GUJARAT TECHNOLOGICALUNIVERSITY

MASTERS IN COMPUTERAPPLICATION

Year –1(Semester–II) (W.E.F. Dec 2017)

Subject Name: Fundamentals of Programming - II
Subject Code: 3620001

1. Objectives:

1. To be able to understand and use pointers in C programs.
2. To be able to create user defined data types in C
3. To be able to write C application which can do input/output on files.

2. Prerequisites: Basic knowledge of C programming

3. Course Contents:

Sr. No.	Course Content	No. of Lectures
1	**Unit 1: Pointers** Introduction, Understanding Memory Addresses, Address Operator (&), Pointers (Declaring a Pointer, Initializing Pointers, Indirection Operator and Dereferencing, void Pointer, Null Pointer, Use of Pointers), Arrays and Pointers(One-dimensional Arrays and Pointers, Passing an Array to a Function, Differences between Array Name and Pointer), Pointer and String, Pointer Arithmetic (Assignment, Addition or Subtraction on Integer, Subtraction of Pointers, Comparing Pointers), Pointers to Pointers, Array of Pointers, Pointers to an Array, Two-dimensional Arrays and Pointers (Passing Two-dimensional Array to a Function), Three-dimensional Arrays, Pointers to Functions (Declaration of a Pointer to a Function, Initialization of Function Pointers, Calling a Function using a Function Pointer, Passing a Function to Another Function, How to Return a Function Pointer, Arrays of Function Pointers.	**06**
2	**Unit 2: Dynamic Memory Allocation & Advanced Pointer Programming** Dynamic Memory Allocation (Dynamic Allocation of Arrays, Freeing Memory, Reallocating Memory Blocks, Implementing Multidimensional Arrays using Pointers), Offsetting a Pointer, Memory Leak and Memory Corruption, Pointer and Const Qualifier (Pointer to Constant, Constant Pointers, Constant Parameters)	**06**
3	**Unit 3 User-defined Data Types and Variables: Structures, Unions, Enumerations, Bit-fields.** Structures (Declaring Structures and Structure Variables, Accessing the Members of a Structure, Initialization of Structures,Copying and Comparing Structures, typedef and its Use in Structure Declarations, Nesting of Structures, Arrays of Structures, Initializing Arrays of Structures, Arrays within the Structure, Structures and Pointers,	**08**

	Structures and Functions), Union (Declaring a Union and its Members, Accessing and Initializing Members of a Union, Structure Versus Union, Enumeration Types, Bitfields	
4	**Unit 4 : Files** Files in C (Using Files in C, Declaration of a File Pointer,Opening a File, Closing and Flushing Files) Working with Text Files (Character Input and Output, End of File (EOF), Detecting the End of a File using feof() Function), Working with Binary Files, Direct File Input and Output (Sequential Versus Random File Access), Files of Records (Working with Files of Records) Random Access to Files of Records, Other File Management Functions (Deleting a File, Renaming a File) Low-Level I/O	**08**
5	**Unit 5: Linked Lists** Singly Linked Lists (Insertion of a Node in a Singly Linked List, Deletion of a Node from a Singly Linked List, Sorting a Singly Linked List, Destroying a Singly Linked List, More Complex Operations on Singly Linked Lists),Circular Linked Lists (Appending a Node, Displaying a Circular Linked List, Inserting a Node after a Specified Node, Inserting a Node before a Particular Node, Deleting a Node, Sorting a Circular Linked List), Doubly Linked Lists (Operations on Doubly Linked Lists, Advantages/Disadvantages of Doubly Linked) Lists, Introduction to Circular Doubly Linked Lists, Applications of Linked Lists (Dynamic Storage Management, Garbage Collection and Compaction), Disadvantages of Linked Lists, Array versus Linked List Revisited	**08**
6	**Unit 6: Bitwise Operators & Pre-Processors** Bitwise Operator (Bitwise AND, Bitwise OR, Bitwise Exclusive-OR, Bitwise NOT, Bitwise Shift Operator), Command-line Arguments, The C Preprocessor (The C Preprocessor Directives, Predefined Identifiers), Type Qualifier (const Qualifier, volatile Qualifier, restrict Qualifier) Variable Length Argument List, Memory Models and Pointers	**04**

4. Text Book(s):

1. Programming in C, 2nd Edition, Pradip Dey, Manas Ghosh , OXFORD

5. Other Reference Books:

1. Programming in ANSI C, by Balaguru samy, Publisher - Tata McGraw Hill.
2. Programming with ANSI and Turbo C, by Ashok N Kamthane, Publisher – Pearson Education.
3. Mastering C, by Venugopal & Prasad, Publisher – Tata McGraw Hill.
4. C: The Complete Reference, by Herbert Schildt, Publisher – Tata McGraw Hill.
5. Let us C, by Yashwant Kanitkar, Publisher – BPB Publication

6. Accomplishment

After completion of the course students should become reasonably good at problem solving and algorithm development. They would become capable of solving problems using computers through C programming language.