



# Green University

## ASSIGNMENT SHEET

Section : DE-221

Name : Khondokar Saim .....

ID : 221902353 .....

Lecturer : Md. Atikuzzaman .....

Subject : CSE-103 .....

Date : 23 / 07 / 2022 .....

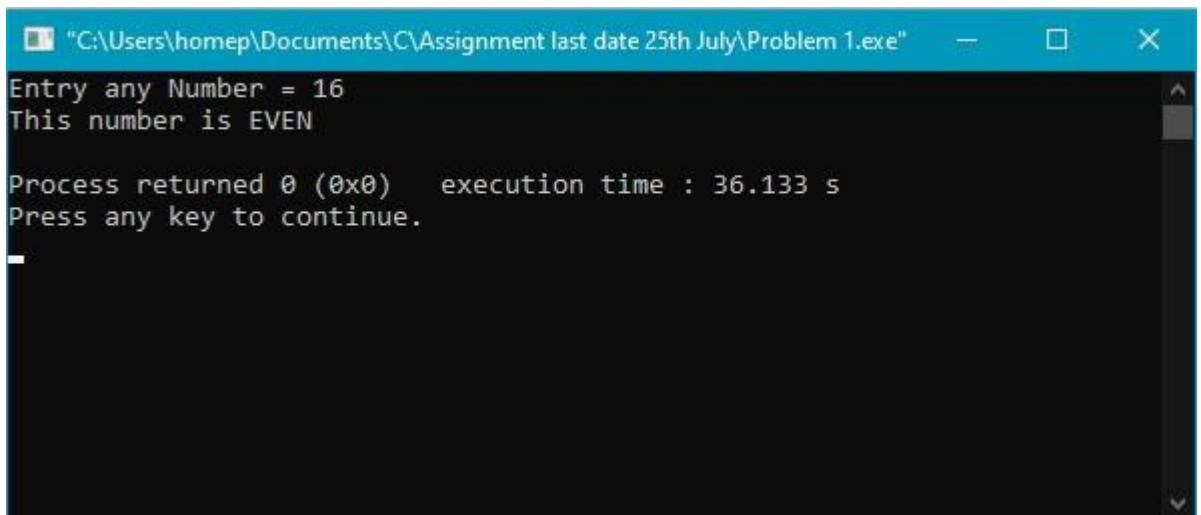
## Problem-1 :

### ➤ C Program to Check Whether a Number is Even or Odd.

#### CODE :

```
/*  
Check Whether a Number is Even or Odd  
*/  
#include<stdio.h>  
int main ()  
{  
    int num;  
  
    printf("Entry any Number = ");  
    scanf("%d",&num);  
  
    if(num%2==0)  
        printf("This number is EVEN\n");  
  
    else  
        printf("This number is ODD\n");  
  
    return 0;  
}
```

#### RESULT :



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 1.exe"  _  □  ×  
Entry any Number = 16  
This number is EVEN  
  
Process returned 0 (0x0)   execution time : 36.133 s  
Press any key to continue.  
_
```

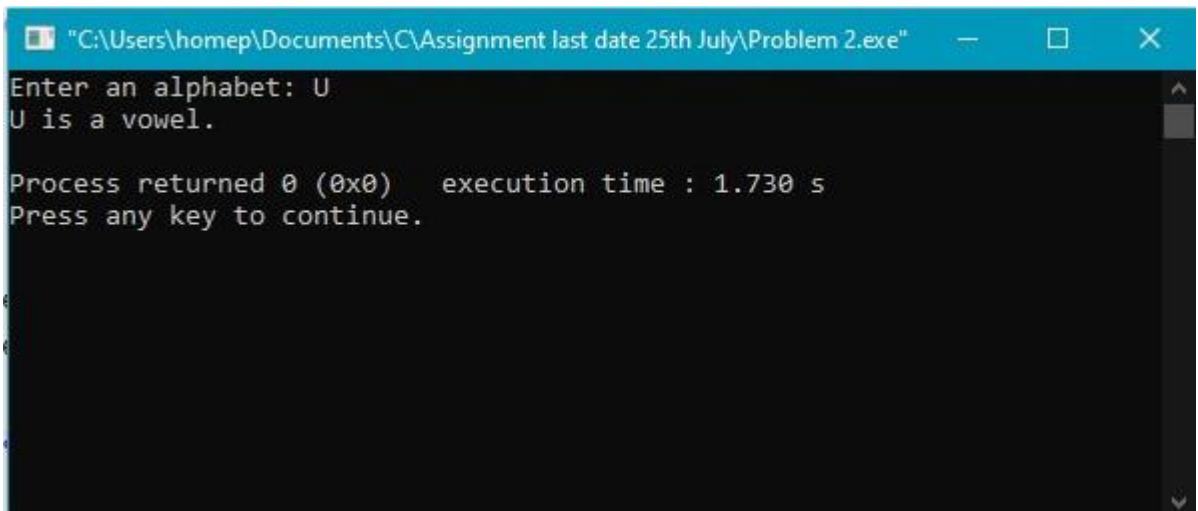
## Problem-2 :

### ➤ C Program to Check Whether a Character is a Vowel or Consonant.

#### CODE :

