

RAJKIYA ENGINEERING COLLEGE AMBEDKAR NAGAR

BRANCH:

(Department of Applied Science and Humanities)



PRACTICAL FILE

ON

**PRINCIPLE OF PROGRAMMING
LANGUAGE (KCS-151P)**

**SESSION
2021-2022**

SUBMITTED TO:

SUBMITTED BY:

ROLL NO:

BRANCH:

SEMESTER:

(INDEX)

Sr. No.	Program	Page No.	Date	Remark
1.	Write a program to calculate the area of triangle using formula $at=\sqrt{s(s-a)(s-b)(s-c)}$			
2.	Basic salary of an employee is input through the keyboard. The DA is 25% of the basic salary while the HRA is 15% of the basic salary. Provident Fund is deducted at the rate of 10% of the gross salary (BS+DA+HRA). Program to calculate the Net Salary.			
3.	Write a program to determine the roots of quadratic equation.			
4.	Write a program to find the largest of three numbers using nested if else.			
5.	Write a program to receive marks of physics, chemistry & maths from user & check its eligibility for course if a) Marks of physics > 40 b) Marks of chemistry > 50 c) Marks of math's > 60 d) Total of physics & math's marks > 150 or e) Total of three subjects marks > 200			
6.	Write a program to find the value of y for a particular value of n. The a, x, b, n is input by user if n=1 $y=ax\%b$ if n=2 $y=ax^2+b^2$ if n=3 $y=a-bx$ if n=4 $y=a+x/b$			
7.	Write a program to construct a Fibonacci series upto n terms.			
8.	Write a program to find whether the number is Armstrong number.			
9.	Write a program to generate sum of series $1!+2!+3!+\dots+n!$			

10.	Write a program to find the sum of following series $1 - X1/1! + X2/2! - \dots Xn/n!$.			
11.	Write a program to print the entire prime no between 1 and 300.			
12.	Write a program to print out all the Armstrong number between 100 and 500.			
13.	Write a program to draw the following figure: 3 2 1 2 1 * ** ***			
14.	Write a program to receive a five-digit no and display as like 24689: 2 4 6 8 9			
15.	Write a function that return sum of all the odd digits of a given positive no entered through keyboard.			
16.	Write a program to print area of rectangle using function & return its value to main function.			
17.	Write a program to calculate the factorial for given number using function.			
18.	Write a program to find sum of Fibonacci series using function.			
19.	Write factorial function & use the function to find the sum of series $S = 1! + 2! + \dots + n!$.			

20.	Write a program to find the factorial of given number using recursion.			
21.	Write a program to find the sum of digits of a 5 digit number using recursion.			
22.	Write a program to calculate the GCD of given numbers using recursion.			
23.	Write a program to convert decimal number in to binary number.			
24.	Write a program to convert binary number in to decimal number.			
25.	Write a program to delete duplicate element in a list of 10 elements & display it on screen.			
26.	Write a program to merge two sorted array & no element is repeated during merging.			
27.	Write a program to evaluate the addition of diagonal elements of two square matrixes.			
28.	Write a program to find the transpose of a given matrix & check whether it is symmetric or not.			
29.	Write a program to print the multiplication of two N*N (Square) matrix.			
30.	Write a program in C to check whether the given string is a palindrome or not.			
31.	Write program to sort the array of character (String) in alphabetical order like STRING in GINRST.			
32.	Write a program to remove all the blank space from the string & print it, also count the no of characters.			
33.	Write a program to store the following string “zero”, “one” ----- “five”. Print the no in words, given in figure as 3205.			
34.	Write a program to compare two given dates. To store a date uses a structure that contains three members namely day, month and year. If the dates are equal then display message equal otherwise unequal.			
35.	Define a structure that can describe a hotel. It should have the member that includes the name, address, grade, room charge and number of rooms. Write a function to print out hotel of given grade in order of room charges.			
36.	Define a structure called cricket with player name, team name, batting average, for 50 players & 5 teams. Print team wise list contains names of player with their batting average.			
37.	Write a c program to copy & count the character content of one file says a.txt to another file b.txt.			
38.	Write a program to take 10 integers from file and write square of these integer in other file.			
39.	Write a program to read number from file and then write all ‘odd’ number to file ODD.txt & all even to file EVEN.txt.			
40.	Write a program to print all the prime number, between 1 to 100 in file prime.txt.			
41.	Write the following C program using pointer: a) To sort the list of numbers through pointer b) To reverse the string through pointer.			

42.	Write a program to find the largest no among 20 integers array using dynamic memory allocation.			
43.	Using Dynamic Memory Allocation, Write a program to find the transpose of given matrix.			
44.	Write a program to find the factorial of given number using command line argument.			
45.	Write a program to find the sum of digits of a 5 digit number using command line argument.			

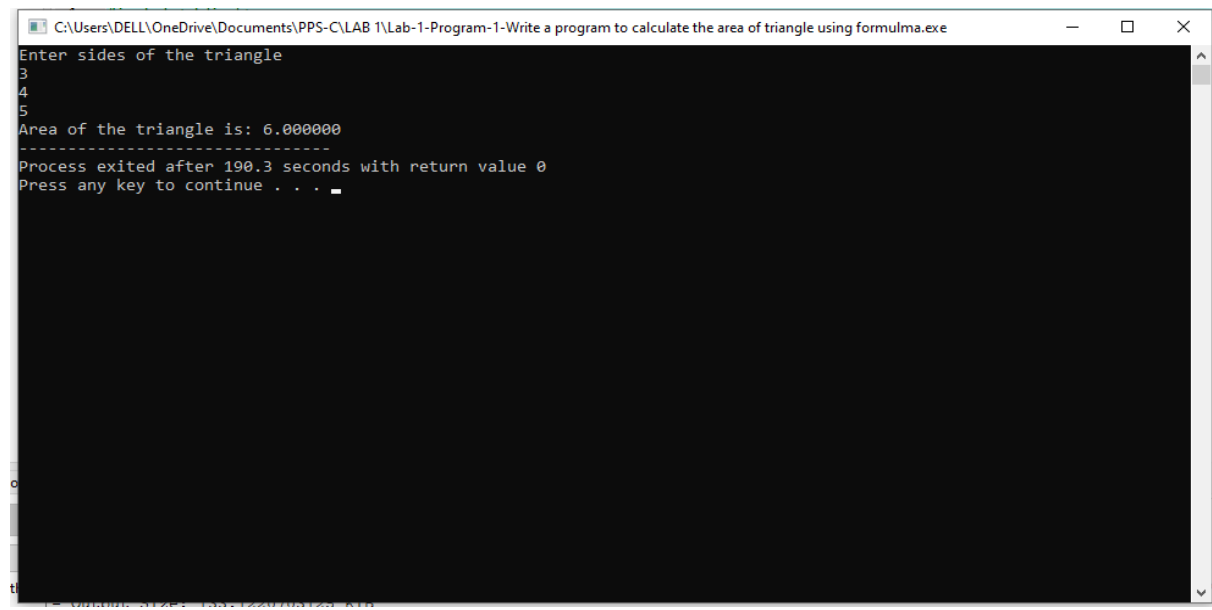
Lab-1

Program-1: Write a program to calculate the area of triangle using formula $at=\sqrt{s(s-a)(s-b)(s-c)}$

Code:

```
#include<stdio.h>
#include<math.h>
int main()
{
    float s,a,b,c,area;
    printf("Enter sides of the triangle\n");
    scanf("%f%f%f",&a,&b,&c);
    s=(a+b+c)/2;
    area=sqrt(s*(s-a)*(s-b)*(s-c));
    printf("Area of the triangle is: %f",area);
    return 0;
}
```

Output:



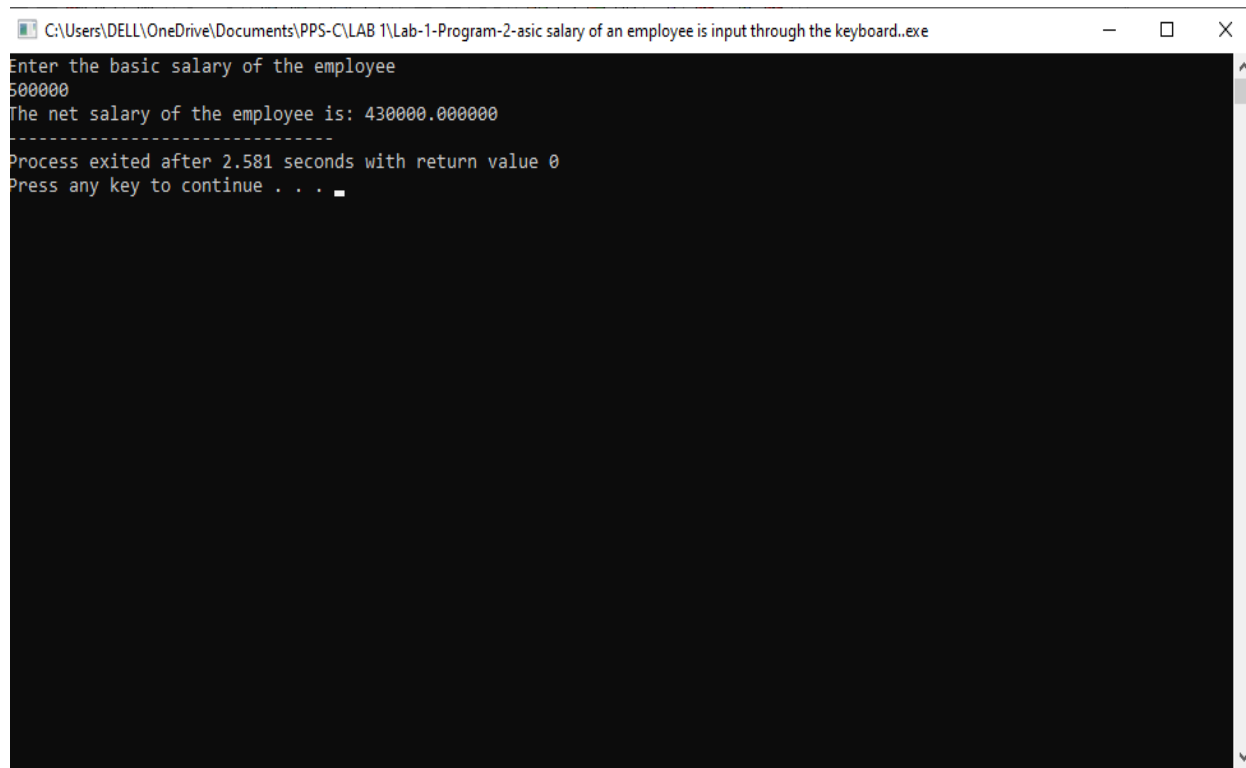
```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 1\Lab-1-Program-1-Write a program to calculate the area of triangle using formula.exe
Enter sides of the triangle
3
4
5
Area of the triangle is: 6.000000
-----
Process exited after 190.3 seconds with return value 0
Press any key to continue . . .
```

Program-2: Basic salary of an employee is input through the keyboard. The DA is 25% of the basic salary while the HRA is 15% of the basic salary. Provident Fund is deducted at the rate of 10% of the gross salary (BS+DA+HRA). Program to calculate the Net Salary.

Code:

```
#include<stdio.h>
#include<math.h>
int main()
{
    float BS,DA,HRA,Net_salary;
    printf("Enter the basic salary of the employee\n");
    scanf("%f",&BS);
    DA=0.25*BS; // 25 percent of the basic salary
    HRA=0.15*BS; // 15 percent of the basic salary
    Net_salary=BS-(0.10*(BS+DA+HRA));
    printf("The net salary of the employee is: %f",Net_salary);
    return 0;
}
```

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 1\Lab-1-Program-2-asic salary of an employee is input through the keyboard..exe
Enter the basic salary of the employee
500000
The net salary of the employee is: 430000.000000
-----
Process exited after 2.581 seconds with return value 0
Press any key to continue . . .
```

Program-3: Write a program to determine the roots of quadratic equation.

Code:

```
#include <math.h>
#include <stdio.h>
int main() {
    double a, b, c, discriminant, root1, root2, realPart, imagPart;
    printf("Enter coefficients a, b and c: ");
    scanf("%lf %lf %lf", &a, &b, &c);

    discriminant = b * b - 4 * a * c;

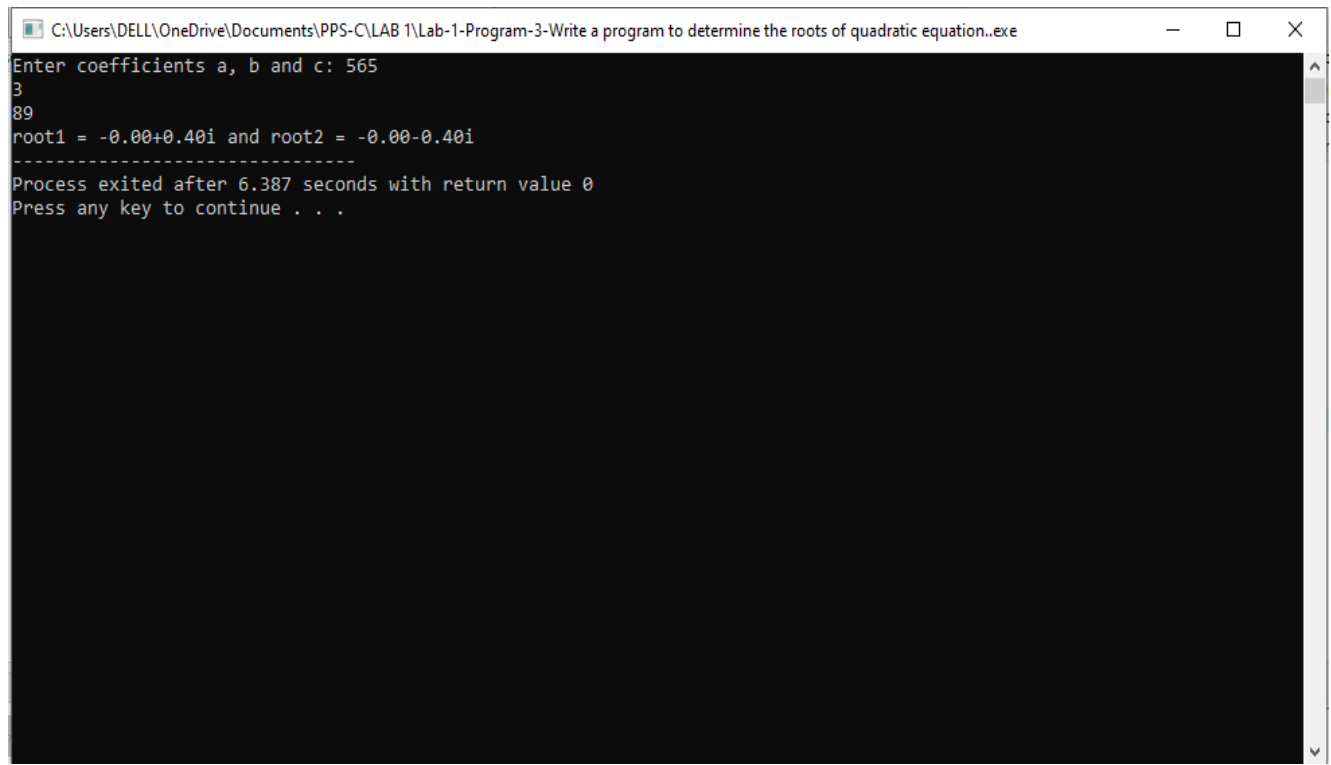
    // condition for real and different roots
    if (discriminant > 0) {
        root1 = (-b + sqrt(discriminant)) / (2 * a);
        root2 = (-b - sqrt(discriminant)) / (2 * a);
        printf("root1 = %.2lf and root2 = %.2lf", root1, root2);
    }

    // condition for real and equal roots
    else if (discriminant == 0) {
        root1 = root2 = -b / (2 * a);
        printf("root1 = root2 = %.2lf;", root1);
    }

    // if roots are not real
    else {
        realPart = -b / (2 * a);
        imagPart = sqrt(-discriminant) / (2 * a);
        printf("root1 = %.2lf+%.2lfi and root2 = %.2f-%.2fi", realPart, imagPart, realPart, imagPart);
    }

    return 0;
}
```

Output:



A screenshot of a Windows command prompt window. The title bar shows the file path: C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 1\Lab-1-Program-3-Write a program to determine the roots of quadratic equation..exe. The window has standard minimize, maximize, and close buttons. The command prompt shows the following text: "Enter coefficients a, b and c: 565", followed by "3" and "89" on separate lines. Then it displays "root1 = -0.00+0.40i and root2 = -0.00-0.40i". Below this is a dashed line, followed by "Process exited after 6.387 seconds with return value 0" and "Press any key to continue . . .". The rest of the window is black.

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 1\Lab-1-Program-3-Write a program to determine the roots of quadratic equation..exe
Enter coefficients a, b and c: 565
3
89
root1 = -0.00+0.40i and root2 = -0.00-0.40i
-----
Process exited after 6.387 seconds with return value 0
Press any key to continue . . .
```

Program-4: Write a program to find the largest of three numbers using nested if else.

