Rajarshi Janak University
Faculty of Science and Technology
Course of Study for B.Sc. CSIT
(First Semester/First Year)

Course Title: Computer Concept and Programming
Nature of Course: Theory (3 Hr)+Practical (3 Hr)
Credit hrs.: 3

Course Code: CSIT 104
Full Mark: Ext(60+20)+Int(40)
Pass Mark: Ext(16+8)+Int(16)

Course Objectives:
This course will enable students to:

- Familiarize with writing of algorithms, fundamentals of C and philosophy of problem solving.
- Implement different programming constructs and decomposition of problems into functions.
- Use and implement data structures like arrays and structures to obtain solutions.
- Define and use of pointers with simple applications.
- Define different modes of graphics with simple application

Course Contents

Unit-1
Computer Fundamental and Programming Methodology [3 hrs.]
Introduction, Components of PC, Computer Architecture, Memory Types, Memory Hierarchy, Computer Peripherals, Input and Output Devices, Basic of Computer Networking, Computer Program, Steps for Program Development, Problem Solving Tools: Algorithmic Thinking and Flowchart, Pseudocode, Program Control Structures, Program Methodology, Program Models.

Unit-2
Overview of C Language [4 hrs.]
Introduction, C Character Set, Tokens, Identifiers, Keywords, Constants, Variables, Data Types, Type Conversion, Operators and Expressions, Structure of a C program, Managing Input and Output Operations, Common Errors in Programming, Debugging Basics.

Unit-3
Control Structures [6 hrs.]
Introduction, Decision Making Statements, Looping Statements, Branching Statements, Common Pitfalls: Infinite loops, Misplacement of Conditionals.

Unit-4
Arrays and Strings [6 hrs.]
Introduction, Types of Arrays, One-Dimensional Array, Multidimensional Arrays, String as Array of Characters, String Handling Functions

Unit-5

Functions **[8 hrs.]**

Introduction, User Defined Functions, Return Statement, Function Call, Types of Functions based on their Return Type and Function Call, Inline Functions, Recursions, Arrays and Functions, Preprocessor Directives and Standard Library Functions

Unit-6

Pointers **[5 hrs.]**

Introduction, Pointer Declaration, Initializing Pointers, Arithmetic Operations with Pointer, Pointers and Arrays, Pointers and Strings, Pointers and Functions, Pointer to a Pointer, Pointer to Void, Dynamic Memory Allocation

Unit-7

Structures and Unions **[5 hrs.]**

Introduction, Structure Declaration and Accessing Structure Elements, Initialization of a Structure, Array and Structures, Nested Structures, Structure and Function, Unions, Pointer to Structures and Unions

Unit-8

File Handling in C **[5 hrs.]**

Introduction, Basic Terminology Associated with Files, Types of Files, Streams and Files, Binary vs Text File ad File Buffering, File System Structures, Various Types of File Access Methods, Input and Output Operations on Files and Standard Devices, File Operations, Error Handling in Files, Command Line Arguments

Unit-9

Introduction to Graphics **[3 hrs.]**

Modes, Initialization, Graphics Function, drawing shapes, lines, working with Colours

Laboratory Work:

The Laboratory work must cover programming part of all the topics covered in the course. The instructor can conduct the programming as required. Some important contents that should be included in lab exercises are as follows:

1. Create algorithms and flowchart for solving the problem.
2. Create, compile, debug, run and test simple C programs
3. Create decision making programs using control statements like; if, if..else, if..else ladder, nested if, and switch cases.
4. Create programs using loops (for, while, do while, nested loops) and realize the differences between entry controlled and exit controlled loops.
5. Create, manipulate arrays and matrices (single and multi-dimensional), work with pointers, dynamically allocate/de-allocate storage space during runtime, manipulate strings (character arrays) using various string handling functions.
6. Create user-defined functions with/without parameters or return type, create recursive functions, use function call by value and call by address, work with automatic, global and static variables.

7. Create programs that addresses pointer arithmetic, pointers and arrays, pointer and character strings, pointers and functions, pointer and structure, and dynamic memory allocation.
8. Create and use simple structures, array of structures, nested structure. Passing structure and array of structure to function, concept of pointer to structure
9. Create files that address random access and input/output operations in file, create files to keep records and manipulation of records etc.
10. Create graphics program that address some basic functions of *graphics.h* header file, e.g. line(), arc(), circle(), rectangle() etc.

Mini project can be assigned by Course Instructor covering the content from the syllabus taught.

Text Books/Reference Books

- Balagurusamy, E. (2019), **Programming in ANSI C (8th ed.),** New Delhi, India: Tata McGraw-Hill.
- Bryon S Gottfried (2018), **Programming with C (4th ed.),** Mc Graw Hill India.
- PK Sinha, **Computer Fundamentals (8th ed.)**, BPB Publications, India
- Dennis M. Ritchie, Brian W. Kernighan, **The C Programming Language (2nd ed.)**, Prentice Hall.
- Kanetkar, Y. P. (2022), **Let us C (15th Ed.)**, New Delhi, BPB Publication