```
/*  
Check Whether a Character is a Vowel or Consonant.  
*/  
#include <stdio.h>  
int main()  
{  
    char c;  
    int lowercase_vowel;  
    int uppercase_vowel;  
  
    printf("Enter an alphabet: ");  
    scanf("%c",&c);  
  
    lowercase_vowel = (c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u');  
  
    uppercase_vowel = (c == 'A' || c == 'E' || c == 'I' || c == 'O' || c == 'U');  
  
    if (lowercase_vowel || uppercase_vowel)  
        printf("%c is a vowel.\n",c);  
  
    else  
        printf("%c is a consonant.\n",c);  
  
    return 0;  
}
```

#### RESULT :



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 2.exe"  
Enter an alphabet: U  
U is a vowel.  
  
Process returned 0 (0x0)   execution time : 1.730 s  
Press any key to continue.
```

### Problem-3 :

#### ➤ C Program to Find the Largest Number Among Three Numbers

CODE :

```
/*  
Find the Largest Number Among Three Numbers  
*/  
#include<stdio.h>  
int main()  
{  
    int Number1;  
    int Number2;  
    int Number3;  
  
    printf("Enter 1st Number = ");  
    scanf("%d",&Number1);  
  
    printf("Enter 2nd Number = ");  
    scanf("%d",&Number2);  
  
    printf("Enter 3rd Number = ");  
    scanf("%d",&Number3);  
  
    if(Number1>Number2&&Number1>Number3)  
        printf("\nLargest Number = %d\n",Number1);  
  
    else if(Number2>Number1&&Number2>Number3)  
        printf("\nLargest Number= %d\n",Number2);  
  
    else if(Number3>Number1&&Number3>Number2)  
        printf("\nLargest Number = %d\n",Number3);  
  
    else  
        printf("This Number is Equal\n");  
  
    return 0;  
}
```

## RESULT :

```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 3.exe"
Enter 1st Number = 16
Enter 2nd Number = 48
Enter 3rd Number = 33

Largest Number= 48

Process returned 0 (0x0)   execution time : 14.278 s
Press any key to continue.
```

## Problem-4 :

### ➤ C Program to Find the Roots of a Quadratic Equation.

CODE :

```
/*
Find Roots of a Quadratic Equation
*/
#include <math.h>
#include <stdio.h>
int main()
{
    double a,b,c,discriminant,root1,root2;

    printf("Enter coefficients a : ");
    scanf("%lf",&a);

    printf("Enter coefficients b : ");
    scanf("%lf",&b);

    printf("Enter coefficients c: ");
    scanf("%lf",&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("\nroot1 = %.2lf\n",root1);
        printf("\nroot2 = %.2lf\n",root2);
    }

    // condition for real and equal roots
    else if (discriminant == 0)
    {
        root1 = -b / (2 * a);

        root2 = -b / (2 * a);

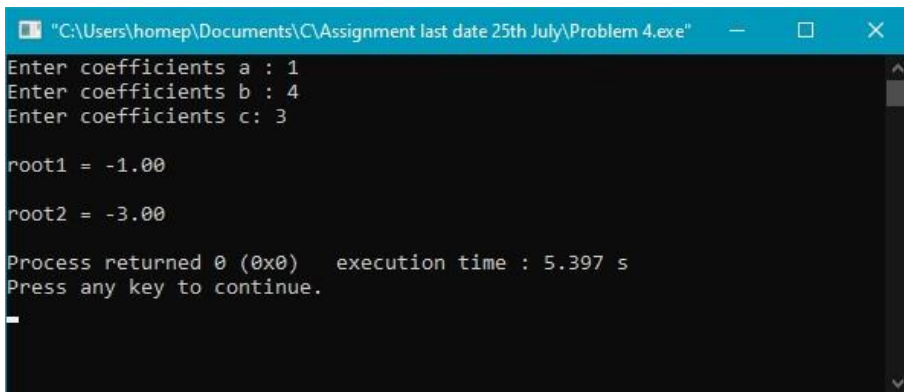
        printf("\nroot1 = %.2lf\n",root1);
        printf("\nroot2 = %.2lf\n",root2);
    }
}
```

```
// if roots are not real
else
{
    printf("\nRoots are imaginary\n");
}

return 0;
}
```