Code:

```
#include <stdio.h>

int main() {

    double n1, n2, n3;

    printf("Enter three numbers: ");
    scanf("%lf %lf %lf", &n1, &n2, &n3);

    // outer if statement
    if (n1 >= n2) {

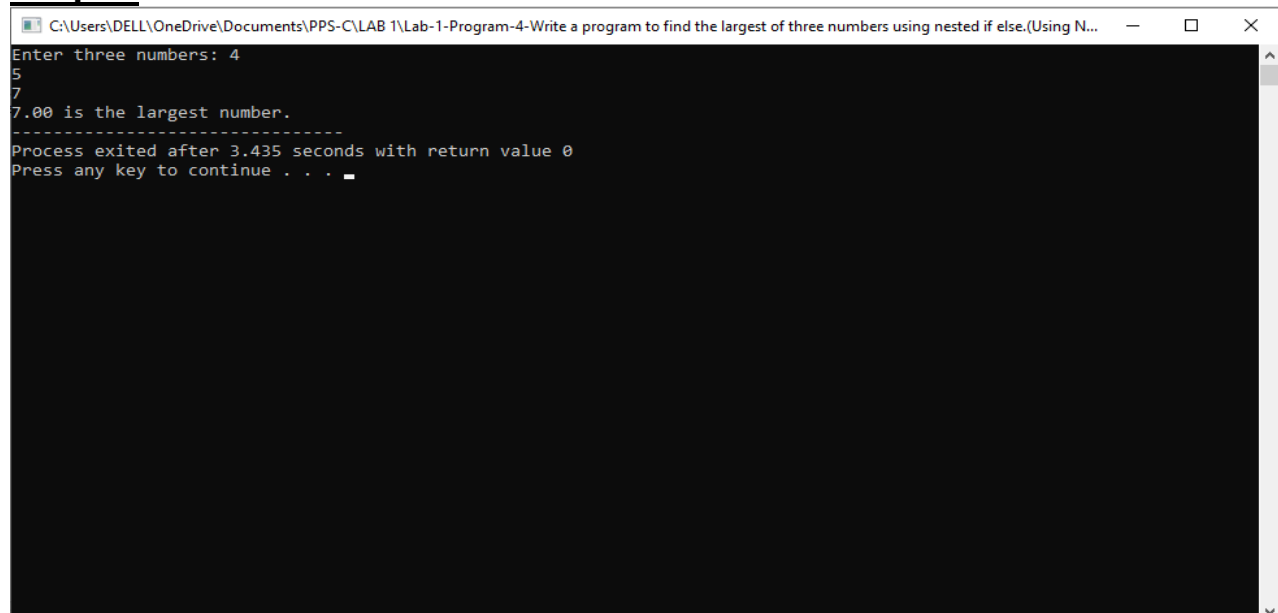
        // inner if...else
        if (n1 >= n3)
            printf("%.2lf is the largest number.", n1);
        else
            printf("%.2lf is the largest number.", n3);
    }

    // outer else statement
    else {

        // inner if...else
        if (n2 >= n3)
            printf("%.2lf is the largest number.", n2);
        else
            printf("%.2lf is the largest number.", n3);
    }

    return 0;
}
```

Output:



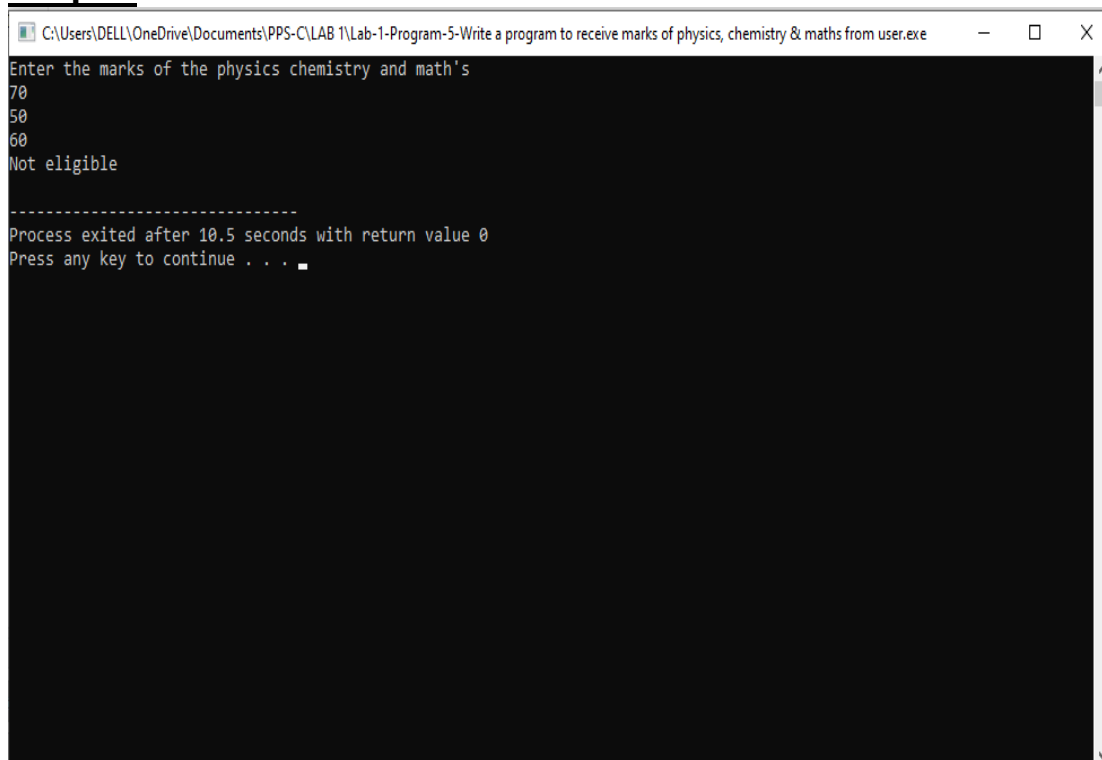
Program-5: Write a program to receive marks of physics, chemistry & maths from user & check its eligibility for course if

- a) Marks of physics > 40
- b) Marks of chemistry > 50
- c) Marks of math's > 60
- d) Total of physics & math's marks > 150 or
- e) Total of three subjects marks > 200

Code:

```
#include<stdio.h>
#include<math.h>
int main()
{
    int phy,che,math,total,PM;
    printf("Enter the marks of the physics chemistry and math's\n");
    scanf("%d%d%d",&phy,&che,&math);
    total=(phy+che+math);
    PM=phy+math;
    if(phy>40 && che>50 && math>60 && (PM>150||total>200))
    {
        printf("You are eligible for the course\n");
    }
    else
        printf("Not eligible\n");
    return 0;
}
```

Output:



Lab-2

Program-6: Write a program to find the value of y for a particular value of n. The a, x, b, n is input by user if n=1 $y=ax\%b$ if n=2 $y=ax^2+b^2$ if n=3 $y=a-bx$ if n=4 $y=a+x/b$

Code:

/*Write a program to find the value of y for a particular value of n. The a, x, b, n is input by user

if n=1 $y=ax\%b$

if n=2 $y=ax^2+b^2$

if n=3 $y=a-bx$

if n=4 $y=a+x/b$ */

```
#include<stdio.h>
```

```
#include<math.h>
```

```
int main()
```

```
{
```

```
    int a,x,b,n,y;
```

```
    printf("Enter the value of a,x,b,n\n");
```

```
    scanf("%d%d%d%d",&a,&x,&b,&n);
```

```
    if(n==1)
```

```
        y=(a*x)%b;
```

```
    if(n==2)
```

```
        y=(a*x*x)+(b*b);
```

```
    if(n==3)
```

```
        y=a-b*x;
```

```
    if(n==4)
```

```
        y=a+(x/b);
```

```
    printf("Value of the y is: %d",y);
```

```
    return 0;
```

```
}
```

Output:

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 2\Lab-2-Program-6-Write a program to find the value of y for a particular value of n. The a, x, exe
Enter the value of a,x,b,n
5
2
9
2
Value of the y is: 101
-----
Process exited after 11.57 seconds with return value 0
Press any key to continue . . .
```

Program-7: Write a program to construct a Fibonacci series upto n terms.

Code:

```
//Write a program to construct a Fibonacci series upto n terms.
```

```
//Fibonacci Series up to n terms
```

```
#include <stdio.h>
```

```
int main() {
```

```
    int i, n;
```

```
    // initialize first and second terms
```

```
    int t1 = 0, t2 = 1;
```

```
    // initialize the next term (3rd term)
```

```
    int nextTerm = t1 + t2;
```

```
    // get no. of terms from user
```

```
    printf("Enter the number of terms: ");
```

```
    scanf("%d", &n);
```

```
    // print the first two terms t1 and t2
```

```
    printf("Fibonacci Series: %d, %d, ", t1, t2);
```

```
    // print 3rd to nth terms
```

```
    for (i = 3; i <= n; ++i) {
```

```
        printf("%d, ", nextTerm);
```

```
        t1 = t2;
```

```
        t2 = nextTerm;
```

```
        nextTerm = t1 + t2;
```

```
    }
```

```
    return 0;
```

```
}
```

Output:


```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 2\Lab-2-Program-7-Write a program to construct a Fibonacci series upto n terms.(Fibonacci Series up ...  
Enter the number of terms: 45  
Fibonacci Series: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946, 17711  
, 28657, 46368, 75025, 121393, 196418, 317811, 514229, 832040, 1346269, 2178309, 3524578, 5702887, 9227465, 14930352, 24  
157817, 39088169, 63245986, 102334155, 165580141, 267914296, 433494437, 701408733,  
-----  
Process exited after 6.209 seconds with return value 0  
Press any key to continue . . .
```

Program-8: Write a program to find whether the number is Armstrong number.

Code:

```
//Write a program to find whether the number is Armstrong number.
```

```
//Check Armstrong Number of n digits
```

```
#include <math.h>
```

```
#include <stdio.h>
```

```
int main() {
```

```
    int num, originalNum, remainder, n = 0;
```

```
    float result = 0.0;
```

```
    printf("Enter an integer: ");
```

```
    scanf("%d", &num);
```

```
    originalNum = num;
```

```
    // store the number of digits of num in n
```

```
    for (originalNum = num; originalNum != 0; ++n) {
```

```
        originalNum /= 10;
```

```
    }
```

```
    for (originalNum = num; originalNum != 0; originalNum /= 10) {
```

```
        remainder = originalNum % 10;
```

```
        // store the sum of the power of individual digits in result
```

```
        result += pow(remainder, n);
```

```
    }
```

```
    // if num is equal to result, the number is an Armstrong number
```

```
    if ((int)result == num)
```

```
        printf("%d is an Armstrong number.", num);
```

```
    else
```

```
        printf("%d is not an Armstrong number.", num);
```

```
    return 0;
```

```
}
```

Output:

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 2\Lab-2-Program-8-Write a program to find whether the number is Armstrong number.(Check Armstr...
Enter an integer: 45673
45673 is not an Armstrong number.
-----
Process exited after 4.816 seconds with return value 0
Press any key to continue . . .
```

Program-9: Write a program to generate sum of series $1!+2!+3!+\dots+n!$

Code:

//Write a program to generate sum of series $1!+2!+3!+\dots+n!$

```
#include<stdio.h>
```

```
int main()
```

```
{
```

```
    int num,i,j,fact,sum=0;//variables
```

```
    printf("Enter the last number of series:\n");
```

```
    scanf("%d",&num);//last number of series
```

```
    for(i=1;i<=num;i++)//loop for finding factorial and sum
```

```
    {
```

```
        fact=1;
```

```
        if(i!=num)
```

```
            printf("%d!+ ",i);
```

```
        else
```

```
            printf("%d!= ",i);
```

```
        for(j=1;j<=i;j++)
```

```
            fact=fact*j;
```

```
        sum=sum+fact;
```

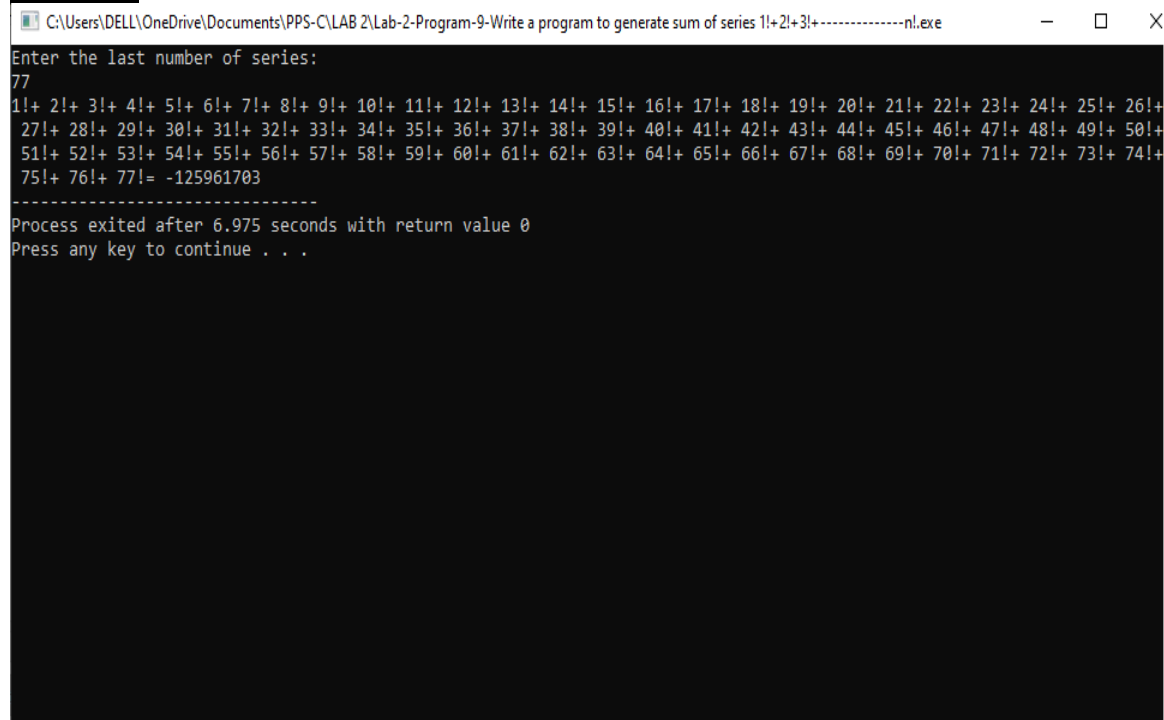
```
    }
```

```
    printf("%d",sum);
```

```
    return 0;
```

```
}
```

Output:



Program-10: Write a program to find the sum of following series $1 - X^1/1! + X^2/2! - \dots X^n/n!$.

Code:

```
/*Write a program to find the sum of following series 1-X1/1!+X2/2!-
.....Xn/n!.*/*
```

```
#include<stdio.h>
```

```
#include<math.h>
```

```
// function for finding factorial
```

```
int fact(int n)
```

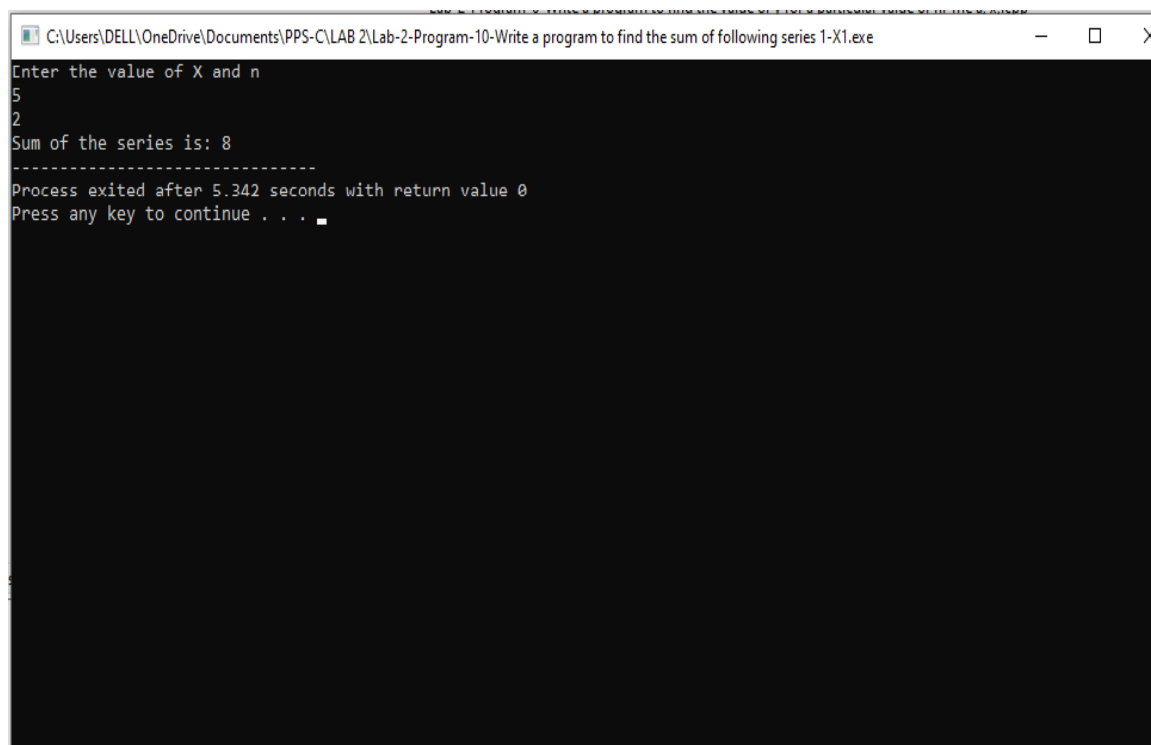
```
{
    int i,f=1;
    for(i=1;i<=n;i++)
        f=f*i;
    return f;
}
```

```
// driver function
```

```
int main()
```

```
{
    int n,sum=1,i,f,X,p;
    printf("Enter the value of X and n\n");
    scanf("%d%d",&X,&n);
    for(i=1;i<=n;i++)
    {
        f=fact(i);
        p=pow(X,i);
        if(i%2==0)
            sum=sum+(p/f);
        else
            sum=sum-(p/f);
    }
    printf("Sum of the series is: %d",sum);
    return 0;
}
```

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 2\Lab-2-Program-10-Write a program to find the sum of following series 1-X1.exe
Enter the value of X and n
5
2
Sum of the series is: 8
-----
Process exited after 5.342 seconds with return value 0
Press any key to continue . . .
```

Lab-3

Program-11: Write a program to print the entire prime no between 1 and 300.

Code:

```
//Write a program to print the entire prime no between 1 and 300.
#include<stdio.h>
#include<conio.h>
int main()
{
    int num, max=300, i, flag;

    //create a for loop to run a counter from 1 to 300.

    for(num=1;num<=max;num++)
    {
        flag=0;
        //second for loop to check the number from 1 to 300 is prime or not.
        for(i=2;i<=num/2;i++)
        {
            if(num%i==0)
            {
                flag=1;
                break;
            }
        }

        if(flag==0 & num!=1) // we remove 1 from this condition
            printf("%d\t", num);
    }
}
```

Output:

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 3\Lab-3-Program-11-Write a program to print the entire prime no between 1 and 300.exe
2  3   5   7   11  13  17  19  23  29  31  37  41  43  47
   53  59  61  67  71  73  79  83  89  97  101 103 107 109
  113 127 131 137 139 149 151 157 163 167 173 179 181 191
  193 197 199 211 223 227 229 233 239 241 251 257 263 269
  271 277 281 283 293
-----
Process exited after 0.0722 seconds with return value 0
Press any key to continue . . .
```


Program-12: Write a program to print out all the Armstrong number between 100 and 500.

