

Topics to Study

**Mod 1**

Data Types

Various types of Operators

**Mod 2**

Decision Making

Looping control

Jump statements

**Mod 3**

Arrays

Character Arrays and Strings

**Mod 4**

User Defined Functions

Structures and Unions

**Mod 5**

Pointers

Dynamic Memory Allocation using Pointers

File Handling

Mod 1 – Exp 1

Mod 2 – Exp 2 , Exp 3

Mod 3 – Exp 4 , Exp 5

Mod 4 – Exp 6, Exp 7

Mod 5 – Exp

**Exp 1**

1. Program to find area and circumference of various Geometric shapes.
2. b. Program to calculate EMI (Equated Monthly Instalment) of loan amount if principal, rate of interest and time in years is given by the user.

E = (P.r.(1+r)n) / ((1+r)n – 1)

***Extra***

Develop a C program to determine the total cost of a shopping cart. Prompt the user to input the price and quantity of each item purchased, and then calculate the subtotal, tax (considering a fixed tax rate), and the total amount including tax. Use the formula:

Total Cost = Subtotal + Tax

Tax = Subtotal \* TaxRate

Subtotal = ∑(price of each item \* quantity of each item)

#include<stdio.h>

void main(){

int total,subtotal,total\_tax,single\_price,quantity,tax\_percent;

float taxrate;

printf("Enter the price of item : ");

scanf("%d",&single\_price);

printf("Enter the quantity of item(no. of items): ");

scanf("%d",&quantity);

printf("Enter the tax rate(in percent): ");

scanf("%d",&tax\_percent);

taxrate = (float)tax\_percent/100;

subtotal = single\_price\*quantity;

total\_tax = subtotal \* taxrate;

total = subtotal + total\_tax;

printf("Total value of your items is : %d",subtotal);

printf("\nTotal tax of your items is : %d",total\_tax);

printf("\nFinal Bill : %d",total);

}

**Exp 2**

1. Write a program to accept 3 numbers from the user and find the largest of the 3 numbers using

**If - else if-else**

**Ternary operator**

#include<stdio.h>

void main(){

int n1,n2,n3;

printf("Enter three numbers : ");

scanf("%d %d %d",&n1,&n2,&n3);

if(n1>n2 && n1>n3){

printf("n1 is greatest.");

}

else if(n2>n3 && n2>n1){

printf("n2 is greatest");

}

else{

printf("n3 is great");

}

}

1. Write a C program to find the grade of a student using switch case statements.

Write a C program to determine the type of a triangle based on the lengths of its three sides. The program should prompt the user to enter the lengths of the three sides and then classify the triangle as follows:

Equilateral: All sides are equal.

Isosceles: Two sides are equal.

Scalene: No sides are equal.

#include<stdio.h>

void main(){

int s1,s2,s3;

printf("Enter the length of three side : ");

scanf("%d %d %d",&s1,&s2,&s3);

if(s1==s2 && s2==s3){

printf("It is equilateral.");

}

else if(s1==s2 || s2==s3 || s3==s1){

printf("Isoceles");

}

else{

printf("Scalene");

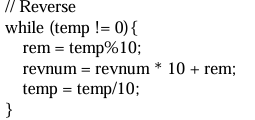
}

}

**Exp 3**

Write a menu driven program for following option

1. To find whether a number is palindrome or not. (e.g. 1221 is palindrome) using while loop only)



1. To calculate the sum of the Fibonacci series up to ‘n’ terms(use do-while loop
2. Write a program in C to make such a pattern like a right angle triangle with a number which will repeat a number in a row. (Pattern is given below)

Write a program to enter numbers till the user wants. At the end it should display the count of positive, negative and zeros entered.

Write a program to print all the ASCII values and their equivalent characters using a while loop. The ASCII values vary from 0 to 255.

Write a C program to check whether a given number is a prime number or not using a while loop.

#include<stdio.h>

void main(){

int num,i = 2;

printf("Enter a number : ");

scanf("%d",&num);

while(i<num){

if(num%i==0){

printf("%d is not a prime number",num);

return;

}

i++;

}

printf("The number is prime");

}

// Write a C program to generate and print the first N terms of the Fibonacci sequence using a for loop.