## RESULT:

### Case-1

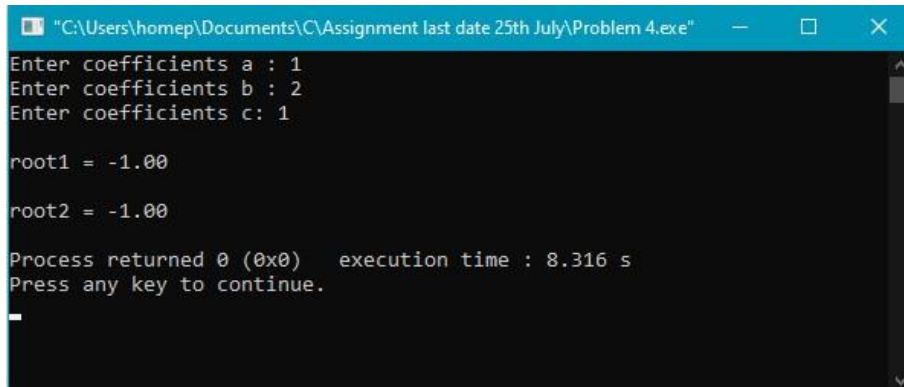


```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 4.exe"
Enter coefficients a : 1
Enter coefficients b : 4
Enter coefficients c : 3

root1 = -1.00
root2 = -3.00

Process returned 0 (0x0)   execution time : 5.397 s
Press any key to continue.
```

### Case-2

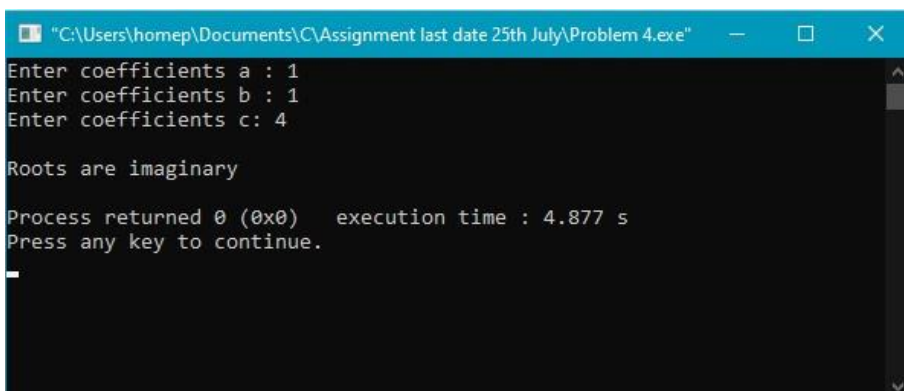


```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 4.exe"
Enter coefficients a : 1
Enter coefficients b : 2
Enter coefficients c : 1

root1 = -1.00
root2 = -1.00

Process returned 0 (0x0)   execution time : 8.316 s
Press any key to continue.
```

### Case-3



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 4.exe"
Enter coefficients a : 1
Enter coefficients b : 1
Enter coefficients c : 4

Roots are imaginary

Process returned 0 (0x0)   execution time : 4.877 s
Press any key to continue.
```

## Problem-5 :

### ➤ C Program to Check Leap Year.

CODE :

```
/*
To Check Leap Year.
*/
#include <stdio.h>
int main()
{
    int year;
    printf("Enter a year: ");
    scanf("%d",&year);

    // leap year if perfectly divisible by 400
    if(year%400==0)
    {
        printf("\n%d is a leap year.\n",year);
    }

    // not a leap year if divisible by 100
    else if(year%100==0)
    {
        printf("\n%d is not a leap year.\n",year);
    }

    // leap year if not divisible by 100
    // but divisible by 4
    else if (year%4==0)
    {
        printf("\n%d is a leap year.\n",year);
    }

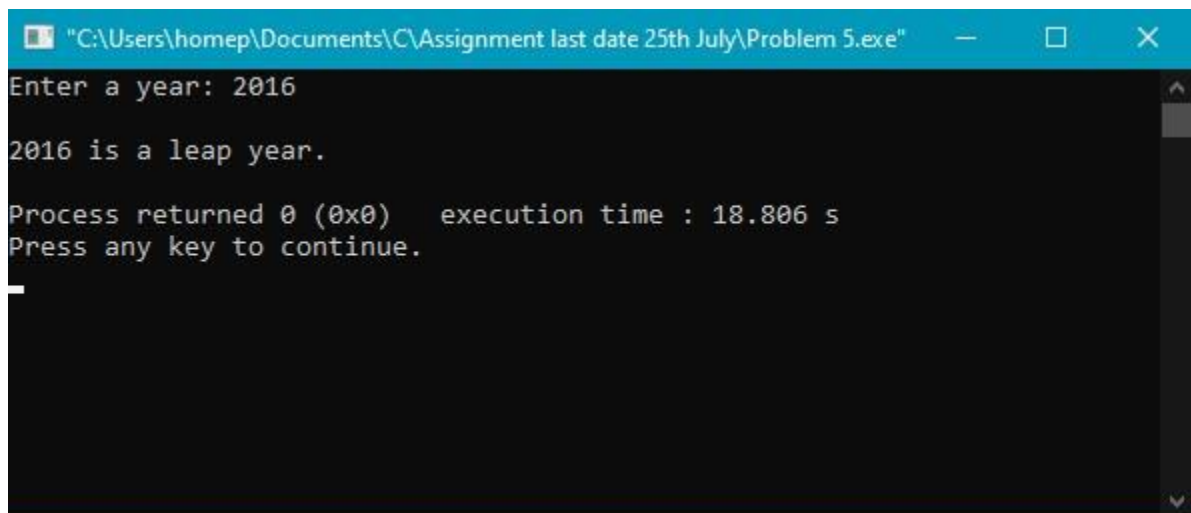
    // all other years are not leap years
    else
    {
        printf("\n%d is not a leap year.\n",year);
    }