Code:

```
//Write a program to print out all the Armstrong number between 100 and 500.
```

```
// USING FUNCTION
```

```
#include<stdio.h>
```

```
#include<conio.h>
```

```
// function for check Armstrong
```

```
int checkArm(int n)
{
    int n1,n2,r,s=1,sum=0; //r=reminder
    n1=n;
    n2=n;
    while(n!=0)
    {
        r=n%10;
        n=n/10;
        while(n1!=0)
        {
            s=s*r;
            n1=n1/10;
        }
        sum=sum+s;
        s=1;
        n1=n2;
    }
    if(sum==n2)
        return 1;
    else
        return 0;
}
```

```
// Driver function
```

```
int main()
{
    int i,a;
    for(i=100;i<=500;i++)
    {
        a=checkArm(i);
        if(a==1)
            printf("%d ",i);
    }
    return 0;
}
```

Output:

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 3\Lab-3-Program-12-Write a program to print out all the Armstrong number between 100 and.exe
153 370 371 407
-----
Process exited after 0.00068 seconds with return value 0
Press any key to continue . . .
```

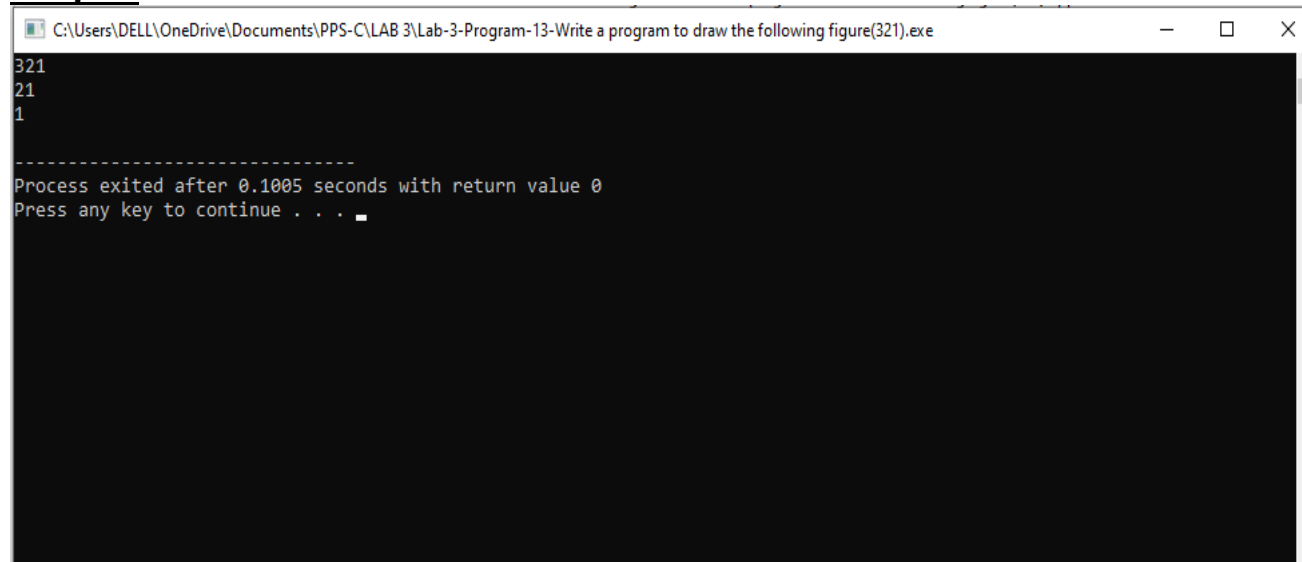
Program-13: Write a program to draw the following figure: 3 2 1 21 1 * ** ***

Code:

```
/*  
Write a program to draw the following figure:  
3 2 1  
21  
1  
*  
**  
***  
*/  
#include<stdio.h>
```

```
int main()  
{  
    int i,j,x;  
    for(i=3;i>=1;i--)  
    {  
        x=i;  
        for(j=1;j<=i;j++)  
        {  
            printf("%d",x);  
            x--;  
        }  
        printf("\n");  
    }  
    return 0;  
}
```

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 3\Lab-3-Program-13-Write a program to draw the following figure(321).exe  
321  
21  
1  
-----  
Process exited after 0.1005 seconds with return value 0  
Press any key to continue . . .
```

Code:

```
/*  
Write a program to draw the following figure:  
3 2 1  
21
```

```

1
*
**
***
*/
#include<stdio.h>

int main()
{
    int i,j;
    for(i=1;i<=3;i++)
    {
        for(j=1;j<=i;j++)
        {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}

```

Output:

```

**
**
***
-----
Process exited after 0.08948 seconds with return value 0
Press any key to continue . . .

```

Program-14: Write a program to receive a five-digit no and display as like 24689: 2 4 6 8 9

Code:

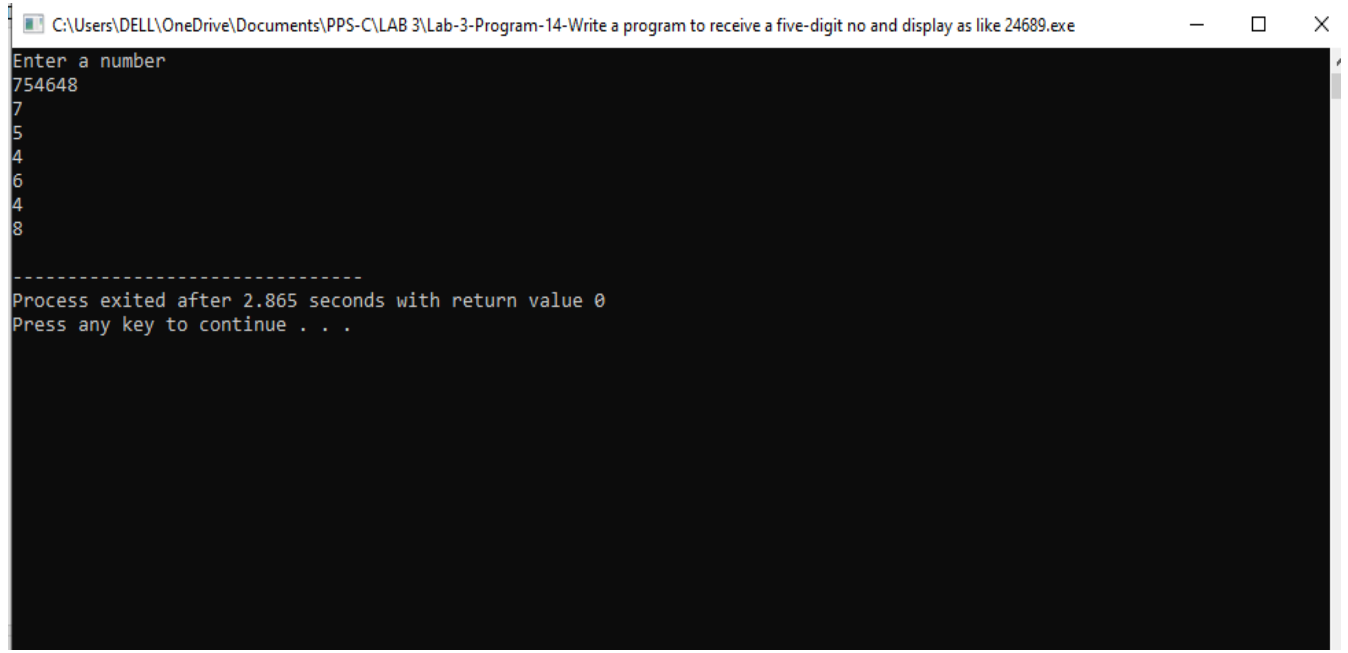
/*Write a program to receive a five-digit no and display as like 24689:

```
2
4
6
8
9*/
#include<stdio.h>

int main()
{
    int n,r,rev=0;
    printf("Enter a number\n");
    scanf("%d",&n);
    // finding reverse
    while(n!=0)
    {
        r=n%10;
        rev=rev*10+r;
        n=n/10;
    }

    // printing of number
    while(rev!=0)
    {
        r=rev%10;
        rev=rev/10;
        printf("%d\n",r);
    }
    return 0;
}
```

Output:



A screenshot of a Windows command prompt window. The title bar at the top reads "C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 3\Lab-3-Program-14-Write a program to receive a five-digit no and display as like 24689.exe". The command prompt shows the prompt "Enter a number" followed by the input "754648". Below the input, the digits "7", "5", "4", "6", "4", and "8" are displayed on separate lines. A dashed line separates this from the exit message: "Process exited after 2.865 seconds with return value 0". The prompt then shows "Press any key to continue . . .".

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 3\Lab-3-Program-14-Write a program to receive a five-digit no and display as like 24689.exe
Enter a number
754648
7
5
4
6
4
8
-----
Process exited after 2.865 seconds with return value 0
Press any key to continue . . .
```

Lab-4

Program-15: Write a function that return sum of all the odd digits of a given positive no entered through keyboard.

Code:

//Write a function that return sum of all the odd digits of a given positive no entered through keyboard.

```
#include<stdio.h>
```

```
// function for add digits
```

```
int Add_odd(int n)
```

```
{
    int r,sum=0;
    while(n!=0)
    {
        r=n%10;
        if(r%2!=0)
            sum=sum+r;
        n=n/10;
    }
    return sum;
}
```

```
//Driver function
```

```
int main()
```

```
{
    int n;
    printf("Enter a number\n");
    scanf("%d",&n);
    printf("Sum of all the odd digits is: %d",Add_odd(n));
    return 0;
}
```

Output:

C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 4\Lab-4-Program-15-Write a function that return sum of all the odd digits of a given positive no.exe

```
Enter a number
45663639
Sum of all the odd digits is: 20
-----
Process exited after 4.27 seconds with return value 0
Press any key to continue . . .
```


Program-16: Write a program to print area of rectangle using function & return its value to main function.

Code:

//Write a program to print area of rectangle using function & return its value to main function.

```
#include<stdio.h>
```

```
// function for calculating the area of rectangle
```

```
int Area(int l,int b)
```

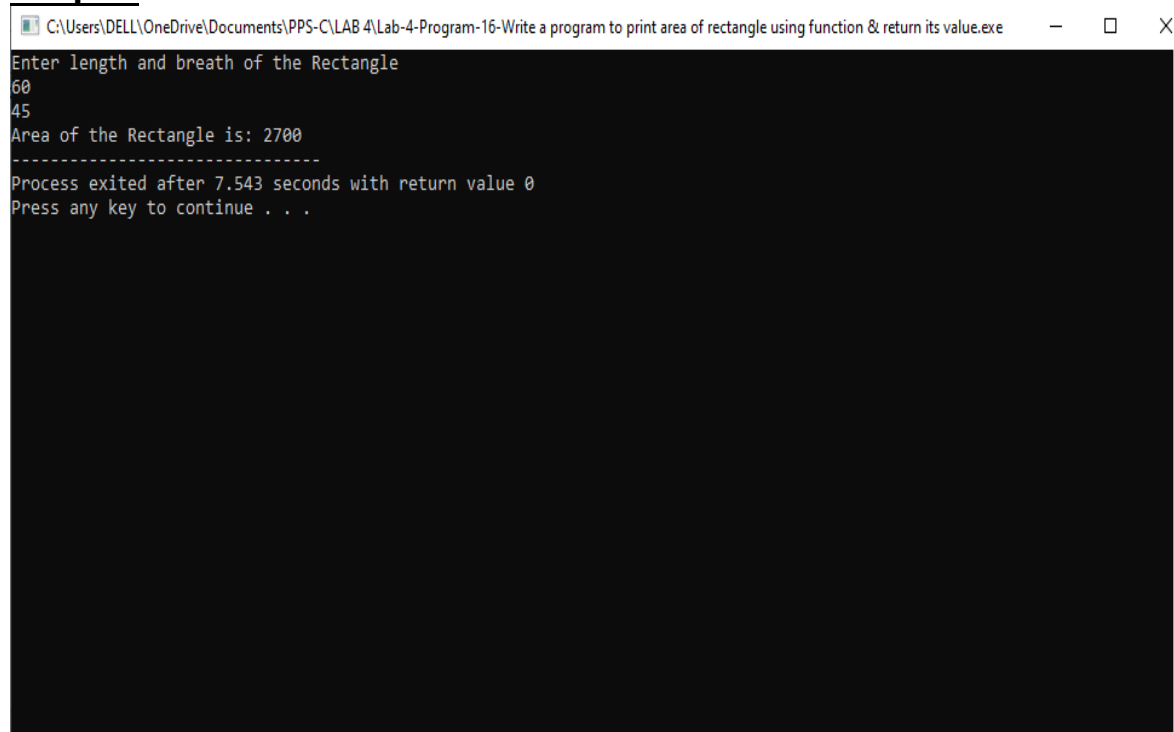
```
{  
    int a;  
    a=l*b;  
    return a;  
}
```

```
//Driver function
```

```
int main()  
{
```

```
    int length,breath,A;  
    printf("Enter length and breath of the Rectangle\n");  
    scanf("%d%d",&length,&breath);  
    A=Area(length,breath); // function calculating  
    printf("Area of the Rectangle is: %d",A);  
    return 0;  
}
```

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 4\Lab-4-Program-16-Write a program to print area of rectangle using function & return its value.exe  
Enter length and breath of the Rectangle  
60  
45  
Area of the Rectangle is: 2700  
-----  
Process exited after 7.543 seconds with return value 0  
Press any key to continue . . .
```

Program-17: Write a program to calculate the factorial for given number using function.

