Course Syllabi: UTA003 Computer Programming (L : T : P :: 3 : 0 : 2)

1. **Course number and name:** UTA003; Computer Programming

2. **Credits and contact hours:** Credits: 4.0; Hours: 5

3. **Text book, title, author, and year**

   - *KernighanBrian W. and Ritchie, Dennis M, The C Programming language, Dorling Kingsley (2008) 2nd ed.*
   - *Balagurusamy, E., Programming in ANSI C, TMH Publications (2007) 3rded.*
   - *Stroustrup, Bjarne, The C++ Programming Language,. Addison Wesley (2000) 3rded.*
   - *Kanetkar, Yashavant, Let Us C, BPB 7th Ed. (2006) 8thed.*

   a. Other supplemental materials
      - Nil

4. **Specific course information**
   a. Brief description of the content of the course (catalog description)

Introduction: Elements of computer processing, Hardware and software, Introduction and feature wise comparison of various Operating Systems, Including DOS, Windows and Linux, Problem solving-Algorithms and Flowcharts.

C Programming Basics: Basic program construction, Structure of a C program, Compilation process. Various compilers available on different OS/ environments including Turbo C, Borland C, GC, GCC, MSVC. Console I/O (printf, scanf), Preprocessor directives, Comments, Data types, Type conversions, Operators - Arithmetic, Relational, Logical, Conditional, Increment/decrement, Library functions, Header files.

Loops and Decision Statements: for loop, *while* loop, *do* loop, Various forms of *if* statement, *switch* statement, *break* statement, *continue* statement, *goto* statement, arrays and strings, Declaring an array, Initializing arrays, Accessing the array elements, Working with multidimensional arrays, Declaring and initializing string variables, Arithmetic operations on characters, String handling functions (string.h), Pointers, Pointers to pointers, Declaring and initializing pointers, Pointer expressions, Pointer increment and scale factor, Pointers and arrays, Pointers and strings.

Functions: Defining functions, passing arguments to functions, returning values from functions, Reference arguments, Variables and storage classes, Static functions, Pointers and functions.

Structures and Union: Declaring and initializing a structure, Accessing the members of a structure, Nested structures, Array of structures, Using structures in functions, Pointers and structures, Declaring and initializing a union.

Files: Reading and writing to text and binary files, Character I/O, String I/O, File pointers, Error handling, Redirection, Command line arguments.

Structured Programming vs. Object Oriented Programming.

Laboratory work: Introduction to Hardware - CPU, Storage devices & media, VDU, I/O Devices. Basic Operating System (DOS/UNIX) commands. Simple programs to demonstrate the use of constants, Variables, printf, scanf and operators. Programs using Loops: Solution of quadratic equation, Summation of finite series, Fibonacci series, Prime numbers, Factorial. Menu driven programs using switch statement. Use of continue and break statements, Conditional operators. Passing variables to functions by values and by reference, Number conversion using array, Sorting, Merging, Arithmetic operations on matrices. String manipulation: Comparing, Copying, Reversing, Findinglength, Extracting characters. Simple programs demonstrating the concept of Pointers, Passing values to functions using pointers for arrays, Structures. Creating various types of records using structures. Storing and retrieving records from a file, copying a data file. Randomly accessing a record, Use of command line arguments.

5. **Specific goals for the course**

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

- Explain the hardware and software concepts of computer system.
- Explain the C programming basics.
- Elaborate the concept of Functions, Structures and Union.
- Differentiate between Structured programming Vs Object oriented programming.

6. **Brief list of topics to be covered**

- C Programming Basics
- Loops and Decision Statements
- Functions
- Structures and Union
- Files