    return 0;
}
```



## RESULT:

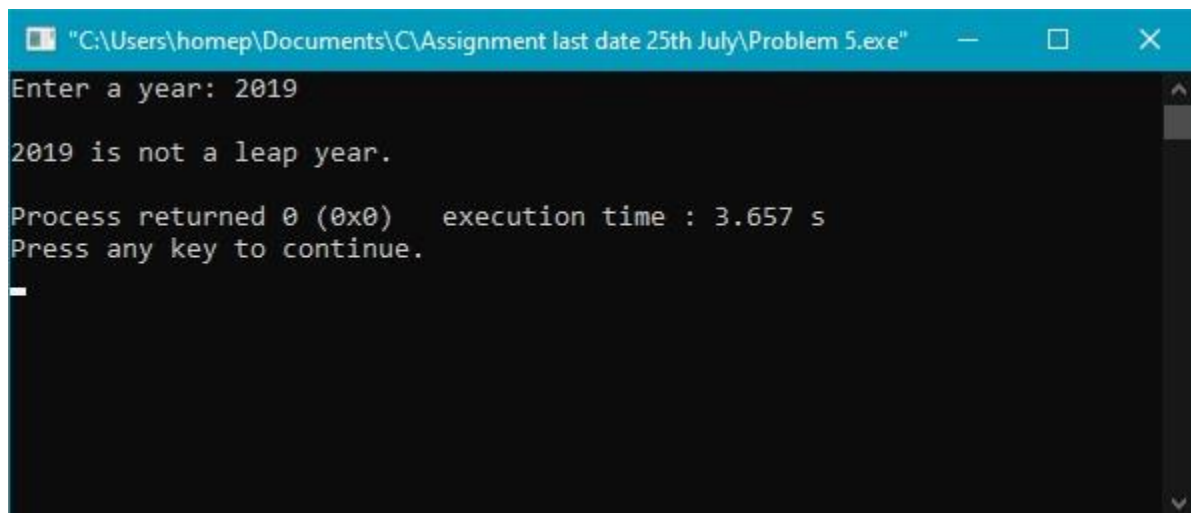
### Case-1



A screenshot of a Windows command prompt window. The title bar is blue and contains the text "C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 5.exe" along with standard window controls. The command prompt has a black background with white text. The text displayed is: "Enter a year: 2016", "2016 is a leap year.", "Process returned 0 (0x0) execution time : 18.806 s", and "Press any key to continue." followed by a cursor on a new line.

```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 5.exe"
Enter a year: 2016
2016 is a leap year.
Process returned 0 (0x0) execution time : 18.806 s
Press any key to continue.
```

### Case-2



A screenshot of a Windows command prompt window. The title bar is blue and contains the text "C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 5.exe" along with standard window controls. The command prompt has a black background with white text. The text displayed is: "Enter a year: 2019", "2019 is not a leap year.", "Process returned 0 (0x0) execution time : 3.657 s", and "Press any key to continue." followed by a cursor on a new line.

