Course Name: **Learn Programming Fundamental with C**

Course Code: GE4B-07

Mode- Offline/ Blended **Credits: 6**

Career Objective: Programming is an increasingly important skill, whether you aspire to a career in software development, or in other fields. This course is the first in the specialization Introduction to Programming in C. Programming is fundamentally about figuring out how to solve a class of problems and writing the algorithm, a clear set of steps to solve any problem in its class. This course will introduce you to a powerful problem-solving process—the Seven Steps—which you can use to solve any programming problem. In this course, you will learn how to develop an algorithm, and then progress to reading code and understanding how programming concepts relate to algorithms.
The C language is particularly well suited as an introduction to coding: It's a tried-and-true language, and it allows understanding computing processes at a deep level.

Sl	Course Outcome	Mapped modules
CO1	Understanding program, programming and its requirements	M1
CO2	Understanding Algorithm	M2
CO3	Understanding Basic Data Type and Type conversion	M3
CO4	Understanding c programming fundamental, compiling Debugging, Running program	M4
CO5	Understanding Data Types flow of control	M5
CO6	Understanding Advance function recursion, array , pointer	M6

Detailed Syllabus:

Module	Content	Total Hours	%ageof questions	Blooms Level (if applicable)	Remarks (Ifany)
Module 1	will learn how to approach a programming problem methodically. This module discuss about to execute a piece of code by hand	11	10	2	
Module2	Discussion about the basic data types, "non-number" types, and complex, custom types	11	10	2	
Module 3	History of CCompiling, debugging, and running a program with different examples	11	20	2,3	

Module 4	Logical operators, expressions, and short-circuit evaluationThe conditional statement if and if-elseThe iterative statement	11	10	2	
Module 5	Enums as an ADTEnums codeThe C preprocessorUse assert for program correctnessAssert code Introduction to struc	8	25	2,3	
Module 6	Intro to the ADT listList of one element codeFull list codeDetails of list processingHonors: Introduction to binary trees	8	25	1,2,3	
		60	100		

Module 1-Discuss about a powerful process for solving any programming problem—the Seven Steps. You will learn how to approach a programming problem methodically, so you can formulate an algorithm that is specific and correct. This module discuss about to execute a piece of code by hand, and clearly illustrate what each statement does and what the state of the program is.
Module 2- This module discuss about types beyond integers, both their conceptual representations, and their hardware representations in binary. Discussion about the basic data types, "non-number" types, and complex, custom types
Module 3-History of CCompiling, debugging, and running a program, Compiling, debugging, and running a program Example - Circle codeExample – MarathonSimple input/output –Fahrenheit Simple input/output – milesCharacter sets and tokensCommentsKeywordsIdentifiersOperatorsExpressions and precedence Expression and evaluationDeclarationsFundamental types and size ofThe char type
Module 4-Logical operators, expressions, and short-circuit evaluationThe conditional statement if and if-elseThe iterative statement whilewhile-cnt-char-explained, while-code – exampleThe for statement and its while analogoddball operators-conditional and commaternary-operator code exampleBreak and continue and switch Function definition, Return statementFunction prototype, Function variables—with call-by-value explained, Function definitions and scope rulesSimple recursion, Recursion- factorial codeRecursion Fibonacci code, Pointers and simple arrays, initialize arraysWhat is a pointerCall-by-reference simulatedarray as a parameterarray-bubble-sort codemerge sort overview
Module 5-Discussion about:- Enums as an ADTEnums codeThe C preprocessorPreprocessor codeUse assert for program correctnessAssert code Introduction to struc (More advanced ADTs)How to access struct membersIntroduction to the ADT stackUsing a stack to reverse a string
Module 6-Discussion about:- Intro to the ADT listList of one element codeFull list codeDetails of list processingHonors: Introduction to binary trees Honors: Detailed binary tree codeIntroduction to File I/OBasic File/I/O codeDouble Spacing a FileUse of Main (argc, argv) Honors - List Code with deletion

List of experiments:

1) Understanding program, programming and its requirements
2) Program to display different data types and their type conversion
3) Understand different kinds of algorithm for different programs.
4) Program to Understanding Data Types flow of control
5) Program to Understanding Advance function recursion, array , pointer

Suggested Reading:

1) Let Us C by Yashavant Kanetkar
2) "The C Programming Language" by Brian W Kernighan / Dennis Ritchie