Code:

//Write a program to calculate the factorial for given number using function.

```
#include<stdio.h>
```

```
void factorial(int);
```

```
int main()
```

```
{  
    int num;
```

```
    printf("Enter a positive number to find Factorial\n");  
    scanf("%d", &num);
```

```
    factorial(num);
```

```
    return 0;
```

```
}
```

```
void factorial(int num)
```

```
{  
    int count, fact = 1;
```

```
    if(num == 0)  
    {  
        printf("Factorial of 0 is 1 (!0 = 1)\n");  
    }
```

```
    else
```

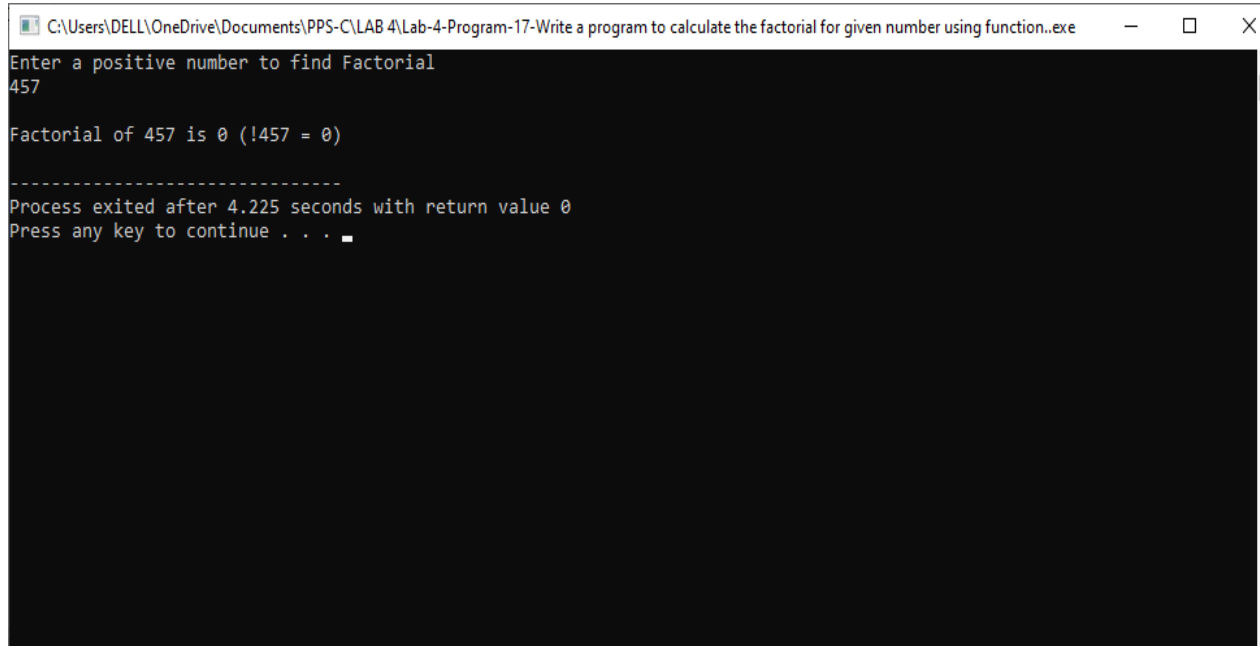
```
    {  
        for(count = 1; count <= num; count++)  
        {  
            fact = fact * count;
```

```
        printf("\nFactorial of %d is %d (!%d = %d)\n", num, fact, num, fact);
```

```
    }
```

```
}
```

Output:



A screenshot of a Windows command prompt window. The title bar shows the file path: C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 4\Lab-4-Program-17-Write a program to calculate the factorial for given number using function..exe. The window has standard minimize, maximize, and close buttons. The command prompt shows the following text: "Enter a positive number to find Factorial", followed by the input "457". The output is "Factorial of 457 is 0 (!457 = 0)". Below this, a separator line of dashes is shown. The text continues with "Process exited after 4.225 seconds with return value 0" and "Press any key to continue . . .", with a cursor visible at the end of the second line.

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 4\Lab-4-Program-17-Write a program to calculate the factorial for given number using function..exe
Enter a positive number to find Factorial
457

Factorial of 457 is 0 (!457 = 0)

-----
Process exited after 4.225 seconds with return value 0
Press any key to continue . . .
```

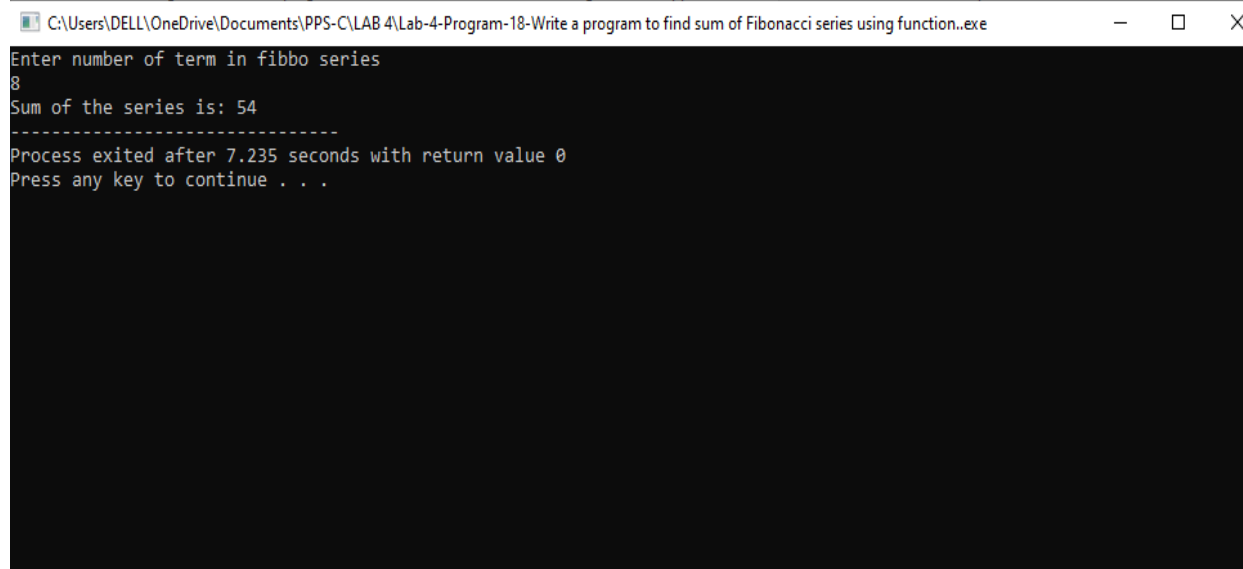
Program-18: Write a program to find sum of Fibonacci series using function.

Code:

//Write a program to find sum of Fibonacci series using function.

```
#include<stdio.h>
int fib(int n)
{
    if(n==1||n==2)
        return 1;
    return (fib(n-1)+fib(n-2));
}
int main()
{
    int n,i,sum=0;
    printf("Enter number of term in fibbo series\n");
    scanf("%d",&n);
    for(i=1;i<=n;i++)
    {
        sum=sum+fib(i);
    }
    printf("Sum of the series is: %d",sum);
    return 0;
}
```

Output:

A screenshot of a Windows command prompt window. The title bar shows the file path: C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 4\Lab-4-Program-18-Write a program to find sum of Fibonacci series using function..exe. The window has standard minimize, maximize, and close buttons. The command prompt shows the following text: "Enter number of term in fibbo series", followed by the input "8", then "Sum of the series is: 54". Below this, there is a dashed line separator, followed by "Process exited after 7.235 seconds with return value 0" and "Press any key to continue . . .".

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 4\Lab-4-Program-18-Write a program to find sum of Fibonacci series using function..exe
Enter number of term in fibbo series
8
Sum of the series is: 54
-----
Process exited after 7.235 seconds with return value 0
Press any key to continue . . .
```

Program-19: Write factorial function & use the function to find the sum of series $S=1!+2!+\dots+n!$.

Code:

//Write factorial function & use the function to find the sum of series $S=1!+2!+\dots+n!$.

```
#include <stdio.h>
```

```
int factorial (int);
```

```
int main (void)
```

```
{
```

```
    int terms;
```

```
    int sum;
```

```
    int i;
```

```
    //Input number of terms
```

```
    printf ("Input number of terms: ");
```

```
    scanf ("%d", &terms);
```

```
    //Calculate sum of series
```

```
    sum = 0;
```

```
    for (i = 1; i <= terms; i++)
```

```
        sum = sum + factorial (i);
```

```
    //Print result
```

```
    printf ("Sum of Series (S = 1! + 2! ... + %d!): %d", terms, sum);
```

```
    return 0;
```

```
};
```

```
int factorial (int num)
```

```
{
```

```
    int fact;
```

```
    int i;
```

```
    //Calculate factorial
```

```
    fact = 1;
```

```
    for (i = 1; i <= num; i++)
```

```
        fact = fact * i;
```

```
    //Return factorial
```

```
    return fact;
```

```
};
```

Output:

C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 4\Lab-4-Program-19-Write factorial function & use the function to find the sum of series.exe

```
Input number of terms: 76
Sum of Series (S = 1! + 2! ... + 76!): -125961703
-----
Process exited after 3.802 seconds with return value 0
Press any key to continue . . .
```

Lab-5

Program-20: Write a program to find the factorial of given number using recursion.

Code:

//Write a program to find the factorial of given number using recursion.

```
#include<stdio.h>
```

```
long int multiplyNumbers(int n);
```

```
int main() {
```

```
    int n;
```

```
    printf("Enter a positive integer: ");
```

```
    scanf("%d",&n);
```

```
    printf("Factorial of %d = %ld", n, multiplyNumbers(n));
```

```
    return 0;
```

```
}
```

```
long int multiplyNumbers(int n) {
```

```
    if (n>=1)
```

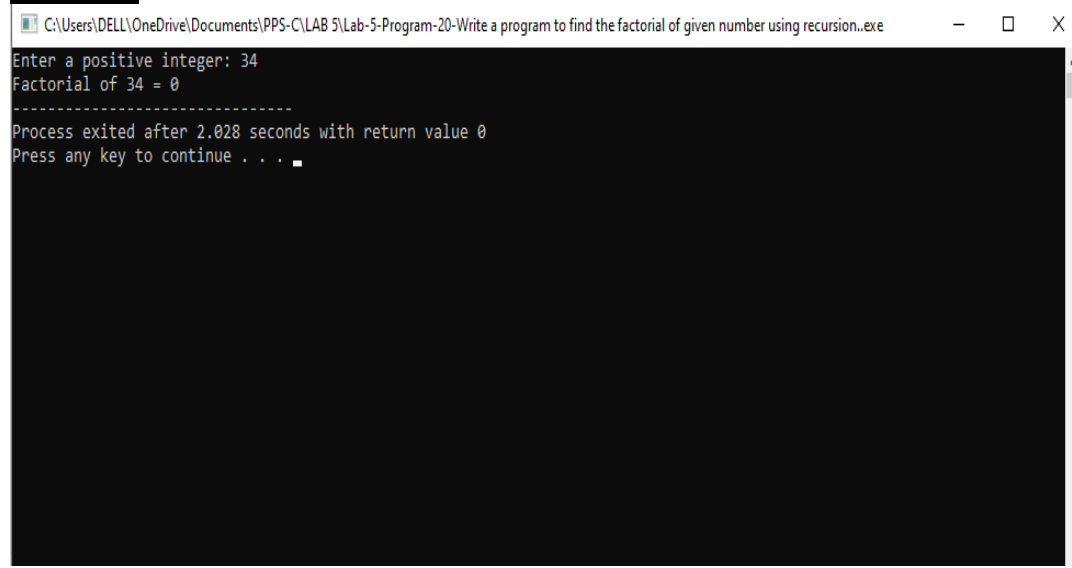
```
        return n*multiplyNumbers(n-1);
```

```
    else
```

```
        return 1;
```

```
}
```

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 5\Lab-5-Program-20-Write a program to find the factorial of given number using recursion..exe
Enter a positive integer: 34
Factorial of 34 = 0
-----
Process exited after 2.028 seconds with return value 0
Press any key to continue . . .
```

Program-21: Write a program to find the sum of digits of a 5 digit number using recursion.

Code:

//Write a program to find the sum of digits of a 5 digit number using recursion.

// Recursive C program to find sum of digits

// of a number

#include <stdio.h>

// Function to check sum of digit using recursion

int sum_of_digit(int n)

{

 if (n == 0)

 return 0;

 return (n % 10 + sum_of_digit(n / 10));

}

// Driven Program to check above

int main()

{

 int num = 12345;

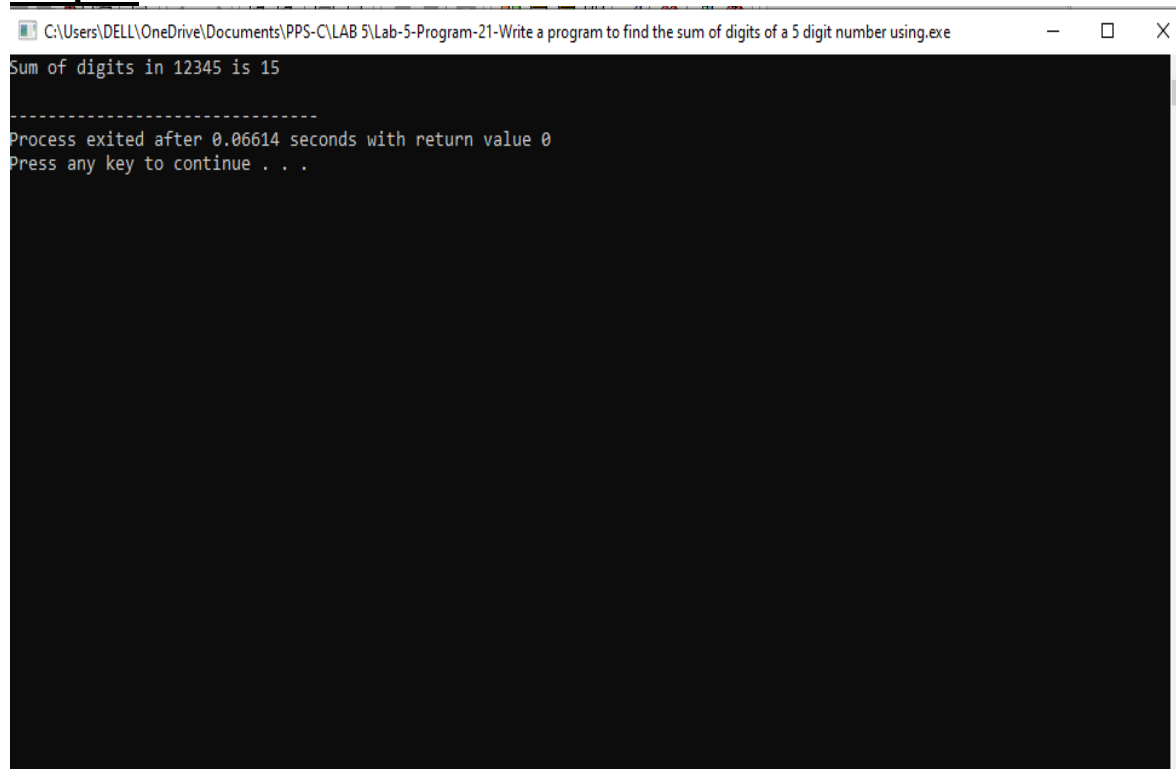
 int result = sum_of_digit(num);

 printf("Sum of digits in %d is %d\n", num, result);

 return 0;

}

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 5\Lab-5-Program-21-Write a program to find the sum of digits of a 5 digit number using.exe
Sum of digits in 12345 is 15
-----
Process exited after 0.06614 seconds with return value 0
Press any key to continue . . .
```


Program-22: Write a program to calculate the GCD of given numbers using recursion.

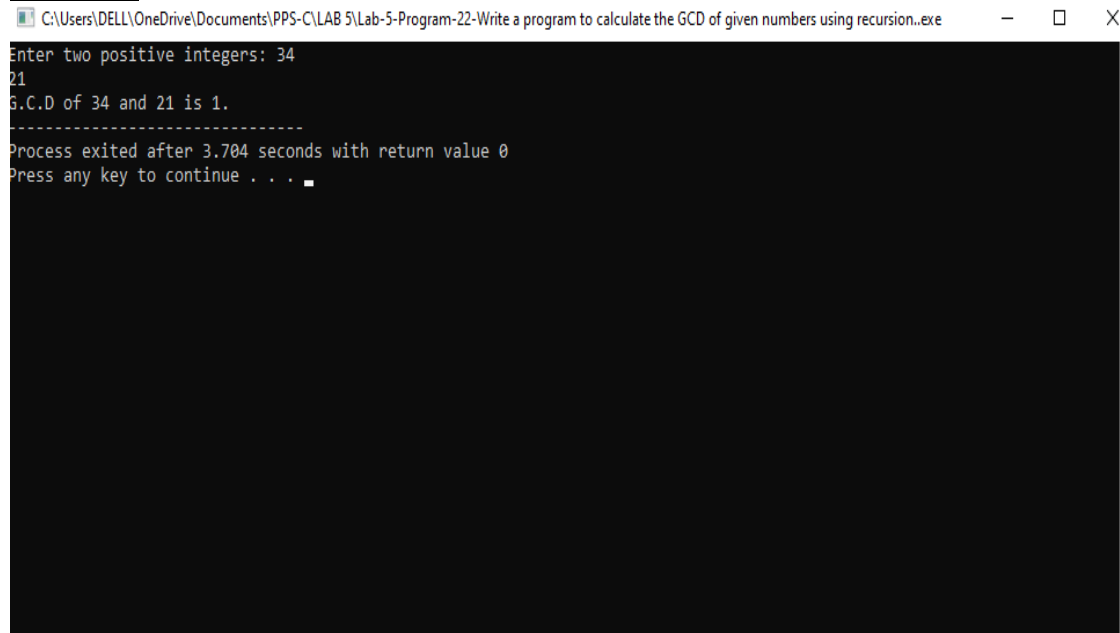
Code:

//Write a program to calculate the GCD of given numbers using recursion.

```
#include <stdio.h>
int hcf(int n1, int n2);
int main() {
    int n1, n2;
    printf("Enter two positive integers: ");
    scanf("%d %d", &n1, &n2);
    printf("G.C.D of %d and %d is %d.", n1, n2, hcf(n1, n2));
    return 0;
}

int hcf(int n1, int n2) {
    if (n2 != 0)
        return hcf(n2, n1 % n2);
    else
        return n1;
}
```

Output:



C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 5\Lab-5-Program-22-Write a program to calculate the GCD of given numbers using recursion..exe

```
Enter two positive integers: 34
21
G.C.D of 34 and 21 is 1.
-----
Process exited after 3.704 seconds with return value 0
Press any key to continue . . .
```

Program-23: Write a program to convert decimal number in to binary number.