```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 5.exe"
Enter a year: 2019
2019 is not a leap year.
Process returned 0 (0x0) execution time : 3.657 s
Press any key to continue.
```

## Problem-6 :

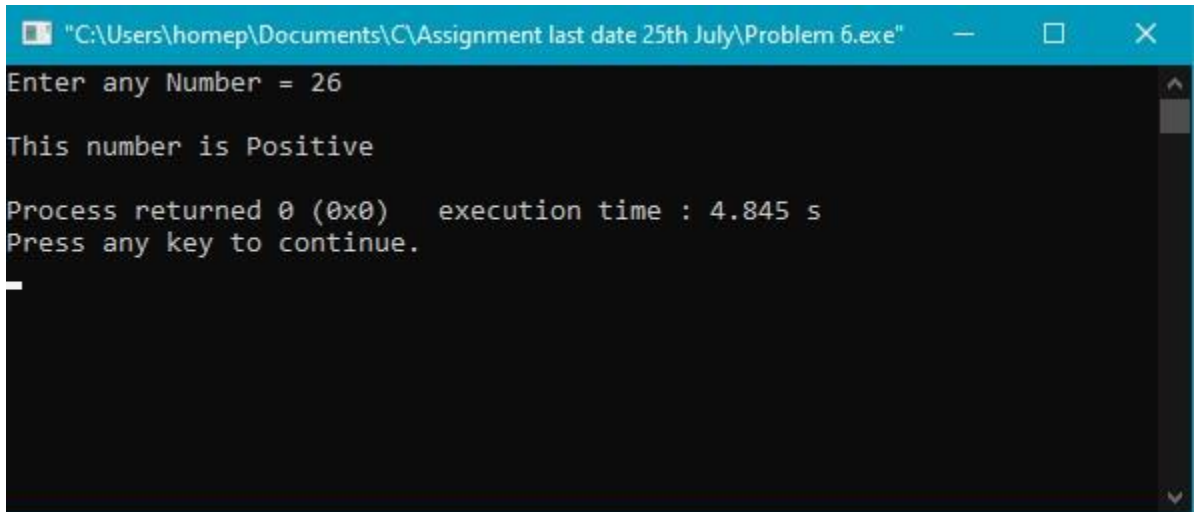
### ➤ C Program to Check Whether a Number is Positive or Negative.

CODE :

```
/*  
To Check Whether a Number is Positive or Negative.  
*/  
#include <stdio.h>  
int main()  
{  
    int num;  
  
    printf("Enter any Number = ");  
    scanf("%d",&num);  
  
    //When number is greater than zero  
    if(num>0)  
    {  
        printf("\nThis number is Positive\n");  
    }  
  
    //When number is less than zero  
    else if(num<0)  
    {  
        printf("\nThis number is Negative\n");  
    }  
  
    //when number is zero  
    else  
        printf("\nThis number is called ZERO\n");  
  
    return 0;  
}
```

## RESULT :

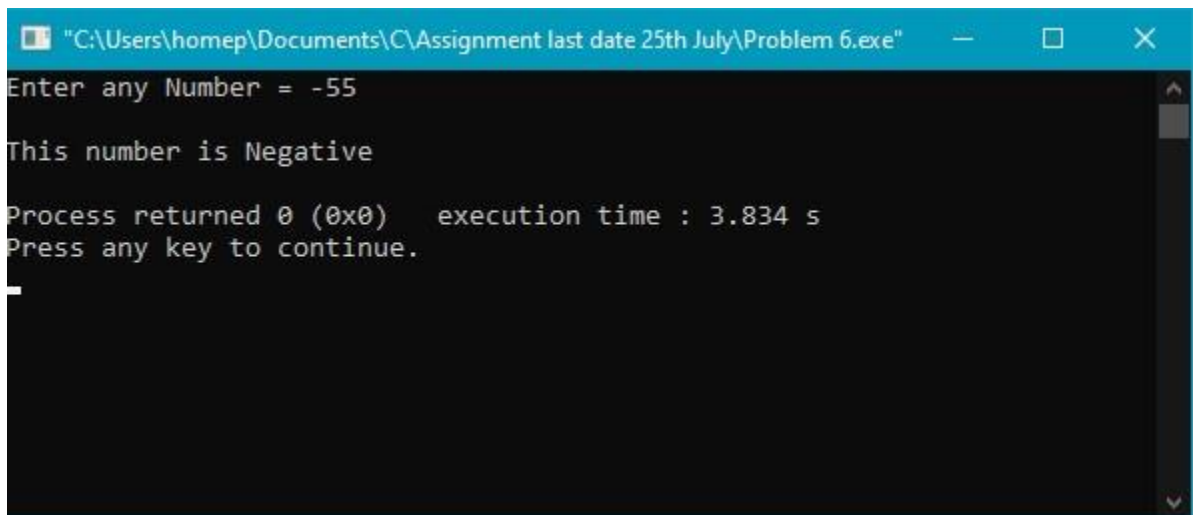
### Case-1



The screenshot shows a Windows command prompt window with a blue title bar. The title bar text is "C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 6.exe". The window content is as follows:

```
Enter any Number = 26  
This number is Positive  
Process returned 0 (0x0)   execution time : 4.845 s  
Press any key to continue.  
_
```

### Case-2



The screenshot shows a Windows command prompt window with a blue title bar. The title bar text is "C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 6.exe". The window content is as follows:

