FACULTY OF COMPUTER APPLICATIONS
Bachelor of Computer Applications

- **Sem.** : 2
- **Subject Code** : 05BC1203
- **Subject** : Programming in C - 2
- **Course Objectives** :
    1. To understand the advance structured and procedural programming to improve C programming skill.
    2. To understand various operations and types of array.
    3. To study the advantages of user defined data type which provides flexibility for application development
    4. To impart the knowledge about pointers; which is the backbone of effective memory handling.
    5. To develop basic understanding of file handling mechanism.
- **Prerequisites** : Basic knowledge of C programming.

Unit No	Topics Covered	No Of Lectures Required
1	**Array :** Introduction, One – Dimensional Array, Two – Dimensional Array, Multi – Dimensional Array **Character Arrays And Strings:** Introduction, Declaring and Initializing String Variables, Reading and Writing String, Arithmetic Operations on String, Concatenation and Comparison of Two Strings, String Handling Functions	10
2	**Structure And Union:** Introduction, Defining, Declaring, Accessing and Initializing Structure, Copying and Comparing Structure Variables, Operations on Individual Members, Arrays of Structure and Arrays Within Structure, Structure and Functions, Unions, Size of Structure, Bit Fields.	10

3	**Pointers :** Introduction, Declaring and Initialization of Pointer Variables, Accessing a Variable Through its Pointer, Pointer Chaining, Pointer Expressions, Pointer Increments and Scale Factor, Dynamic Memory Allocation (malloc, calloc, realloc And free), Pointers and Arrays, Dynamic Arrays, Pointers and Character Strings, Array of Pointers, Pointers as Function Arguments, Functions Returning Pointers, Pointers To Functions, Pointers and Structures.	10
4	**File Management :** Introduction, Defining and Opening a File, Closing a File, Input / Output Operations in File, Error Handling During I/O Operations, Random Access Files, Command Line Arguments	10
5	**Searching :** Introduction, Search Techniques – Linear search, Binary search (Iteration and Recursion) **Sorting :** Introduction, Sorting Techniques – Bubble sort, Insertion sort, Selection sort, Quick sort, Heap sort, Shell sort, Radix sort, Merge sort, Comparison of All Sorting Methods	10

Course Outcomes:

1. Ability to work with arrays of complex objects.
2. Ability to work with textual information, characters and strings.
3. Ability to develop advanced applications using enumerated data types, function pointers and nested structures.
4. Ability to implement programs with pointers and arrays, perform pointer arithmetic, and use the dynamic memory allocation.
5. Ability to work with basics of file handling mechanism that is essential for understanding the concepts in database management systems.
6. Ability to find out better searching and sorting techniques for list of data.

FACULTY OF COMPUTER APPLICATIONS
Bachelor of Computer Applications

Text Book:

1. Programming in ANSI C, E. Balagurusamy, McGraw Hill, 7th Edition.
2. An Introduction to Data Structures with Applications, J. P. Tremblay , P. G. Sorenson, McGraw Hill, 2nd Edition.

Reference Books:

1. Let Us C, Yashavant Kanetkar, BPB Publication, 10thEdition.
2. Programming in C, Reema Thareja, Oxford University Press.
3. Mastering C, by Venugopal & Prasad, Tata McGraw Hill.
4. The Complete Reference, Herbert Schildt,Tata McGraw Hill.
5. The C Programming Language, Brian W. Kernighan and Dennis M. Ritchie, 2nd edition.

Web References:

1. https://www.w3schools.in/c-tutorial
2. https://www.tutorialspoint.com/cprogramming

App References:

1. W3Schools App
2. Tutorials Point App

Syllabus Coverage from text /reference book & web/app reference:

Unit No	Chapter Numbers
1	Book – 1, Chap. 7 (except dynamic arrays), chap. 8
2	Book – 1, Chap. 10
3	Book – 1, Chap . 11, Chap. 13
4	Book – 1, Chap. 12
5	Book – 2, Chap . 6 (6.1.1 to 6.1.7, 6.2.1,6.2.2)