Code:

```
//Write a program to convert decimal number in to binary number.
```

```
// convert decimal to binary
```

```
#include <stdio.h>
```

```
#include <math.h>
```

```
long long convert(int);
```

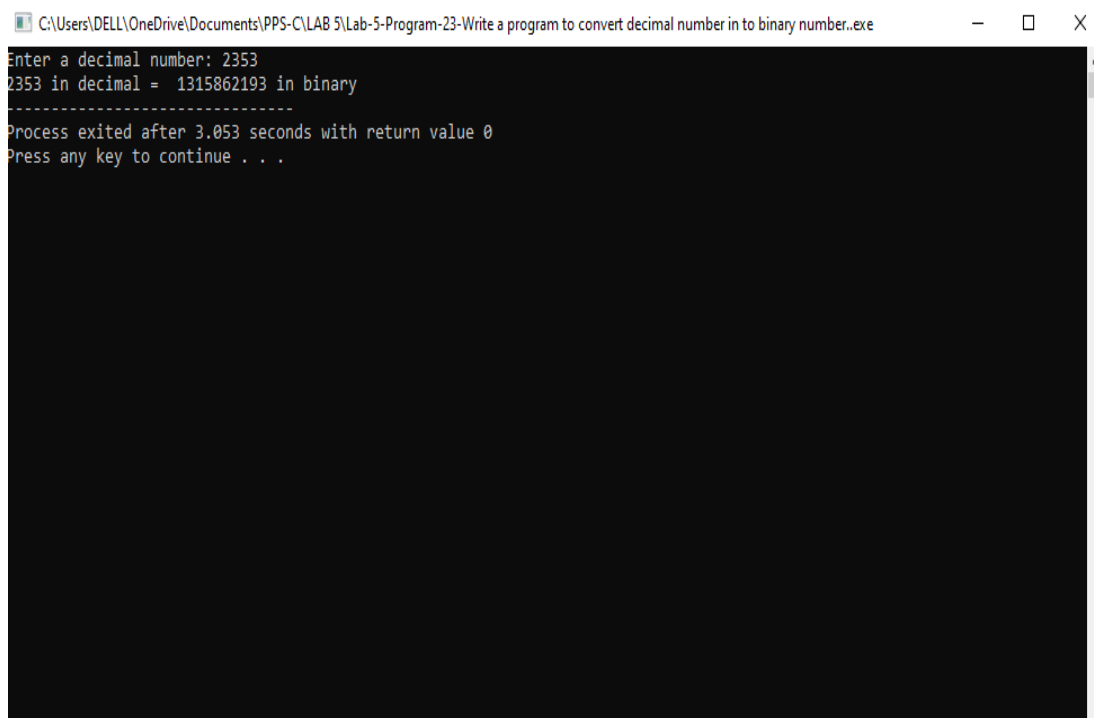
```
int main() {  
    int n, bin;  
    printf("Enter a decimal number: ");  
    scanf("%d", &n);  
    bin = convert(n);  
    printf("%d in decimal = %lld in binary", n, bin);  
    return 0;  
}
```

```
long long convert(int n) {  
    long long bin = 0;  
    int rem, i = 1;
```

```
    while (n!=0) {  
        rem = n % 2;  
        n /= 2;  
        bin += rem * i;  
        i *= 10;  
    }
```

```
    return bin;  
}
```

Output:



A screenshot of a Windows command prompt window. The title bar at the top reads "C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 5\Lab-5-Program-23-Write a program to convert decimal number in to binary number..exe". The command prompt shows the following text: "Enter a decimal number: 2353", "2353 in decimal = 1315862193 in binary", a separator line of dashes, "Process exited after 3.053 seconds with return value 0", and "Press any key to continue . . .". The rest of the window is black.

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 5\Lab-5-Program-23-Write a program to convert decimal number in to binary number..exe
Enter a decimal number: 2353
2353 in decimal = 1315862193 in binary
-----
Process exited after 3.053 seconds with return value 0
Press any key to continue . . .
```

Program-24: Write a program to convert binary number in to decimal number.

Code:

```
//Write a program to convert binary number in to decimal number.
```

```
// convert binary to decimal
```

```
#include <stdio.h>
```

```
#include <math.h>
```

```
// function prototype
```

```
int convert(long long);
```

```
int main() {
```

```
    long long n;
```

```
    printf("Enter a binary number: ");
```

```
    scanf("%lld", &n);
```

```
    printf("%lld in binary = %d in decimal", n, convert(n));
```

```
    return 0;
```

```
}
```

```
// function definition
```

```
int convert(long long n) {
```

```
    int dec = 0, i = 0, rem;
```

```
    while (n!=0) {
```

```
        rem = n % 10;
```

```
        n /= 10;
```

```
        dec += rem * pow(2, i);
```

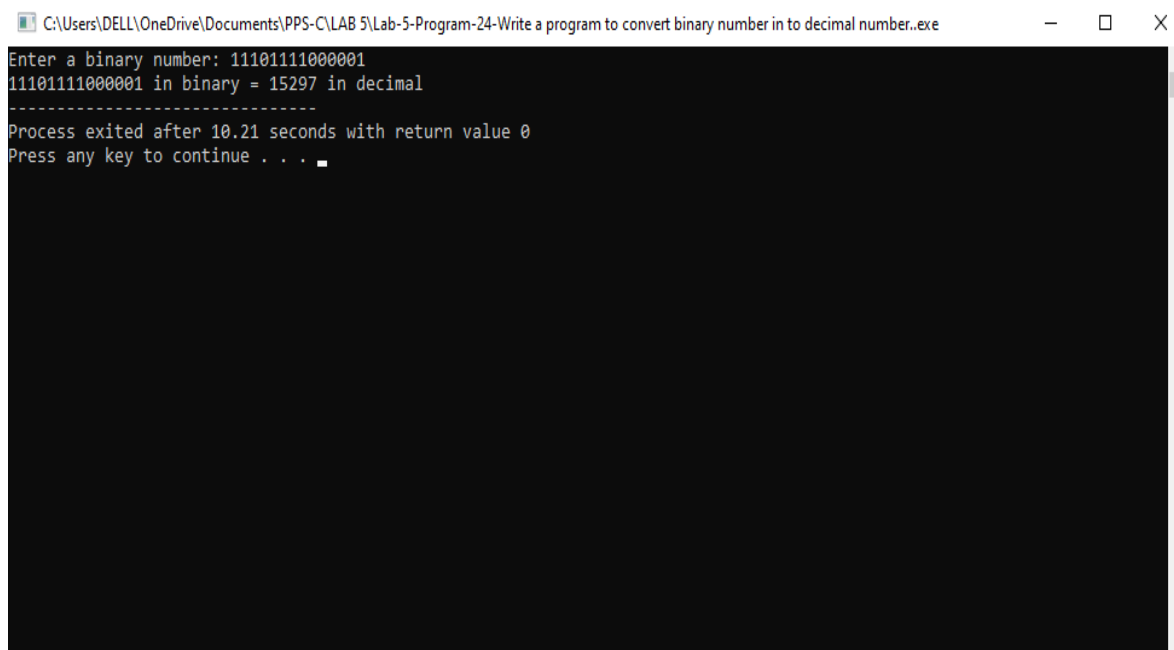
```
        ++i;
```

```
    }
```

```
    return dec;
```

```
}
```

Output:



A screenshot of a Windows command prompt window. The title bar at the top reads "C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 5\Lab-5-Program-24-Write a program to convert binary number in to decimal number..exe". The command prompt shows the following text: "Enter a binary number: 11101111000001", "11101111000001 in binary = 15297 in decimal", a dashed line separator, "Process exited after 10.21 seconds with return value 0", and "Press any key to continue . . .". The rest of the window is black.

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 5\Lab-5-Program-24-Write a program to convert binary number in to decimal number..exe
Enter a binary number: 11101111000001
11101111000001 in binary = 15297 in decimal
-----
Process exited after 10.21 seconds with return value 0
Press any key to continue . . .
```

Lab-6

Program-25: Write a program to delete duplicate element in a list of 10 elements & display it on screen.

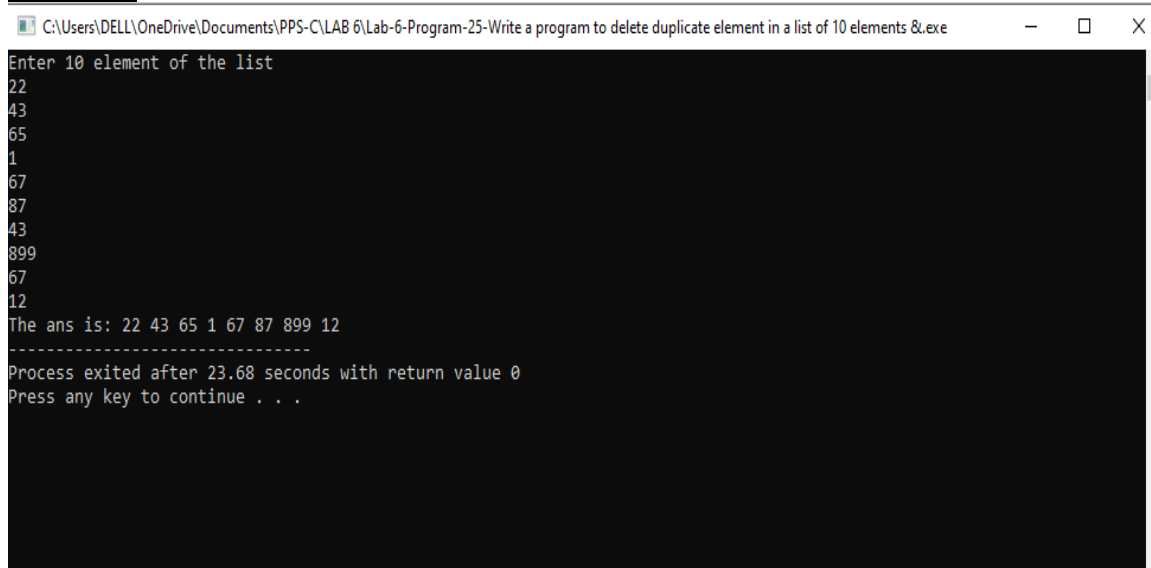
Code:

//Write a program to delete duplicate element in a list of 10 elements & display it on screen.

```
#include<stdio.h>
int main()
{
    int a[10],b[10],i,j,k=0;
    printf("Enter 10 element of the list\n");
    for(i=0;i<10;i++)
        scanf("%d",&a[i]);

    // logic for deleting duplicates
    for(i=0;i<10;i++)
    {
        for(j=0;j<k;j++)
        {
            if(b[j]==a[i])
                break;
        }
        if(j==k)
        {
            b[k]=a[i];
            k++;
        }
    }
    printf("The ans is: ");
    for(i=0;i<k;i++)
        printf("%d ",b[i]);
    return 0;
}
```

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 6\Lab-6-Program-25-Write a program to delete duplicate element in a list of 10 elements &.exe
Enter 10 element of the list
22
43
65
1
67
87
43
899
67
12
The ans is: 22 43 65 1 67 87 899 12
-----
Process exited after 23.68 seconds with return value 0
Press any key to continue . . .
```

Program-26: Write a program to merge two sorted array & no element is repeated during merging.

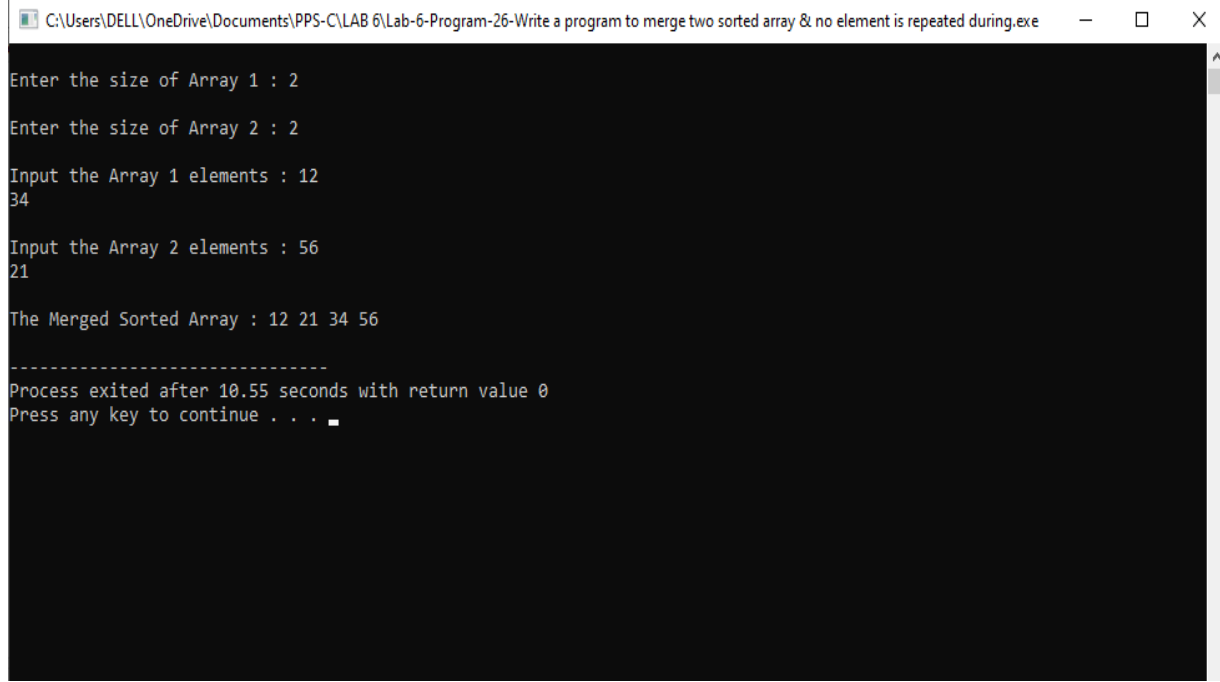
Code:

//Write a program to merge two sorted array & no element is repeated during merging.

```
#include <stdio.h>
#include<stdlib.h>
int merge_arrays(int arr1[], int arr2[], int arr3[], int m, int n)
{
    int i,j;
    for(i = 0; i < m; i++)
    {
        arr3[i] = arr1[i];
    }
    for(i = m, j = 0 ; i < m + n; i++, j++)
    {
        arr3[i] = arr2[j];
    }
}
int main()
{
    int n,m;
    printf("\nEnter the size of Array 1 : ");
    scanf("%d",&m);
    printf("\nEnter the size of Array 2 : ");
    scanf("%d",&n);
    int arr1[m],arr2[n];
    int arr3[m+n];
    int i;
    printf("\nInput the Array 1 elements : ");
    for(i = 0; i<m;i++)
    {
        scanf("%d",&arr1[i]);
    }
    printf("\nInput the Array 2 elements : ");
    for(i = 0;i<n;i++)
    {
        scanf("%d",&arr2[i]);
    }
    merge_arrays(arr1,arr2,arr3,m,n);
    printf("\nThe Merged Sorted Array : ");
    for(i = 0; i < m+n-1; i++)
    {
        for(int j = 0; j < m+n-i-1; j++)
        {
            if(arr3[j] > arr3[j + 1])
            {
                int temp = arr3[j];
                arr3[j ] = arr3[j + 1];
                arr3[j + 1] = temp;
            }
        }
    }
}
```

```
}  
}  
for(i = 0; i < n + m; i++)  
{  
printf("%d ",arr3[i]);  
}  
printf("\n");  
return 0;  
}
```

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 6\Lab-6-Program-26-Write a program to merge two sorted array & no element is repeated during.exe  
Enter the size of Array 1 : 2  
Enter the size of Array 2 : 2  
Input the Array 1 elements : 12  
34  
Input the Array 2 elements : 56  
21  
The Merged Sorted Array : 12 21 34 56  
-----  
Process exited after 10.55 seconds with return value 0  
Press any key to continue . . .
```


Program-27: Write a program to evaluate the addition of diagonal elements of two square matrixes.

