Course Title:	**Principles of Programming using C**	Semester	I /II
BPLCK105D/BPLCK205D	**BPOPS103/203**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Integrated	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P: S)	2:0:2:0	Exam Hours	03
Total Hours of Pedagogy	40 hours	Credits	03

Course Learning Objectives

CLO 1. Elucidate the basic architecture and functionalities of a computer
CLO 2. Apply programming constructs of C language to solve the real-worldproblems
CLO 3. Explore user-defined data structures like arrays, structures and pointers in implementing solutions to problems
CLO 4. Design and Develop Solutions to problems using structured programmingconstructs such as functions and procedures

Teaching-Learning Process

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. AskatleastthreeHOT(HigherorderThinking)questionsintheclass,whichpromotescri ticalthinking.
5. Adopt Problem Based Learning(PBL),which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Showthedifferentwaystosolvethesameproblemandencouragethestudentstocome up with their own creative ways to solve them.
8. Discusshoweveryconceptcanbeappliedtotherealworld-andwhenthat'spossible,ithelps to improve the students' understanding.
9. Use https://pythontutor.com/visualize.html#mode=edit in order to visualize theoperations of C Programs

Module-1: (8 hours)

Introduction to C: Introduction to computers, input and output devices, designing efficient programs. Introduction to C, Structure of C program, Files used in a C program, Compilers, Compiling and executing C programs, variables, constants, Input/output statements in C,

Textbook: Chapter 1.1-1.9, 2.1-2.2, 8.1 - 8.6 ,9.1-9.14
Applications: Elementary for writing C programs

(RBT Levels: L1, L2 and L3)

Module-2: (8 hours)

Operators in C, Type conversion and typecasting.

Decision control and Looping statements: Introduction to decision control, Conditional branching statements, iterative statements, nested loops, break and continue statements, goto statement.

Textbook: Chapter 9.15-9.16, 10.1-10.6

Applications: Writing programs having loops and decision making
(RBT Levels: L1, L2 and L3)

Module-3 : (8 hours)

Functions: Introduction using functions, Function definition, function declaration, function call, return statement, passing parameters to functions, scope of variables, storage classes, recursive functions.
Arrays: Declaration of arrays, accessing the elements of an array, storing values in arrays, Operations onarrays, Passing arrays to functions, two dimensional arrays, operations on two-dimensional arrays, two- dimensional arrays to functions, multidimensional arrays, applications of arrays.

Textbook: Chapter 11.1-11.10, 12.
1-12.10,12.12Applications: Use of arrays and modularity in writing Programs
(RBT Levels: L1, L2 and L3)

Module-4: (8 hours)

Strings and Pointers: Introduction, string taxonomy, operations on strings, Miscellaneous string and character functions, arrays of strings. Pointers: Introduction to pointers, declaring pointer variables, Types of pointers, Passing arguments to functions using pointers

Textbook: Chapter 13.1-13.6, 14-14.7

Applications: How to use Pointers and Strings
(RBT Levels: L1, L2 and L3)

Module-5: (8 hours)

Structure, Union, and Enumerated Data Type: Introduction, structures and functions, Unions, unionsinside structures, Enumerated data type.
Files: Introduction to files, using files in C, reading and writing data files. , Detecting end of file

Textbook: Chapter 15.1 – 15.10, 16.1-16.5

Applications: How to use Structures and File handling
(RBT Levels: L1, L2 and L3)

Course outcome

At the end of the course the student will be able to:

At the end of the course the student will be able to:

CO1: Elucidate the basic architecture and functionalities of a computer and also recognizethe hardware parts.

CO 2: Apply programming constructs of C language to solve the real world problem

CO 3: Explore user-defined data structures like arrays in implementing solutions to problems like searching and sorting

CO 4: Explore user-defined data structures like structures, unions and pointers in implementing solutions.

CO5: Design and Develop Solutions to problems using modular programming constructs using functions.

Course Assessment and Evaluation Details (both CIE and SEE)

Continuous Internal Evaluation: 50 marks		
Theory Assessment Tool	**Marks**	**Reduced marks**
IAT-1	25	25
IAT-2	25	
Assessment -1(activity based)	25	25
Assessment-2(activity based)	25	
Semester End Examination (SEE) : 50 marks		
SEE	**Marks**	**Reduced marks**
Course end examination (Answer any one question from each unit – Internal choice)	100	50

Activity Based Learning / Practical Based learning

- Assign small tasks to Develop and demonstrate using C

Programming Tasks:

Programming Assignments

1 Simulation of a Simple Calculator.

2 Compute the roots of a quadratic equation by accepting the coefficients. Print appropriate messages.

3 An electricity board charges the following rates for the use of electricity: for the first 200 units 80 paise per unit:for the next 100 units 90 paise per unit: beyond 300 units Rs 1 per unit. All users are charged a minimum of Rs. 100 as meter charge. If the total amount is more than Rs 400, then an additional surcharge of 15% of total amount is charged. Write a program to read the name of the user, number of units consumed and print out the charges.

4. Write a C Program to display the following by reading the number of rows as input,

```
                1
              1   2   1
            1 2   3   2 1
          1 2 3   4   3 2   1
```

---

n^{th} row

5 Implement Binary Search on Integers.

6 Implement Matrix multiplication and validate the rules of multiplication.

7 Compute sin(x)/cos(x) using Taylor series approximation. Compare your result with the built-in library function.Print both the results with appropriate inferences.

8 Sort the given set of N numbers using Bubble sort.

9 Write functions to implement string operations such as compare, concatenate, and find string length. Use theparameter passing techniques.

10 Implement structures to read, write and compute average- marks of the students, list the students scoring aboveand below the average marks for a class of N students.

11 Develop a program using pointers to compute the sum, mean and standard deviation of all elements stored inan array of N real numbers.

12. Write a C program to copy a text file to another, read both the input file name and target file name.

Suggested Learning Resources:

Text Books

1. Computer fundamentals and programming in c, "Reema Thareja", Oxford University, Second edition,
2017.

Reference Books:

1. E. Balaguruswamy, Programming in ANSI C, 7th Edition, Tata McGraw-Hill.

2. Brian W. Kernighan and Dennis M. Ritchie, The 'C' Programming Language, Prentice Hall of India.

Web links and Video Lectures (e-Resources):

1. elearning.vtu.ac.in/E-content/courses/video/BS/15PCD23.html

2. https://nptel.ac.in/courses/106/105/106105171/ MOOC courses can be adopted for more clarity in understanding the topics and verities of problem-solving methods.

3. https://tinyurl.com/4xmrexre

COs and POs Mapping (CO-PO mappings are only Indicative)

COs	POs											
	1	2	3	4	5	6	7	8	9	10	11	12
CO1												
CO2												
CO3												
CO4												
CO5												

Level 3- Highly Mapped, Level 2-Moderately Mapped, Level 1-Low Mapped, Level 0- Not Mapped