```
Enter any Number = -55  
This number is Negative  
Process returned 0 (0x0)   execution time : 3.834 s  
Press any key to continue.  
_
```

## Problem-7 :

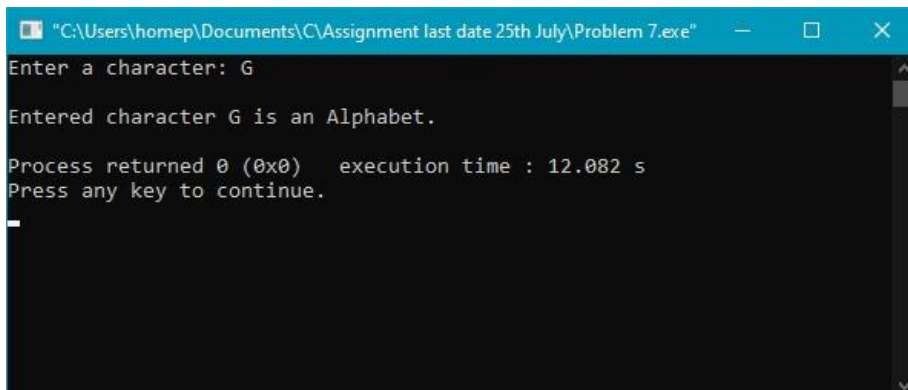
### ➤ C Program to Check Whether a Character is an Alphabet or not

#### CODE :

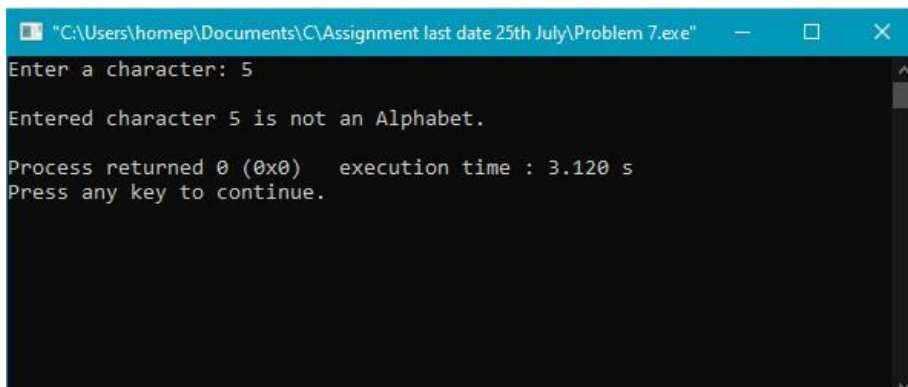
```
/*  
To Check Whether a Character is an Alphabet or not  
*/  
#include <stdio.h>  
int main()  
{  
    char ch;  
    printf("Enter a character: ");  
    scanf("%c",&ch);  
  
    if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))  
        printf("\nEntered character %c is an Alphabet.\n", ch);  
  
    else  
        printf("\nEntered character %c is not an Alphabet.\n",ch);  
  
    return 0;  
}
```

#### RESULT :

##### Case-1



##### Case-2



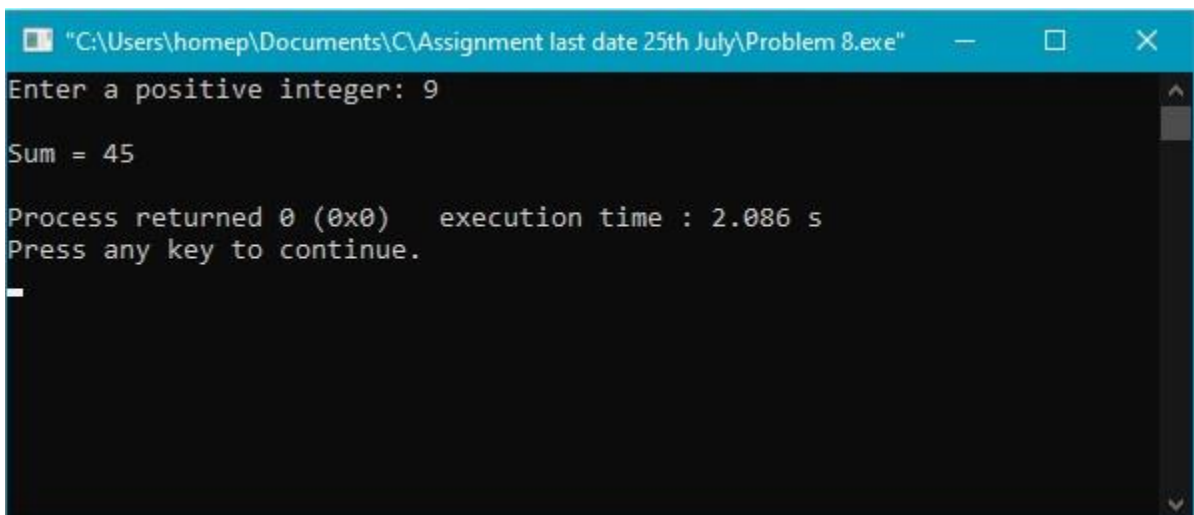
## Problem-8 :

### ➤ C Program to Calculate the Sum of Natural Numbers.

#### CODE :