Code:

//Write a program to evaluate the addition of diagonal elements of two square matrixes.

```
#include<stdio.h>
int main()
{
    int a[10][10],b[10][10],i,j,n1,n2,sum1=0,sum2=0;
    printf("Enter size of both array\n");
    scanf("%d%d",&n1,&n2);
    printf("Enter elements of the first matrix\n");
    for(i=0;i<n1;i++)
    {
        for(j=0;j<n1;j++)
            scanf("%d",&a[i][j]);
    }
    printf("Enter elements of the second matrix\n");
    for(i=0;i<n2;i++)
    {
        for(j=0;j<n2;j++)
            scanf("%d",&b[i][j]);
    }

    // logic for calculating sum of digonal of first matrix
    for(i=0;i<n1;i++)
    {
        for(j=0;j<n1;j++)
        {
            if(j==i)
                sum1=sum1+a[i][j];
        }
    }

    // logic for calculating sum of digonal of second matrix
    for(i=0;i<n2;i++)
    {
        for(j=0;j<n2;j++)
        {
            if(j==i)
                sum2=sum2+b[i][j];
        }
    }

    printf("Sum of digonal elements of the first matrix is: %d\n",sum1);
    printf("Sum of digonal elements of the second matrix is: %d\n",sum2);
    printf("Total Sum of digonal elements of the both matrix is: %d",sum1+sum2);
    return 0;
}
```

Output:

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 6\Lab-6-Program-27-Write a program to evaluate the addition of diagonal elements of two square ma...
Enter size of both array
2
2
Enter elements of the first matrix
23
2
45
7
Enter elements of the second matrix
21
67
89
1
Sum of digonal elements of the first matrix is: 30
Sum of digonal elements of the second matrix is: 22
Total Sum of digonal elements of the both matrix is: 52
-----
Process exited after 59.65 seconds with return value 0
Press any key to continue . . .
```

Program-28: Write a program to find the transpose of a given matrix & check whether it is symmetric or not.

Code:

//Write a program to find the transpose of a given matrix & check whether it is symmetric or not.

```
#include<stdio.h>
int main()
{
    int a[10][10],t[10][10],i,j,flag=0;
    int r,c;
    printf("Enter the number of row and column of the matrix\n");
    scanf("%d%d",&r,&c);
    printf("Enter the element of the matrix\n");
    // Input matrix a[][]
    for(i=0;i<r;i++)
    {
        for(j=0;j<c;j++)
            scanf("%d",&a[i][j]);
    }

    // logic for compute transpose
    for(i=0;i<c;i++)
    {
        for(j=0;j<r;j++)
        {
            t[i][j]=a[j][i];
        }
    }

    // Display the transpose
    printf("the transpose is:-\n");
    for(i=0;i<c;i++)
    {
        for(j=0;j<r;j++)
            printf("%d ",t[i][j]);
        printf("\n");
    }

    //logic for checking matrix is symmetric or not
    if(r!=c)
        printf("Matrix is not symmetric\n");
    else
    {
        for(i=0;i<r;i++)
        {
            for(j=0;j<c;j++)
            {
                if(t[i][j]!=a[i][j]);
                {
                    flag=1;
                    break;
                }
            }
        }
    }
}
```

```

        }
    }
    if(flag==1)
        break;

}
if(flag==0)
    printf("Matrix is symmetric Matrix\n");
else
    printf("Matrix is not a symmetric Matrix\n");
}
return 0;
}

```

Output:

```

C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 6\Lab-6-Program-28-Write a program to find the transpose of a given matrix & check whether it is sy...
Enter the number of row and column of the matrix
2
2
Enter the element of the matrix
21
11
134
2
the transpose is:-
21 134
11 2
Matrix is not a symmetric Matrix

-----
Process exited after 24.42 seconds with return value 0
Press any key to continue . . .

```

Program-29: Write a program to print the multiplication of two N*N (Square) matrix.

Code:

```
#include <stdio.h>

int main()
{
    int m, n, p, q, c, d, k, sum = 0;
    int first[10][10], second[10][10], multiply[10][10];

    printf("Enter the number of rows and columns of first matrix\n");
    scanf("%d%d", &m, &n);
    printf("Enter the elements of first matrix\n");

    for ( c = 0 ; c < m ; c++ )
        for ( d = 0 ; d < n ; d++ )
            scanf("%d", &first[c][d]);

    printf("Enter the number of rows and columns of second matrix\n");
    scanf("%d%d", &p, &q);

    if ( n != p )
        printf("Matrices with entered orders can't be multiplied with each other.\n");
    else
    {
        printf("Enter the elements of second matrix\n");

        for ( c = 0 ; c < p ; c++ )
            for ( d = 0 ; d < q ; d++ )
                scanf("%d", &second[c][d]);

        for ( c = 0 ; c < m ; c++ )
        {
            for ( d = 0 ; d < q ; d++ )
            {
                for ( k = 0 ; k < p ; k++ )
                {
                    sum = sum + first[c][k]*second[k][d];
                }

                multiply[c][d] = sum;
                sum = 0;
            }
        }

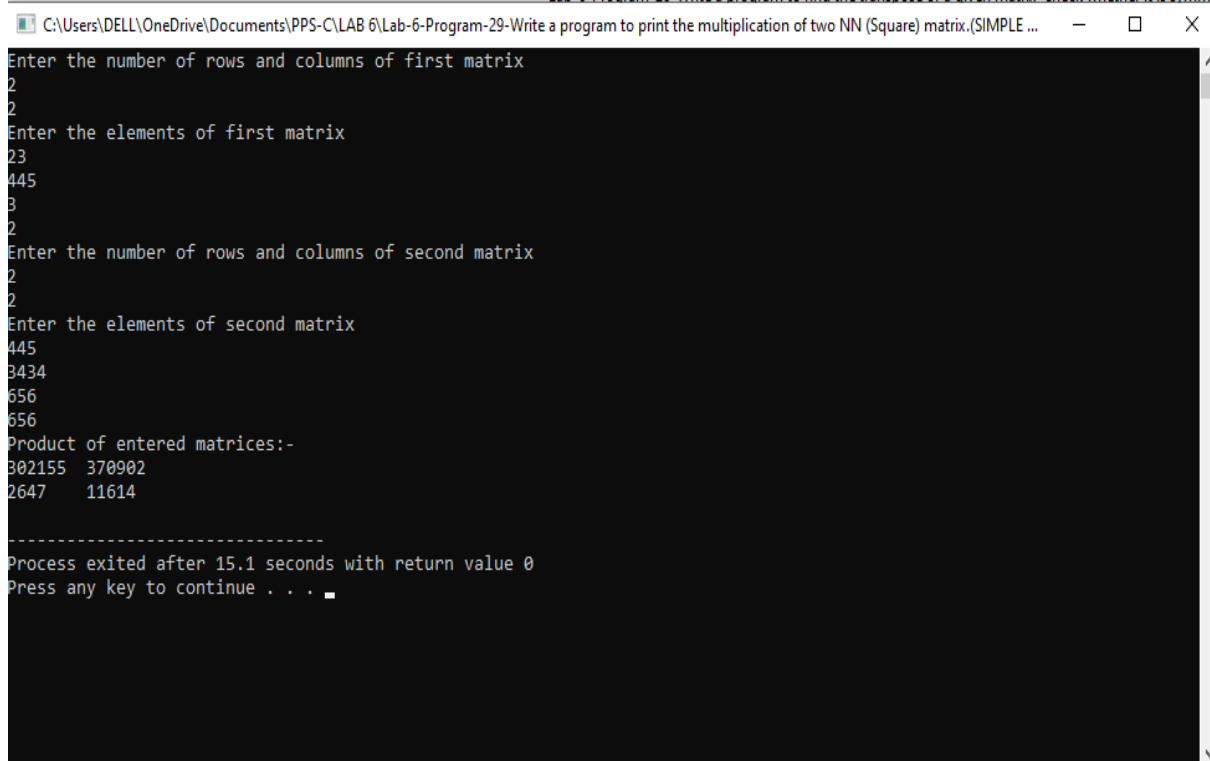
        printf("Product of entered matrices:-\n");

        for ( c = 0 ; c < m ; c++ )
        {
            for ( d = 0 ; d < q ; d++ )
                printf("%d\t", multiply[c][d]);
```

```
    printf("\n");  
}  
}
```

```
return 0;  
}
```

Output:



The screenshot shows a Windows command prompt window with the title bar "C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 6\Lab-6-Program-29-Write a program to print the multiplication of two NN (Square) matrix.(SIMPLE ...". The window contains the following text:

```
Enter the number of rows and columns of first matrix  
2  
2  
Enter the elements of first matrix  
23  
445  
3  
2  
Enter the number of rows and columns of second matrix  
2  
2  
Enter the elements of second matrix  
445  
3434  
656  
656  
Product of entered matrices:-  
302155 370902  
2647 11614  
  
-----  
Process exited after 15.1 seconds with return value 0  
Press any key to continue . . .
```

Lab-7

Program-30: Write a program in C to check whether the given string is a palindrome or not.

Code:

//Write a program in C to check whether the given string is a palindrome or not.

//Using Standard Method

#include <stdio.h>

#include <string.h>

```
int main()
{
    char s[1000];
    int i,n,c=0;

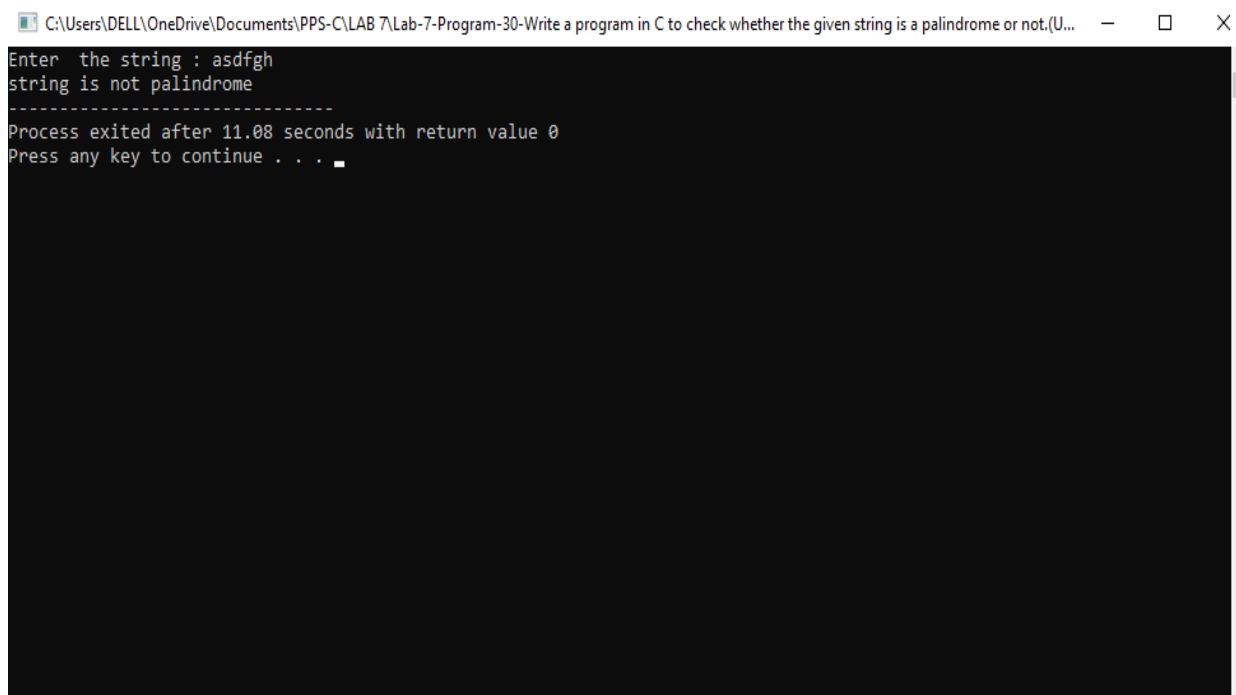
    printf("Enter the string : ");
    gets(s);
    n=strlen(s);

    for(i=0;i<n/2;i++)
    {
        if(s[i]==s[n-i-1])
            c++;

    }
    if(c==i)
        printf("string is palindrome");
    else
        printf("string is not palindrome");

    return 0;
}
```

Output:



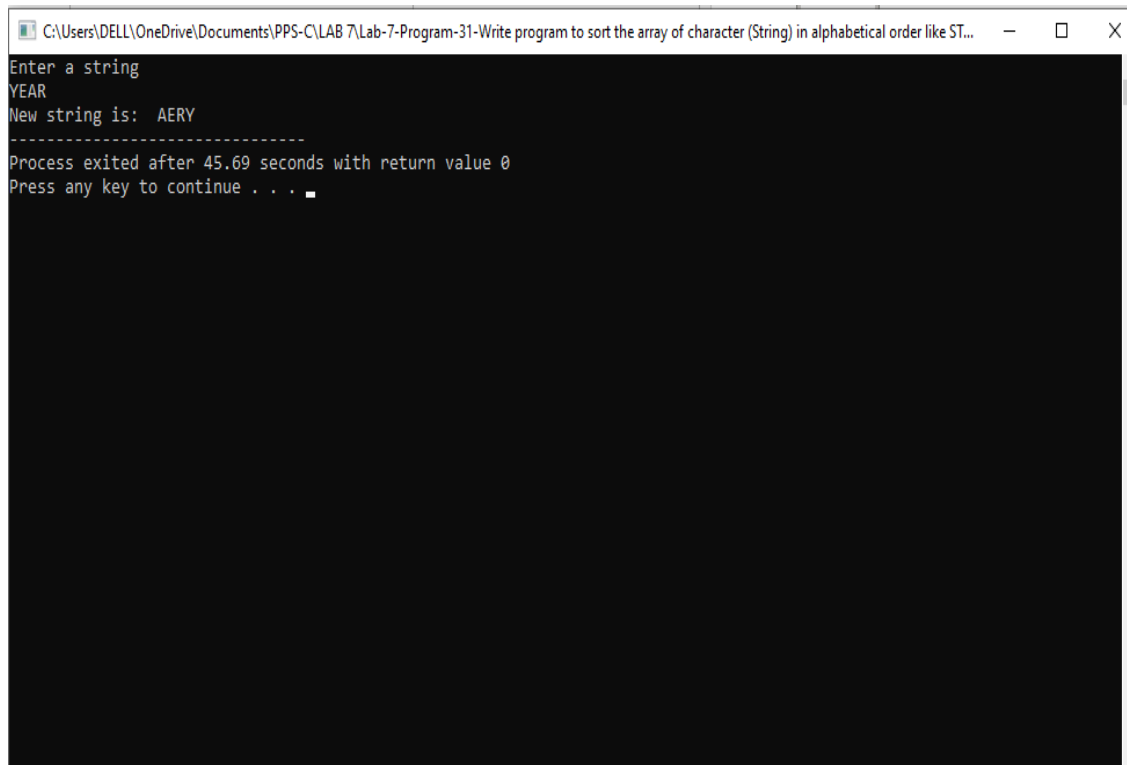
```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 7\Lab-7-Program-30-Write a program in C to check whether the given string is a palindrome or not.(U...
Enter the string : asdfgh
string is not palindrome
-----
Process exited after 11.08 seconds with return value 0
Press any key to continue . . .
```


Program-31: Write program to sort the array of character (String) in alphabetical order like STRING in GINRST.

Code:

```
#include<stdio.h>
//Write program to sort the array of character (String) in alphabetical order like STRING in GINRST.
#include<string.h>
int main()
{
    int i,x,j,k=0,len;
    char string[50],alpha[50];
    printf("Enter a string\n");
    scanf("%[^\\n]",string);
    len=strlen(string);
    x=65;
    while(x<=90 && k<len)
    {
        for(i=0;string[i];i++)
        {
            j=string[i];
            if(j==x || j==x+32)
            {
                alpha[k]=string[i];
                k++;
            }
        }
        x++;
    }
    alpha[k]='\0';
    printf("New string is:\t%s",alpha);
    return 0;
}
```

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 7\Lab-7-Program-31-Write program to sort the array of character (String) in alphabetical order like ST...
Enter a string
YEAR
New string is: AERY
-----
Process exited after 45.69 seconds with return value 0
Press any key to continue . . .
```

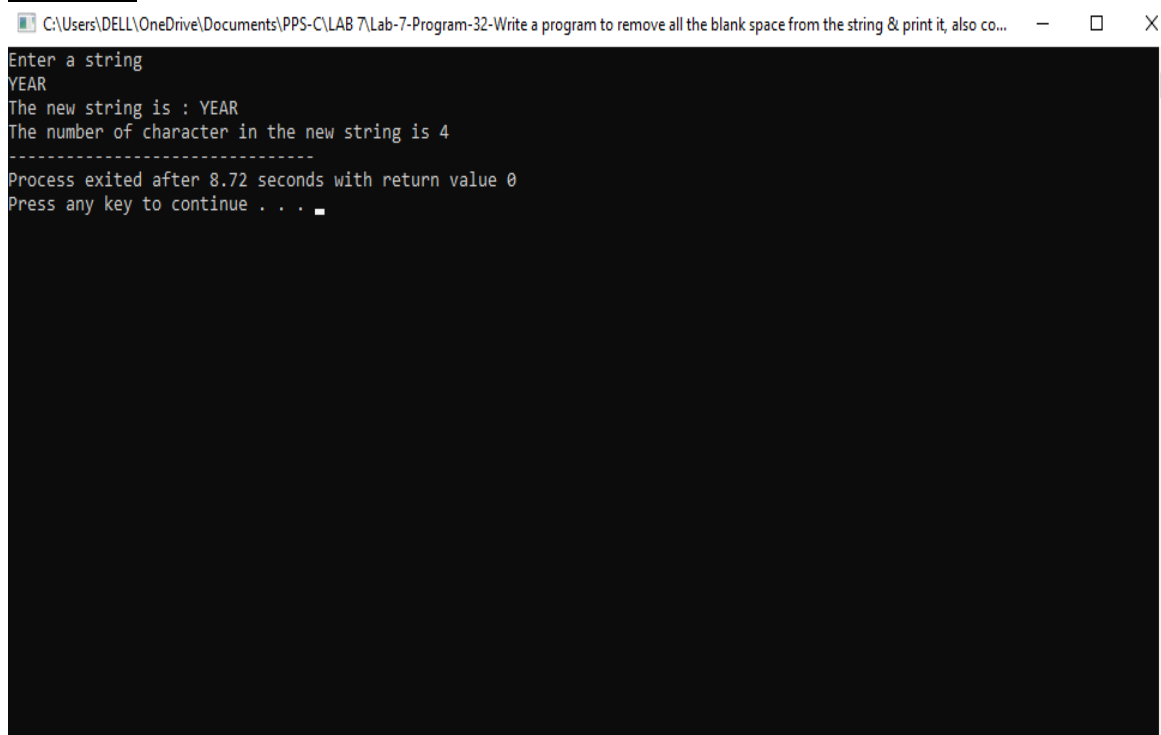
Program-32: Write a program to remove all the blank space from the string & print it, also count the no of characters.

