. Fundamentals of Programming Language

Laboratory Exercises:

1. C Data Types
2. C Control Structures
3. C Functions
4. Arrays and Structures in C
5. Pointers

6. File Handling

Learning Resources:

Reference Books:

1. Pradeep Sinha, Priti Sinha, “Computer Fundamentals”, Sixth edition, bpb publication.
2. Ramon Mata-Toledo, Pauline K. Cushman, “Introduction to Computer Science”, Schaum’s Outline series.
3. Herbert Schildt, "C: The Complete Reference", Fourth Edition, McGraw Hill Professional.
4. Yashwant Kanetkar, “Let us C”, Fifteenth edition, bpb publication.

Web Resources:

Web Links:

1. http://www.studytonight.com/c/overview-of-c.php
2. https://www.tutorialspoint.com/cprogramming

MOOCs:

1. http://nptel.ac.in/courses/106105085/2
2. http://nptel.ac.in/courses/106104074/1
3. https://nptel.ac.in/courses/106/105/106105171
4. https://nptel.ac.in/courses/106/106/106106212/

Pedagogy:

1. Power point presentations
2. Practical Demos
3. Videos
4. Online Classroom
5. Expert Lectures

Assessment Scheme:

Class Continuous Assessment (CCA): 60 Marks

Assignments	Presentations	Mid Term Exam
20 Marks	20 Marks	20 Marks

Laboratory Continuous Assessment (LCA): 50 Marks

Practical	Orals
30 Marks	20 Marks

Term End Examination: 40 Marks

Syllabus: Theory

Module No.	Contents	Workload in Hrs
		Theory
1	**Introduction of Computer System and Problem Solving:** Basics of Computers: Architecture, Processors, Memory, Number Systems, Data Representation- Floating point, Char, String. System Software - Operating system, Editor, Compiler, Assembler, Linker, Loader. Introduction to Problem Solving: Problem solving process/framework, Programming Paradigms: Imperative, Object Oriented, Functional and Logic programming. Characteristics of Programming Languages, Role of programming languages, need of studying programming languages. Programming Design Tools: Algorithms, Pseudo-code and Flowchart, Case studies for Algorithm, Flowchart and Pseudocode. Top-Down and Bottom-Up design approach. Software Development Life Cycle.	07
2	**Fundamentals of C** **Introduction to C:** Fundamentals of C-Programming, Data types, Constants, Variables, Operators, Expression, Pre-processor directives. Data Input and Output. **Control Structures:** Decision control statements, Selection/conditional branching Statements: if, if-else, nested if, if-elif-else statements. Basic	07

	loop Structures/Iterative statements: while loop, for loop, selecting appropriate loop. Nested loops, the break, continue, pass, else statement used with loops Structure of C program, Coding conventions	
3	**Derived data types and Functions in C** **Derived data types:** Array- Single and Multidimensional arrays, Structure – Structure and Array of structure, Union. Strings **Functions in C:** User defined and Library Functions-String Library Functions. Different parameter passing methods (Call by Value and Call by Reference), Passing array to a function, Recursion.	08
4	**Pointers and File handling in C** **Pointers:** Lifetime of Variables, Scope Rules: Static and Dynamic scope. Pointers, Passing Pointers to function, Pointers and Arrays, Dynamic memory allocation and its application. **File Handling in C:** File, Types of Files, File operations. **Fundamentals of Programming Language:** Introduction: Characteristics of Programming Languages, Influencing Factors for the Evolution of Programming Language, Desirable Features and Design Issues. Brief Introduction to Programming Language Paradigms: Imperative, Object Oriented, Functional, Logic and Concurrent Programming Syntactic Structure: Syntax, Semantics, Structure, Character Set Tokens, Sentence-Syntax and Semantics, Expression Notation, Grammar, Syntax Tree, Context Free Grammar, Translators	08

Laboratory:

Assignment No.	Contents	Workload in Hrs
		Lab
1	Write a program in C to check leap year. Write a menu driven program in C to implement the basic arithmetic operations.	04

	Write a program in C to generate multiplication tables.	
2	Write a C Program to calculate salary of an employee given his basic pay (take as input from user). Calculate gross salary of employee. Let HRA be 10 % of basic pay and TA be 5% of basic pay. Let employee pay professional tax as 2% of total salary. Calculate net salary payable after deductions.	02
3	Write a program in C to perform basic operation such as addition, saddle point, inverse, magic square of two matrices.	04
4	Write a C function to compute the factorial of a number with and without recursion.	02
5	Write a C program to accept student details and display their result using array of structures.	02
6	To accept student's five courses marks and compute his/her result. Student is passing if he/she scores marks equal to and above 40 in each course. If student scores aggregate greater than 75%, then the grade is distinction. If aggregate is 60>= and <75 then the grade if first division. If aggregate is 50>= and <60, then the grade is second division. If aggregate is 40>= and <50, then the grade is third division.	02
7	To check whether input number is Armstrong number or not. An Armstrong number is an integer with three digits such that the sum of the cubes of its digits is equal to the number itself. Ex. 371.	02
8	To simulate simple calculator that performs basic tasks such as addition, subtraction, multiplication and division with special operations like computing xy and x!	04
9	To accept the number and Compute a) square root of number, b) Square of number, c) Cube of number d) check for prime, d) factorial of number e) prime factors	04
10	To accept two numbers from user and compute smallest divisor and Greatest Common Divisor of these two numbers.	04
11	To accept a number from user and print digits of number in a reverse order.	02
12	To input binary number from user and convert it into decimal number.	02
13	To generate pseudo random numbers	04

14	To accept list of N integers and partition list into two sub lists even and odd numbers.	02
15	To accept the number of terms a finds the sum of sine series.	04
16	Write a C program that accepts a string from user and perform following string operations- i. Calculate length of string ii. String reversal iii. Equality check of two strings iii. Check palindrome ii. Check substring	04
17	Create Structure EMPLOYEE for storing details (Name, Designation, gender, Date of Joining and Salary). Define function members to compute a)total number of employees in an organization b) count of male and female employee c) Employee with salary more than 10,000 d) Employee with designation “Asst Manager”	04
18	Write a C function to swap two numbers with and without pointers.	04
19	Write a C program to copy contents of one file to another using File handling.	02
20	Write a menu driven program in C to perform all string operations. (In built functions).	02