```
/*  
To Calculate the Sum of Natural Numbers  
*/  
#include <stdio.h>  
int main() {  
    int n,i,sum = 0;  
  
    printf("Enter a positive integer: ");  
    scanf("%d",&n);  
  
    for (i = 1; i <= n; ++i)  
    {  
        sum += i;  
    }  
  
    printf("\nSum = %d\n",sum);  
  
    return 0;  
}
```

#### RESULT :



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 8.exe"  
Enter a positive integer: 9  
Sum = 45  
Process returned 0 (0x0)   execution time : 2.086 s  
Press any key to continue.  
_
```

## Problem-9 :

### ➤ C Program to Find Factorial of a Number

CODE :

```
/*
To Find Factorial of a Number
*/
#include <stdio.h>
int main()
{
    int n,i;
    unsigned long fact = 1;

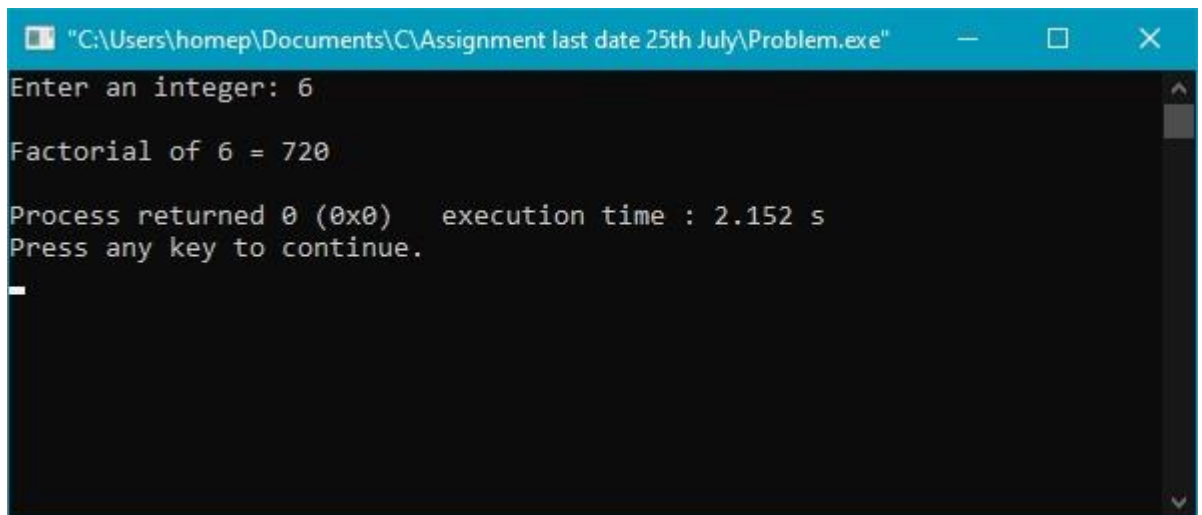
    printf("Enter an integer: ");
    scanf("%d",&n);

    // shows error if the user enters a negative integer
    if (n<0)
        printf("\nError! Factorial of a negative number doesn't exist.\n");
    else
    {
        for (i = 1; i <= n; ++i)
        {
            fact *= i;
        }
        printf("\nFactorial of %d = %llu\n",n,fact);
    }

    return 0;
}
```

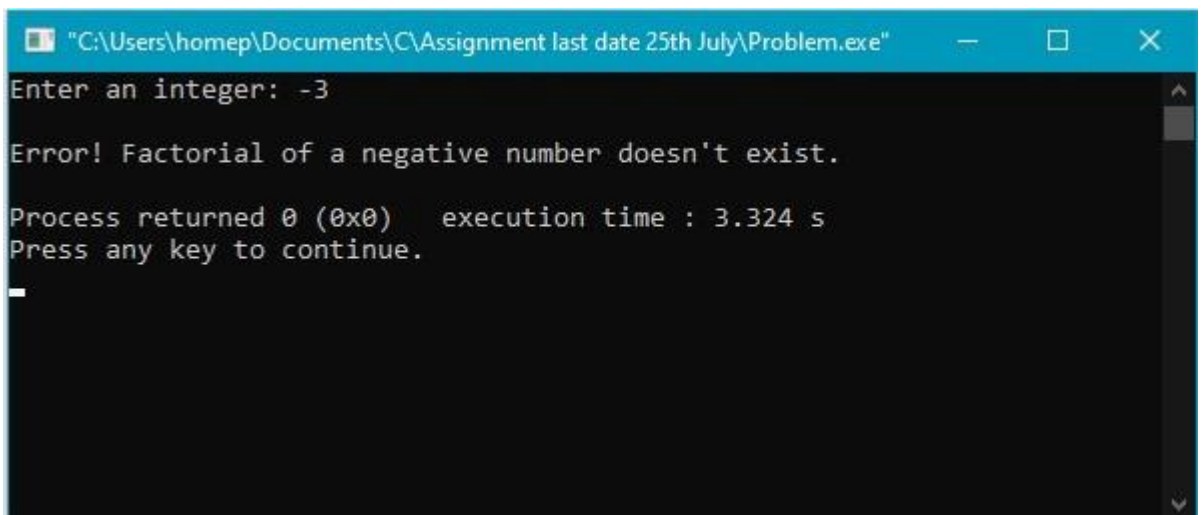
## RESULT:

