| Sr. No | Module Name | Name of Program |
|---|---|---|
| 1. | **Familiarization with programming environment** Introduction to Programming, Writing of Algorithms, Introduction to Drawing flow Charts /Preparation of Flowchart/ Steps for Writing Code in C/ Turbo C. | 1. First Basic Program-Writing a Single Statement.<br>2. Writing a Program to print your Basic details Multi statements. |
| 2 | **Variable types and type conversions** | 1. WAP to perform simple Input-Output Operations in C.<br>2. WAP to add two numbers.<br>3. WAP to perform simple arithmetic operations in C(Addition, Subtraction, Multiplication, Division, Modulus).<br>4. Write a simple program that print the result of all the operators available in c (including pre/ post increment, bitwise and logical).<br>5. WAP to find area and perimeter of circle.<br>6. WAP to find area and perimeter of rectangle.<br>7. Given the values of three variable entered by user, write a program to compute and display the value of x, where x=a/(b-c).<br>8. Write a program to convert one data type to another using auto conversion and casting. Take the value from user input. |
| 3 | **Branching and logical expressions**: Use of If, if- else, Else if, nested if statements and operators with them and switch case statement | 1. WAP to find whether a given number is positive or not.<br>2. WAP to find greatest of two numbers.<br>3. WAP to find greatest of three numbers using nested if/else if statements only.<br>4. WAP to find greatest of three numbers using & operator.<br>5. WAP to find whether a given number is even or odd.<br>6. Given the marks of a student studying five different subject. Calculate average marks of students and assign him/her Grade based on following: Marks is equal or greater than 90 – Grade A Marks equal or more than 75 and less than 90 –Grade B Marks equal or more than 60 and less than 75 –Grade C Marks equal or more than 50 and less than 60 –Grade D Marks less than 50 –Grade F<br>7. WAP to find roots of a quadratic equation: ax2+bx+c=0<br>8. WAP to print day of a week using switch case statement<br>9. WAP to design a simple calculate using switch-case statements |
| 4. | **Loops: do, while and for loops:** Use of while loop, do while, and for loop:their Syntax | 1. WAP to print counting 1 to 10 using all loop<br>2. WAP to print table of any number.<br>3. WAP to print the factorial of given number.<br>4. WAP to print the sum of digits of a given number. |

| | | |
|---|---|---|
| | | 5. WAP to print the Fibonacci series up to 10 level.<br>6. WAP to find whether the given number is Armstrong or Not.<br>7. WAP to find whether the given number is Palindrome or Not.<br>8. WAP to find whether the given number is prime or not.<br>9. WAP to reverse the digits of a given number. |
| 5. | **1D Arrays, 2 D array**<br>Declaration of arrays, syntax, semantics, | 1. Program to insert 5 elements into array and print elements of array.<br>2. WAP to merge two sorted array in one sorted array.<br>3. WAP to add two matrices in 2-D array<br>4. WAP to multiply two matrices in 2-D array.<br>5. WAP to find transpose of a Matrix.<br>6. WAP to find average of 10 numbers using array.<br>7. WAP to print the following numbers in reverse order using array. |
| 6. | **Functions**<br>Simple function declaration, definition,functions with return type, call by value. | 1. WAP to create function display a simple message.<br>2. WAP to create function to add two numbers.<br>3. WAP to create a function to swap two numbers using call by value.<br>4. WAP to generate Fibonacci series using recursive function.<br>5. WAP to swip two integers using call by value and call by reference method of passing arguments to a function. |
| 7. | **Pointers**<br>Pointer declaration, use of pointers in array, functions, call by reference, recursive functions | 1. WAP to understand basic use of pointers.<br>2. WAP to implement call by reference for swapping of two numbers.<br>3. WAP to calculate factorial of a number using recursion.<br>4. WAP to Fibonacci series up to 20 using recursive numbers |
| 8. | **Structure**<br>Basic of Structure, Union and accessing data of structure. | 1. WAP for user defined data type namely Student and implement it using Structure<br>2. WAP for user defined data type namely Book and implement it using Structure..<br>3. WAP to create an array of structure. |
| 9 | **File Operations**<br>File opening modes, creation of files,reading and writing data files. | 1. WAP to read a simple file using file handling.<br>2. WAP to write data in file.<br>3. WAP to append data in existing file. |
| 10. | **Searching and sorting**<br>Various searching and sorting algorithms. | 1. WAP to implement linear search<br>2. WAP to implement binary search<br>3. WAP to implement selection sort.<br>4. WAP to implement insertion sort.<br>5. WAP to implement quick sort.<br>6. WAP to implement merge sort.<br>7. WAP to implement bubble sort. |