CSBT100	Programming for Problem Solving	2L: 0 T: 0P	2 Credits
---------	---------------------------------	-------------	-----------

### **Course Objectives:**

- 1. To learn the fundamentals of computers.
- 2. To understand the various steps in program development.
- 3. To learn the syntax and semantics of C programming language.
- 4. To learn the usage of structured programming approach in solving problems.
- 5. To understated and formulate algorithm for programming script
- 6. To analyze the output based on the given input variables

### **Course Contents:**

**Module I:** Introduction to Programming; Introduction to components of a computer system (disks, memory, processor, where a program is stored and executed, operating system, compilers etc.) Idea of Algorithm: steps to solve logical and numerical problems. Representation of Algorithm: Flowchart/Pseudocode with examples.

From algorithms to programs; source code, variables (with data types) variables and memory locations, Syntax and Logical Errors in compilation, object and executable code.

Module II: Arithmetic expressions and precedence.

Module III: Conditional Branching and Loops. Writing and evaluation of conditionals and consequent

branching. Iteration and loops.

Module IV: Arrays, Arrays (1-D, 2-D), Character arrays and Strings

**Module V:** Basic Algorithms, Searching, Basic Sorting Algorithms (Bubble, Insertion and Selection), Finding roots of equations, notion of order of complexity through example programs (no formal definition required)

**Module VI**: Function, Functions (including using built in libraries), Parameter passing in functions, call by value, Passing arrays to functions: idea of call by reference

**Module VII:** Recursion, Recursion as a different way of solving problems. Example programs, such as Finding Factorial, Fibonacci series, Ackerman function etc. Quick sort or Merge sort.

Module VIII: Structures, Defining structures and Array of Structures

**Module IX:** Pointers, Idea of pointers, Defining pointers, Use of Pointers in self-referential structures, notion of linked list (no implementation)

Module X: File handling (only if time is available, otherwise should be done as part of the lab).

## **PRACTICALS:**

- 1. Familiarization with programming environment
- 2. Simple computational problems using arithmetic expressions
- 3. Problems involving if-then-else structures
- 4. Iterative problems e.g., sum of series
- 5. 1D Array manipulation
- 6. Matrix problems, String operations
- 7. Simple functions
- 8. Programming for solving Numerical methods problems
- 9. Recursive functions
- 10. Pointers and structures
- 11. File operations

### **TEXT/REFERENCE BOOKS:**

- 1. AICTE's Prescribed Textbook: Programming for Problem Solving, Khanna Book Publishing Co.
- 2. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill.
- 3. E. Balaguruswamy, Programming in ANSIC, Tata McGraw-Hill.
- 4. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India.

# Alternative NPTEL/SWAYAM Course:

S. No.	NPTEL Course Name	Instructor	Host Institute
1	INTRODUCTION TO	PROF. SATYADEV	IITK
-	PROGRAMMING IN C	NANDAKUMAR	
2	PROBLEM SOLVING THROUGH	PROF. ANUPAM	IIT KGP
	PROGRAMMING IN C	BASU	

S. No.	Experiment Name	Experiment Link(s)
1	Simple computational problems using arithmetic expressions.	http://ps- iiith.vlabs.ac.in/exp7/Introduction.html? do main=Computer%20Science&lab=Probl e m%20Solving%20Lab
2	Iterative problems e.g., sum of series.	http://ps- iiith.vlabs.ac.in/exp4/Introduction.html? do
		<u>main=Computer%20Science&amp;lab=Probl</u> <u>e m%20Solving%20Lab</u>
3	1D Array manipulation.	http://cse02- iiith.vlabs.ac.in/exp4/index.html
4	Matrix problems, String operations.	http://ps- iiith.vlabs.ac.in/exp5/Introduction.html? do main=Computer%20Science&lab=Probl e m%20Solving%20Lab
5	Simple functions.	http://cse02- iiith.vlabs.ac.in/exp2/index.html
6	Programming for solving Numerical methods problems.	http://ps- iiith.vlabs.ac.in/exp1/Introduction.html? do main=Computer%20Science&lab=Probl e m%20Solving%20Lab
7	Recursive functions.	http://ps- iiith.vlabs.ac.in/exp6/Introduction.html? do main=Computer%20Science&lab=Probl e m%20Solving%20Lab

# EXPERIMENTS THAT MAY BE PERFORMED THROUGH VIRTUAL LABS:

**COURSE OUTCOMES:** The student will learn following through lectures:

- To formulate simple algorithms for arithmetic and logical problems.
- To translate the algorithms to programs (in C language).
- To test and execute the programs and correct syntax and logical errors.
- To implement conditional branching, iteration and recursion.
- To decompose a problem into functions and synthesize a complete program using divide and conquer approach.
- To use arrays, pointers and structures to formulate algorithms and programs.
- To apply programming to solve matrix addition and multiplication problems and searching and sorting problems.
- To apply programming to solve simple numerical method problems, namely rot finding of function, differentiation of function and simple integration.

The student will learn following through Practicals:

- To formulate the algorithms for simple problems.
- To translate given algorithms to a working and correct program.
- To be able to correct syntax errors as reported by the compilers.
- To be able to identify and correct logical errors encountered at run time.
- To be able to write iterative as well as recursive programs.
- To be able to represent data in arrays, strings and structures and manipulate them through a program.
- To be able to declare pointers of different types and use them in defining self-referential structures.
- To be able to create, read and write to and from simple text files.

\*\*\*\*