Code:

//Write a program to remove all the blank space from the string & print it, also count the no of characters.

```
#include<stdio.h>
#include<string.h>
int main()
{
    char string[50],newS[50];
    int i,count,j=0;
    printf("Enter a string\n");
    scanf("%[^\n]",string);
    for(i=0;string[i];i++)
    {
        if(string[i]!=' ')
        {
            newS[j]=string[i];
            j++;
        }
    }
    newS[j]='\0'; // It is important to put null at
                // the end of string
    strcpy(string,newS); // copy new string into prev
    printf("The new string is : %s",string);
    count=strlen(string);
    printf("\nThe number of character in the new string is %d",count);
    return 0;
}
```

Output:



Program-33: Write a program to store the following string “zero”, “one” -----“five”. Print the no in words, given in figure as 3205.

Code:

//Write a program to store the following string “zero”, “one” -----“five”. Print the no in words, given in figure as 3205.

```
#include <stdio.h>
```

```
#include <string.h>
```

```
char* num_to_word (int);
```

```
int main (void)
```

```
{
```

```
    int num = 0;
```

```
    int temp;
```

```
    int digit;
```

```
    char digit_in_word[100];
```

```
    char num_in_words[500];
```

```
    //Empty string
```

```
    num_in_words[0] = '\0';
```

```
    //Input number
```

```
    printf ("Input number: ");
```

```
    scanf ("%d", &num);
```

```
    //Convert number to word
```

```
    temp = num;
```

```
    do
```

```
    {
```

```
        digit = temp % 10;
```

```
        switch (digit)
```

```
        {
```

```
            case 0:
```

```
                strcpy (digit_in_word, "Zero ");
```

```
                break;
```

```
            case 1:
```

```
                strcpy (digit_in_word, "One ");
```

```
                break;
```

```
            case 2:
```

```
                strcpy (digit_in_word, "Two ");
```

```
                break;
```

```
            case 3:
```

```
                strcpy (digit_in_word, "Three ");
```

```
                break;
```

```
            case 4:
```

```
                strcpy (digit_in_word, "Four ");
```

```
                break;
```

```
            case 5:
```

```
                strcpy (digit_in_word, "Five ");
```

```
                break;
```

```
            case 6:
```

```

        strcpy (digit_in_word, "Six ");
        break;
    case 7:
        strcpy (digit_in_word, "Seven ");
        break;
    case 8:
        strcpy (digit_in_word, "Eight ");
        break;
    case 9:
        strcpy (digit_in_word, "Nine ");
        break;
    }
    strcat (digit_in_word, num_in_words);
    strcpy (num_in_words, digit_in_word);
    temp = temp / 10;
}while (temp != 0);

//Print number in words
printf ("In words: %s", num_in_words);

return 0;
};

```

Output:

```

C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 7\Lab-7-Program-33-Write a program to store the following string ôzeroô, ôoneô -----ôfiveô. Print t...
Input number: 345
In words: Three Four Five
-----
Process exited after 3.476 seconds with return value 0
Press any key to continue . . .

```

Lab-8

Program-34: Write a program to compare two given dates. To store a date uses a structure that contains three members namely day, month and year.

If the dates are equal then display message equal otherwise unequal.

Code:

//Write a program to compare two given dates. To store a date uses a structure that contains three members namely day, month and year. If the dates are equal then display message equal otherwise unequal.

```
#include<stdio.h>
```

```
struct data
{
    int day,month,year;
}d1,d2;

int main()
{
    printf("Enter first date\n");
    scanf("%d%d%d",&d1.day,&d1.month,&d1.year);
    printf("Enter the second date\n");
    scanf("%d%d%d",&d2.day,&d2.month,&d2.year);
    if(d1.day==d2.day && d1.month==d2.month && d1.year==d2.year)
        printf("Equal\n");
    else
        printf("Unequal");
    return 0;
}
```

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 8\Lab-8-Program-34-Write a program to compare two given dates. To store a date uses a.exe
Enter first date
22
08
2002
Enter the second date
11
02
2002
Unequal
-----
Process exited after 18.85 seconds with return value 0
Press any key to continue . . .
```

Program-35: Define a structure that can describe a hotel. It should have the member that includes the name, address, grade, room charge and number of rooms.

Write a function to print out hotel of given grade in order of room charges.

Code:

/*Define a structure that can describe a hotel. It should have the member that includes the name, address, grade, room charge and number of rooms.

Write a function to print out hotel of given grade in order of room charges. */

```
#include <stdio.h>
```

```
#include <string.h>
```

```
typedef struct hotel
```

```
{
    char name[30];
    char add[200];
    char grade;
    float charge;
    int no_of_rooms;
} hotel;
```

```
int main (void)
```

```
{
    hotel list[100];
    hotel key;
    int no_of_hotels = 0;
    int choice;
    int success;
    char del_name[30];
    char grd;
    int i, j;
```

```
while (1)
```

```
{
    //Menu
    printf ("****Menu***\n");
    printf ("1. List Hotels\n");
    printf ("2. Edit Hotels\n");
    printf ("3. Exit\n");
    printf ("Choice: ");
    scanf ("%d", &choice);
```

```
    //Exit
```

```
    if (choice == 3)
        break;
```

```
    switch (choice)
```

```
    {
        //List hotels
        case 1:
            printf ("\nHotel Grade: ");
            fflush (stdin);
```

```

scanf ("%c", & grd);

printf ("List of hotels of grade %c\n\n", grd);
for (i = 0, j= 1; i <= no_of_hotels - 1; i++)
{
    if (list[i].grade == grd)
    {
        printf ("%d. %s\n", j++, list[i].name);
        printf ("%s\n", list[i].add);
        printf ("No of Rooms: %d\n", list[i].no_of_rooms);
        printf ("Rent: Rs. %.2f\n\n", list[i].charge);
    }
}
printf ("Total %d hotel found.\n\n", j - 1);
break;

//Edit Hotels
case 2:
    while (1)
    {
        //Hotel Edit Menu
        printf ("\n***Hotel Edit Menu***\n");
        printf ("1. Add Hotel\n");
        printf ("2. Delete Hotel\n");
        printf ("3. Exit Hotel Edit Menu\n");
        printf ("Choice: ");
        scanf ("%d", &choice);

        //Exit loop
        if (choice == 3)
        {
            printf ("\n");
            break;
        }

        switch (choice)
        {
            //Add Hotel
            case 1:
                printf ("\nAdd Hotel\n");
                printf ("Input Grade: ");
                fflush (stdin);
                scanf ("%c", &key.grade);
                printf ("Input Name: ");
                fflush (stdin);
                gets (key.name);
                printf ("Input Address: ");
                fflush (stdin);
                gets (key.add);
                printf ("Input Number of Rooms: ");
                scanf ("%d", &key.no_of_rooms);

```



```

printf ("Input Rent: Rs. ");
scanf ("%f", &key.charge);

//Sorting
for (i = no_of_hotels - 1; i >= 0 && key.charge < list[i].charge; i--)
    list[i + 1] = list[i];
list[i + 1] = key;
printf ("%s added successfully.\n\n", list[i + 1].name);
no_of_hotels++;
break;

//Delete Hotel
case 2:
    printf ("\nDelete Hotel\n");
    printf ("Input Name: ");
    fflush (stdin);
    gets (del_name);
    printf ("Input Grade: ");
    fflush (stdin);
    scanf ("%c", &grd);

    success = 0;
    for (i = 0; i <= no_of_hotels - 1; i++)
        if (!strcmp (del_name, list[i].name) && grd == list[i].grade)
        {
            for (j = i; j <= no_of_hotels - 2; j++)
                list[j] = list[j + 1];
            success = 1;
            no_of_hotels--;
        }

    if (success)
        printf ("Hotel %s deleted successfully.\n\n", del_name);
    else
        printf ("Hotel %s is not found.\n\n", del_name);
    break;

default:
    printf ("Error! Wrong Choice. Try Again\n");
}
}
break;

default:
    printf ("Error! Wrong Choice. Try Again\n\n");
}
}

return 0;
};

```

Output:

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 8\Lab-8-Program-35-Define a structure that can describe a hotel.exe
***Menu***
1. List Hotels
2. Edit Hotels
3. Exit
Choice: 1

Hotel Grade: 1
List of hotels of grade 1

Total 0 hotel found.

***Menu***
1. List Hotels
2. Edit Hotels
3. Exit
Choice: 3

-----
Process exited after 17.54 seconds with return value 0
Press any key to continue . . .
```

Program-36: Define a structure called cricket with player name, team name, batting average, for 50 players & 5 teams. Print team wise list contains names of player with their batting average.

Code:

```
/*Define a structure called cricket with player name, team name, batting
average, for 50 players & 5 teams. Print team wise list contains names of
player with their batting average.*/
/*player name
team name
batting average
Using cricket, declare an array player with 5 elements and write a program to
read the information about all the 5 players and print a team-wise list containing names of player with their
batting average.*/
#include <stdio.h>
#include <string.h>

struct cricket
{
    char player_name[20];
    char team_name[20];
    float batting_avg;
}p[50],t;

int main(void)
{
    int i=0,j=0,n=50;

    for(i=0;i<n;i++)
    {
        printf("\n Enter Player Name : ");
        scanf("%s",p[i].player_name);
        printf("\n Enter Team Name : ");
        scanf("%s",p[i].team_name);
        printf("\n Enter Batting Average : ");
        scanf("%f",&p[i].batting_avg);
    }

    //Sorting of Data based on Team
    for(i=0;i<n-1;i++)
    {
        for(j=i;j<n;j++)
        {
            if(strcmp(p[i].team_name,p[j].team_name)>0)
            {
                t=p[i];
                p[i]=p[j];
                p[j]=t;
            }
        }
    }
}
```

```
}

j=0;
for(i=0;i<n;i++)
{
if(strcmp(p[i].team_name,p[j].team_name)!=0 || i==0)
{
printf("\n Team Name: %s",p[i].team_name);
j=i;
}
printf("\n Player Name    = %s",p[i].player_name);
printf("\n Batting Average = %f",p[i].batting_avg);
}
return 0;
}
```

Lab-9

Program-37: Write a c program to copy & count the character content of one file says a.txt to another file b.txt.

Code:

```
/*Write a c program to copy & count the character content of one file says  
a.txt to another file b.txt. */
```

```
#include<stdio.h>
```

```
FILE *fp,*ft;
```

```
int main()
```

```
{
```

```
    int count=0;
```

```
    char ch;
```

```
    fp=fopen("a.txt","r");
```

```
    ft=fopen("b.txt","w");
```

```
    if(fp==NULL || ft==NULL)
```

```
        printf("File is unable to open\n");
```

```
    else
```

```
    {
```

```
        while((ch=fgetc(fp))!=EOF)
```

```
        {
```

```
            fputc(ch,ft);
```

```
            count++;
```

```
        }
```

```
    }
```

```
    // It is important to close the file after operation
```

```
    fclose(fp);
```

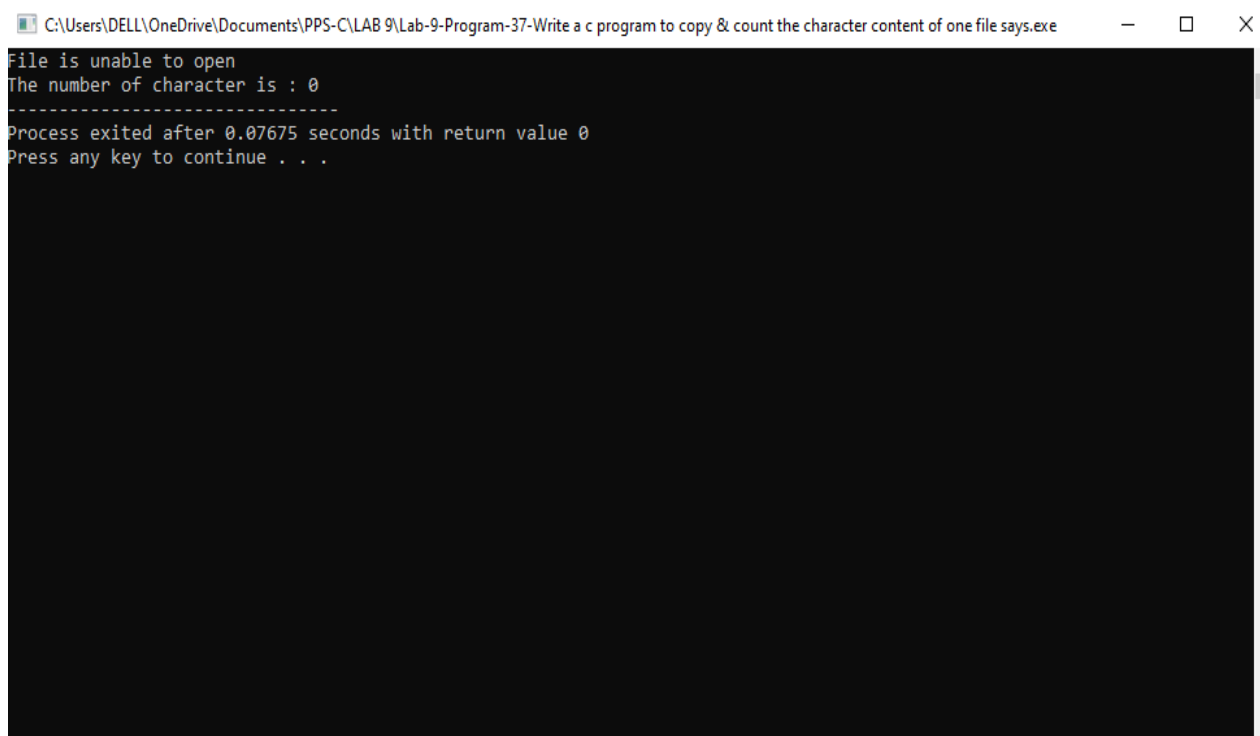
```
    fclose(ft);
```

```
    printf("The number of character is : %d",count);
```

```
    return 0;
```

```
}
```

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 9\Lab-9-Program-37-Write a c program to copy & count the character content of one file says.exe
File is unable to open
The number of character is : 0
-----
Process exited after 0.07675 seconds with return value 0
Press any key to continue . . .
```

Program-38: Write a program to take 10 integers from file and write square of these integer in other file.

Code:

```
/*Write a program to take 10 integers from file and write square of these  
integer in other file. */
```

```
#include <stdio.h>
```

```
int main (void)
```

```
{
```

```
    FILE *f_ptr_1, *f_ptr_2;
```

```
    char path_1[256], path_2[256];
```

```
    int num;
```

```
    int i;
```

```
    //Input first file path
```

```
    printf ("Input first file path: ");
```

```
    fflush (stdin);
```

```
    scanf ("%s", path_1);
```

```
    //Open first file
```

```
    f_ptr_1 = fopen (path_1, "r");
```

```
    if (f_ptr_1 == NULL)
```

```
    {
```

```
        printf ("Error! Unable to open first file.");
```

```
        return 0;
```

```
    }
```

```
    //Input second file path
```

```
    printf ("Input second file path: ");
```

```
    fflush (stdin);
```

```
    scanf ("%s", path_2);
```

```
    //Open second file
```

```
    f_ptr_2 = fopen (path_2, "w");
```

```
    if (f_ptr_2 == NULL)
```

```
    {
```

```
        printf ("Error! Unable to open second file.");
```

```
        return 0;
```

```
    }
```

```
    //Read integer from first file and write square in second file
```

```
    for (i = 0; i <= 9; i++)
```

```
    {
```

```
        if (fscanf (f_ptr_1, "%d", &num) != EOF)
```

```
            fprintf (f_ptr_2, "%d ", num * num);
```

```
        else
```

```
            break;
```

```
    }
```

```
    //Close files
```

```
fclose (f_ptr_1);  
fclose (f_ptr_2);  
printf ("File Saved Successfully.");  
  
return 0;  
};
```

Program-39: Write a program to read number from file and then write all 'odd' number to file ODD.txt & all even to file EVEN.txt.

Code:

/*Write a program to read number from file and then write all 'odd' number to file ODD.txt & all even to file EVEN.txt.