Marwadi University

FACULTY OF COMPUTER APPLICATIONS
Bachelor of Computer Applications

PRACTICALS

Unit No	List of Practicals
1	1. Enter N elements and print them. 2. Enter N elements and find total and average of them. 3. Enter N elements and find maximum and minimum value. 4. Enter N elements and find how many are positive, negative, even and odd. 5. Copy the elements of one array into another. 6. Enter N elements and arrange them in reverse order. 7. Merge two arrays. 8. Write a program to print 3 * 3 matrix. 9. Write a program to add two matrices. 10. Write a program to subtract two matrices. 11. Write a program to multiply two matrices. 12. Enter your name and print it vertically. 13. Enter a string and find out length of string with using string function and without string function. 14. Enter a string and print it in reverse order with using string function and without string function. 15. Enter a string and find out how many spaces and vowels in the string. 16. Compare two strings and find out they are same or not. 17. Check the string is palindrome or not.
2	1. Design a structure named student to store data about student which contains following data element : Data Item Type Length Rollno int ------- Name char 20 Department char 40 Year of joining int ---- Score float ------ Write a program to input data and display data. 2. Modify above program for 5 students record. 3. Sort above data by name. 4. Sort above data by score in descending order. 5. Write a function to print names of students whose grade is greater than 5.0.

	6. Write a function to print data of all students who joined in 2019. 7. Add Two Complex Numbers by Passing Structure to a Function. 8. Demonstrate difference between structure and union. 9. Write a program that compares two given dates. To store a date use a structure that contains three members namely date, month and year. If the dates are equal then display message "equal" otherwise "not equal". 10. Write a program to demonstrate nested structure.
3	1. Swap two variables using pointer and function. 2. find out area of circle using pointer and function 3. Find out maximum and minimum number using pointer to function. 4. Enter N elements and find out sum and average of them using dynamic array. 5. Enter N elements and find out sum of prime numbers using dynamic array. 6. Enter N elements and find out sum of odd and even numbers using dynamic array 7. Add two matrices using array of pointers. 8. Subtract two matrices using pointer to pointer. 9. Enter 5 strings and arrange them in ascending order. 10. Enter 2 strings and concate them. 11. Enter a string and print it in uppercase letters. 12. Enter a string and print it in lowercase letters. 13. Enter a string and print it in reverse order. 14. Compare two strings and find out strings are same or not. 15. Copy a string into another. 16. Enter a string and find out the string is palindrome or not.
4	1. Wap to write content in file and then read that file and print the content on screen. 2. Wap to read a file and count no. of characters,blank spaces, tabs and lines in file 3. Wap to copy one file int another file. 4. Wap to append the content in given file. 5. Wap to write string in "greencity.txt" and print it in reverse order in file "cleancity.txt" 6. Wap to write a file DATA.Txt which has 1 to 10 numbers and then read this file. If number is odd then store it in "ODD.txt" and if it is even store it in "EVEN.txt"

	7. Wap to add records in file as per following structure. (empcode, empname , salary)then read the content and find out employee who earns highest salary. 8. Wap to add records in binary file as per following structure. (empcode, empname, salary) and read the content and print on screen 9. Wap to read a file and find out no. of upper case letters , lower case letters, special symbols, digits and vowels in the file. 10. Wap to read a file and replace all 'a' with 'x'. 11. Wap to find the size of file without traversing character by character. 12. Wap to enter a number in one file and print its reverse in another file. 13. Wap to print no. of arguments and list of strings given in command line. 14. Wap to copy one file into another using command line arguments. 15. Wap which demonstrate fseek(), ftell() and rewind() functions.
5	1. Write a program to enter N elements and search a value from the list using linear search. 2. Write a program to enter N elements and search a value from the list using binary search (without recursion). 3. Write a program to enter N elements and search a value from the list using binary search. (with recursion) 4. Write a program to enter N elements and arrange the list in ascending order using bubble sort. 5. Write a program to enter N elements and arrange the list in ascending order using selection sort. 6. Write a program to enter N elements and arrange the list in ascending order using shell sort. 7. Write a program to enter N elements and arrange the list in ascending order using quick sort. 8. Write a program to enter N elements and arrange the list in ascending order using merge sort. 9. Write a program to enter N elements and arrange the list in ascending order using Insertion sort. 10. Write a program to sort 5 strings in descending order.