#include<stdio.h>

void main(){

int t1=0,t2=1,next,num;

printf("Enter a number : ");

scanf("%d",&num);

printf("%d %d",t1,t2);

for(int i=3;i<=num;i++){

next = t1+t2;

printf(" %d",next);

t1=t2;

t2=next;

}

}

**Exp 4**

a. Program to sort the 1D array in the ascending or descending order and then accept the element from user and insert in the same array at its correct place by keeping array sorted.

b. Write a program to find the Transpose of a Matrix.

1. Write a program to enter n numbers, store them in an array and rearrange the array in the reverse order.
2. Write a program which performs the following tasks:

Initialize an integer array of 10 elements in main( )

Pass the entire array to a function modify( )

In modify( ) multiply each element of array by 3

Return the control to main( ) and print the new array elements in main( )

**Exp 5**

1. Write a program that searches for a substring within a given string.
2. Write a program to check if one string is the rotation of another.

Write a C program to toggle case of each character in a string i.e. if a character is in uppercase, change it to lower case and vice-versa.

Write a program in C to reverse a given string without using any library functions.

#include<stdio.h>

void main(){

int n;

char inp[100];

printf("Enter the length of the Array :");

scanf("%d",&n);

printf("Enter the string :");

scanf("%s",inp);

printf("Reversed array : ");

for(int j=n-1;j>=0;j--){

printf("%c",inp[j]);

}

}

**Exp 6**

a) Write a program to find the GCD of two numbers using recursion.

b) Write a program to find the LCM of two numbers by using a) above.

Write a C program to find the minimum, maximum and sum of elements in an array using functions.

**Exp 7**

Write a program to manage an employee database using structure and union in C. Each Employee has the following information:

1. Employee ID(integer)

2. Name(string)

3. Department(string)

4. Salary(float)

You need to implement the following functionalities:

1. Create a structure named Employee with the appropriate data members to store the information mentioned above.

2. Create a union named EmployeeInfo that can hold either the Name or Department information.

3. Write a function addEmployee that takes user input for each employee's information and stores it in an array of structures.

4. Write a function printEmployeeDetails that takes an employee's ID as input and prints all available details for that employee.

5. Write a function updateEmployeeInfo that takes an employee's ID and allows the user to update either the Name or Department information using the EmployeeInfo union.

6. Implement a menu-driven program that allows the user to perform the above operations. Include options to add a new employee, print employee details, update employee information, and exit the program.

WAP to accept student name, roll number and percentage for 10 students using array of structures and arrange them in descending order of their percentage.

WAP to display employee name, ID and year of experience using union.

**Exp 8**

1. Write a program that calculates and prints the transpose of a given matrix using pointers(use function to find transpose of a matrix).
2. WAP to accept a string from the user and calculate the length of a given string using pointers.

Write a file copy program in C that copies a file into another.

WAP to count the number of characters and number of lines in a file.

Write a program in C to merge two files into a third file. Your program should prompt the user to enter the names of the two input files and the name of the output file. Then, it should open the input files, read their contents, and write them into the output file.

#include<stdio.h>

void main(){

FILE \*srcfile\_1,\*srcfile\_2,\*dstfile;

char ch\_1,ch\_2,filename1[999],filename2[999],filename3[999];

printf("Enter the name of the first source file : ");

scanf("%s",filename1);

printf("Enter the name of the second source file : ");

scanf("%s",filename2);

printf("Enter the name of the destination file : ");

scanf("%s",filename3);

srcfile\_1 = fopen(filename1,"r");

srcfile\_2 = fopen(filename2,"r");

dstfile = fopen(filename3,"w");

while((ch\_1 = fgetc(srcfile\_1)) != EOF){

fputc(ch\_1,dstfile);

}

fputc('\n',dstfile);

while((ch\_2 = fgetc(srcfile\_2)) != EOF){

fputc(ch\_2,dstfile);

}

fclose(srcfile\_1);

fclose(srcfile\_2);

fclose(dstfile);

printf("Both file contents have been successfully copied...");

}