```
*/
#include <stdio.h>
main()
{
    FILE *f1, *f2, *f3;
    int number, i;
    printf("Contents of DATA file\n\n");
    f1 = fopen("DATA", "w"); /* Create DATA file */
    for(i = 1; i <= 30; i++)
    {
        scanf("%d", &number);
        if(number == -1) break;
        putw(number, f1);
    }
    fclose(f1);

    f1 = fopen("DATA", "r");
    f2 = fopen("ODD", "w");
    f3 = fopen("EVEN", "w");

    /* Read from DATA file */
    while((number = getw(f1)) != EOF)
    {
        if(number % 2 == 0)
            putw(number, f3); /* Write to EVEN file */
        else
            putw(number, f2); /* Write to ODD file */
    }
    fclose(f1);
    fclose(f2);
    fclose(f3);

    f2 = fopen("ODD", "r");
    f3 = fopen("EVEN", "r");
    printf("\n\nContents of ODD file\n\n");

    while((number = getw(f2)) != EOF)
        printf("%4d", number);
    printf("\n\nContents of EVEN file\n\n");

    while((number = getw(f3)) != EOF)
        printf("%4d", number);
    fclose(f2);
    fclose(f3);
}
```

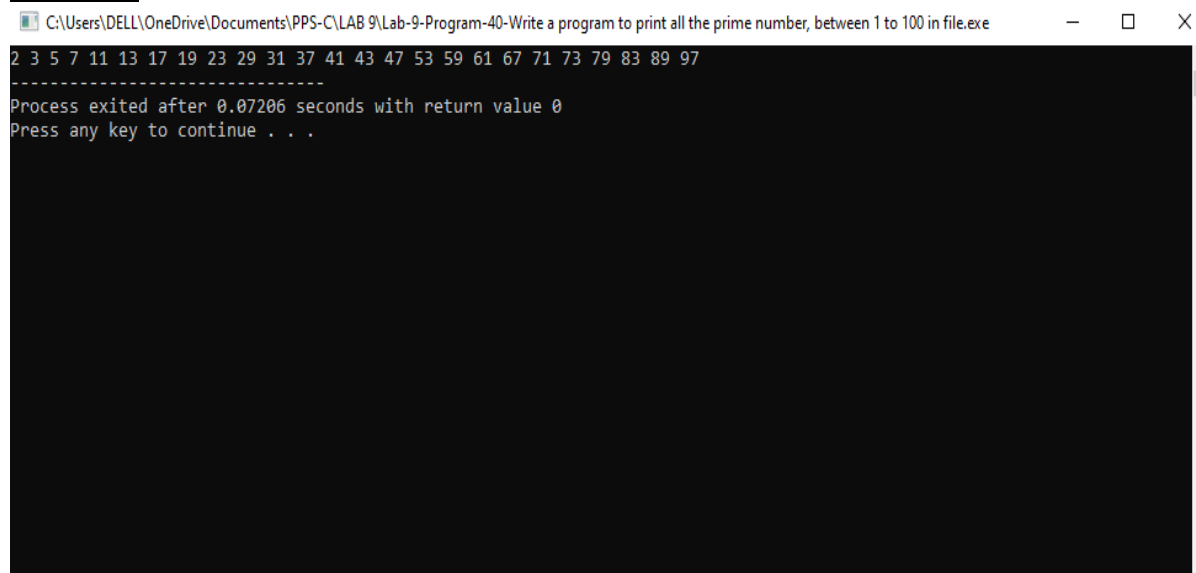
Program-40: Write a program to print all the prime number, between 1 to 100 in file prime.txt.

Code:

/*Write a program to print all the prime number, between 1 to 100 in file prime.txt.

```
*/  
#include <stdio.h>  
int main()  
{  
    int ct=0,n=0,i=1,j=1;  
    while(n<25)  
    {  
        j=1;  
        ct=0;  
        while(j<=i)  
        {  
            if(i%j==0)  
                ct++;  
            j++;  
        }  
        if(ct==2)  
        {  
            printf("%d ",i);  
            n++;  
        }  
        i++;  
    }  
}
```

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 9\Lab-9-Program-40-Write a program to print all the prime number, between 1 to 100 in file.exe  
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97  
-----  
Process exited after 0.07206 seconds with return value 0  
Press any key to continue . . .
```

Program-41: Write the following C program using pointer: a) To sort the list of numbers through pointer b) To reverse the string through pointer.

Code:

```
/*Write the following C program using pointer:
```

```
a) To sort the list of numbers through pointer
```

```
b) To reverse the string through pointer.
```

```
*/
```

```
//To sort the list of numbers through pointer
```

```
#include <stdio.h>
```

```
int main (void)
```

```
{
```

```
    int ar[20];
```

```
    int no_of_elements;
```

```
    int key;
```

```
    int i, j;
```

```
    //Input no of elements
```

```
    printf ("Input number of elements: ");
```

```
    scanf ("%d", &no_of_elements);
```

```
    //Input array
```

```
/*Write the following C program using pointer:
```

```
a) To sort the list of numbers through pointer
```

```
b) To reverse the string through pointer.
```

```
*/
```

```
//To reverse the string through pointer.
```

```
#include <stdio.h>
```

```
int main (void)
```

```
{
```

```
    char str[20];
```

```
    int length;
```

```
    char temp;
```

```
    int i, j;
```

```
    //Input string
```

```
    printf ("Input string: ");
```

```
    fflush (stdin);
```

```
    scanf ("%s", str);
```

```
    //fgets(str, 20, stdin);
```

```
    //Calculate string length
```

```
    length = 0;
```

```
    while (*(str + length) != '\0')
```

```
        length++;
```

```
    //Reverse string
```

```
    for (i = 0; i <= length / 2 - 1; i++)
```

```
{
```

```

    temp = *(str + i);
    *(str + i) = *(str + length - 1 - i);
    *(str + length - 1 - i) = temp;
}

//Print reverse string
printf ("Reverse String: ");
puts (str);

return 0;
}; for (i = 0; i <= no_of_elements - 1; i++)
    scanf ("%d", &ar[i]);

//Print Array
printf ("Array: ");
for (i = 0; i <= no_of_elements - 1; i++)
    printf ("%d ", ar[i]);

//Sorting
for (i = 1; i <= no_of_elements - 1; i++)
{
    key = *(ar + i);
    for (j = i - 1; j >= 0 && *(ar + j) > key; j--)
        *(ar + j + 1) = *(ar + j);
    *(ar + j + 1) = key;
}

//Print Array after sorting
printf ("\nSorted Array: ");
for (i = 0; i <= no_of_elements - 1; i++)
    printf ("%d ", ar[i]);

return 0;
};

```

Output:

```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 9\Lab-9-Program-41-Write the following C program using pointer(To sort the list of numbers through ...
Input number of elements: 5
Input elements: 233
232
1
33
55
Array: 233 232 1 33 55
Sorted Array: 1 33 55 232 233
-----
Process exited after 6.923 seconds with return value 0
Press any key to continue . . .
```

Code:

/*Write the following C program using pointer:

a) To sort the list of numbers through pointer

b) To reverse the string through pointer.

*/

//To reverse the string through pointer.

#include <stdio.h>

int main (void)

```
{
    char str[20];
    int length;
    char temp;
    int i, j;
```

//Input string

printf ("Input string: ");

fflush (stdin);

scanf ("%s", str);

//fgets(str, 20, stdin);

//Calculate string length

length = 0;

while (*(str + length) != '\0')

length++;

//Reverse string

for (i = 0; i <= length / 2 - 1; i++)

```
{
    temp = *(str + i);
```

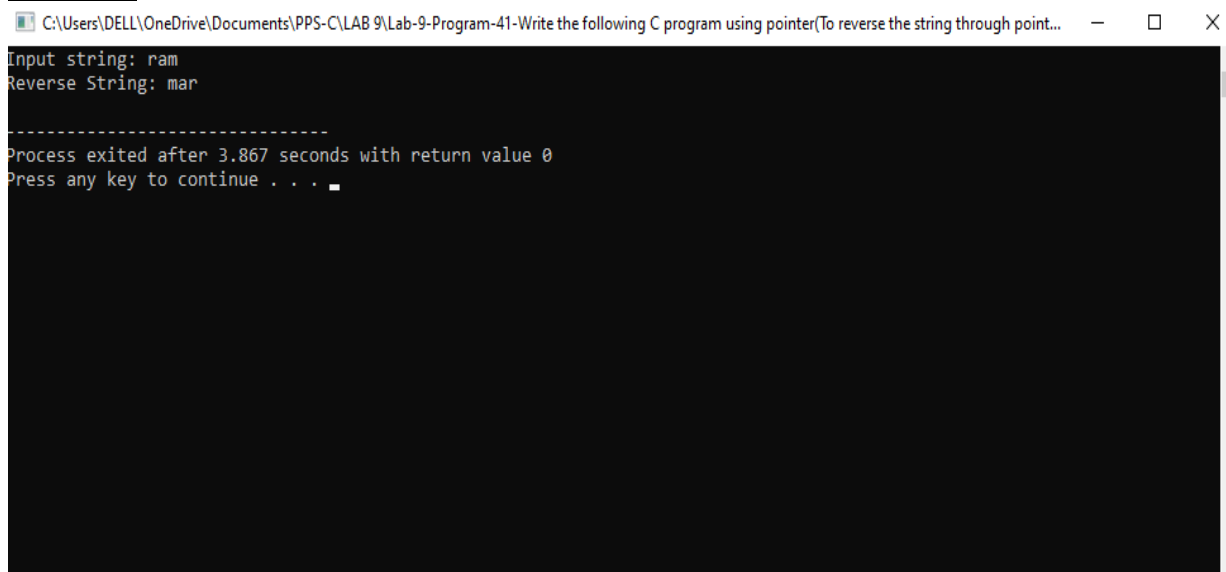
```
*(str + i) = *(str + length - 1 - i);
*(str + length - 1 - i) = temp;
}
```

```
//Print reverse string
printf ("Reverse String: ");
puts (str);
```

```
return 0;
```

```
};
```

Output:



```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 9\Lab-9-Program-41-Write the following C program using pointer(To reverse the string through point...
Input string: ram
Reverse String: mar

-----
Process exited after 3.867 seconds with return value 0
Press any key to continue . . .
```

Lab-10

Program-42: Write a program to find the largest no among 20 integers array using dynamic memory allocation.

Code:

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
    int i, num;
    float *data;

    printf("Enter total number of elements(1 to 100): ");
    scanf("%d",&num);

    // allocates the memory for 'num' elements
    data=(float*)calloc(num,sizeof(float));

    if(data==NULL)
    {
        printf("Error! Memory not Allocated.");
        exit(0);
    }

    printf("\n");

    // store the number entered by the User
    for(i=0;i<num;i++)
    {
        printf("Enter element %d:",i+1);
        scanf("%f",data+i);
    }

    // store largest number at address data
    for(i=0;i<num;i++)
    {
        if( *data < *(data+i))
            *data = *(data+i);
    }

    printf("Largest Element = %.2f",*data);

    return 0;
}
```

Program-43: Using Dynamic Memory Allocation, Write a program to find the transpose of given matrix.

Code:

/*Using Dynamic Memory Allocation, Write a program to find the transpose of given matrix.

*/

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
int main()
```

```
{
```

```
    //declaring pointers
```

```
    int *tranMatrix;
```

```
    int row, col;
```

```
    printf("Enter rows and columns in the matrix: ");
```

```
    scanf("%d%d", &row, &col);
```

```
    //memory allocating for matrix using dynamic memory allocation
```

```
    tranMatrix = (int *)calloc(row * col, sizeof(int));
```

```
    printf("Enter the rows and column value in matrix format: \n");
```

```
    for (int i = 0; i < row; i++)
```

```
    {
```

```
        for (int j = 0; j < col; j++)
```

```
        {
```

```
            scanf("%d", tranMatrix + (i * col + j) * sizeof(int));
```

```
        }
```

```
    }
```

```
    printf("Transpose of matrix is: \n");
```

```
    for (int i = 0; i < col; i++)
```

```
    {
```

```
        for (int j = 0; j < row; j++)
```

```
        {
```

```
            printf("%4d", *(tranMatrix + (j * col + i) * sizeof(int)));
```

```
        }
```

```
        printf("\n");
```

```
    }
```

```
    return 0;
```

```
}
```

Output:


```
C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 10\Lab-10-Program-43-Using Dynamic Memory Allocation, Write a program to find the transpose.exe
Enter rows and columns in the matrix: 2
2
Enter the rows and column value in matrix format:
21
32
2
23
Transpose of matrix is:
  21  2
  32  23

-----
Process exited after 9.464 seconds with return value 0
Press any key to continue . . .
```

Program-44: Write a program to find the factorial of given number using command line argument.

Code:

```
/*
Write a program to find the factorial of given number using command line
argument.
*/
// C program to find factorial of a number
// using command line arguments

#include <stdio.h>
#include <stdlib.h> /* atoi */

// Function to find factorial of given number
unsigned int factorial(unsigned int n)
{
    int res = 1, i;
    for (i = 2; i <= n; i++)
        res *= i;
    return res;
}

// Driver code
int main(int argc, char* argv[])
{
    int num, res = 0;

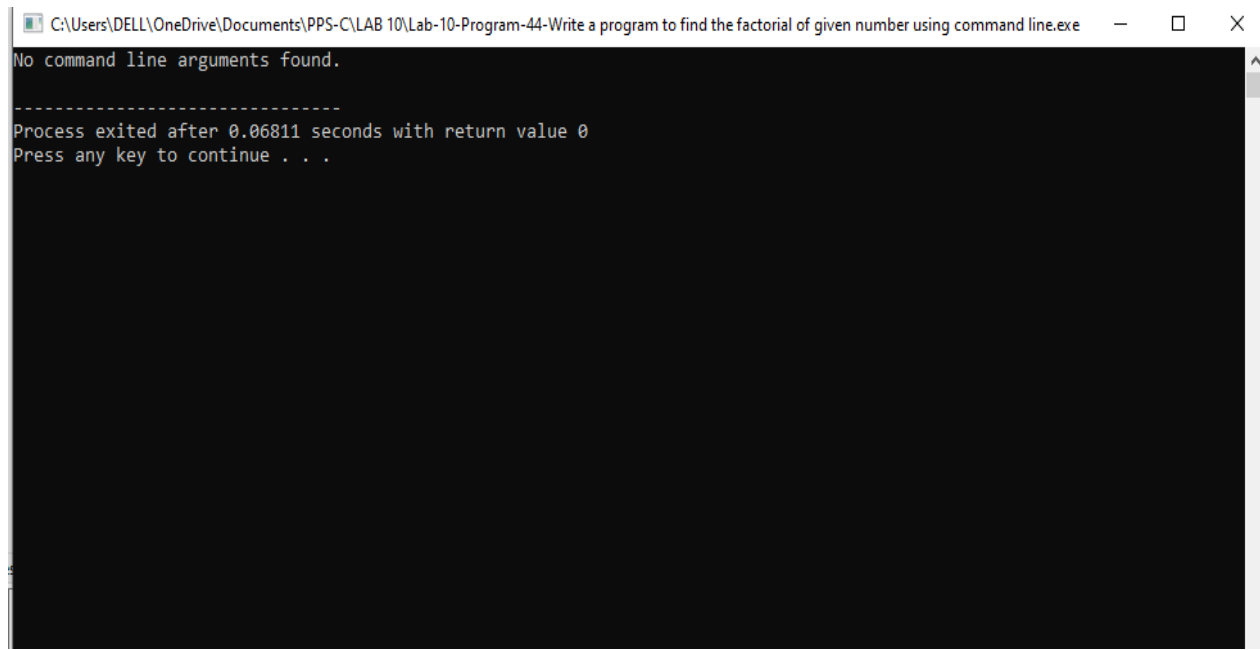
    // Check if the length of args array is 1
    if (argc == 1)
        printf("No command line arguments found.\n");

    else {

        // Get the command line argument and
        // Convert it from string type to integer type
        // using function "atoi( argument)"
        num = atoi(argv[1]);

        // Find the factorial
        printf("%d\n", factorial(num));
    }
    return 0;
}
```

Output:



C:\Users\DELL\OneDrive\Documents\PPS-C\LAB 10\Lab-10-Program-44-Write a program to find the factorial of given number using command line.exe

```
No command line arguments found.  
  
-----  
Process exited after 0.06811 seconds with return value 0  
Press any key to continue . . .
```

Program-45: Write a program to find the sum of digits of a 5 digit number using command line argument.

Code:

```
/*  
Write a program to find the sum of digits of a 5 digit number using  
command line argument.  
*/  
#include <stdio.h>  
#include <stdlib.h>  
  
int main(int argc, char * argv[])  
{  
    long num, temp, digit, sum = 0;  
    if(argc == 1 || argc > 2)  
    {  
        printf("Enter the number\n");  
        exit(1);  
    }  
    num = atoi (argv[1]) ;  
    temp = num;  
    while (num > 0)  
    {  
        digit = num % 10;  
        sum = sum + digit;  
        num /= 10;  
    }  
    printf("Sum of the digits of %ld = %ld\n", temp, sum);  
    return 0;  
}
```