### Case-1



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem.exe"
Enter an integer: 6
Factorial of 6 = 720
Process returned 0 (0x0)   execution time : 2.152 s
Press any key to continue.
-
```

### Case-2



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem.exe"
Enter an integer: -3
Error! Factorial of a negative number doesn't exist.
Process returned 0 (0x0)   execution time : 3.324 s
Press any key to continue.
-
```

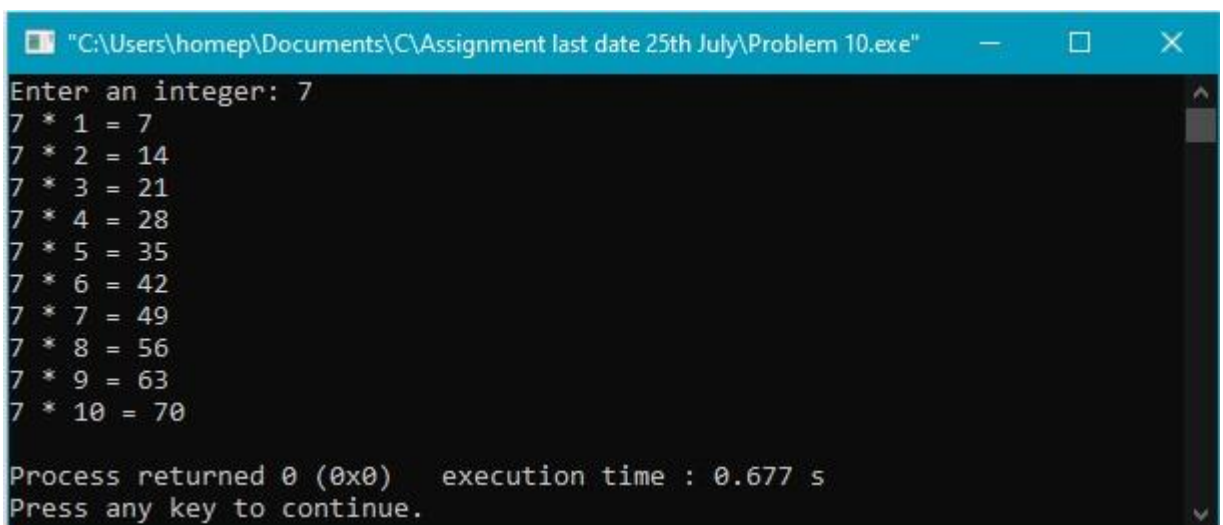
## Problem-10 :

### ➤ C Program to Generate Multiplication Table

#### CODE :

```
/*  
To Generate Multiplication Table  
*/  
#include <stdio.h>  
int main()  
{  
    int n,i;  
  
    printf("Enter an integer: ");  
    scanf("%d", &n);  
  
    for (i = 1; i <= 10; ++i)  
    {  
        printf("%d * %d = %d \n",n, i,n * i);  
    }  
    return 0;  
}
```

#### RESULT :



The screenshot shows a Windows command prompt window titled "C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 10.exe". The prompt displays the output of the C program, which generates a multiplication table for the number 7. The output is as follows:

```
Enter an integer: 7  
7 * 1 = 7  
7 * 2 = 14  
7 * 3 = 21  
7 * 4 = 28  
7 * 5 = 35  
7 * 6 = 42  
7 * 7 = 49  
7 * 8 = 56  
7 * 9 = 63  
7 * 10 = 70  
  
Process returned 0 (0x0)   execution time : 0.677 s  
Press any key to continue.
```



## Problem-11 :

### ➤ C Program to Display Fibonacci Sequence

#### CODE :

```
/*
To Display Fibonacci Sequence.
*/
#include <stdio.h>
int main()
{
    int i;
    int n;
    int t1 = 0, t2 = 1;
    int nextTerm = t1 + t2;

    printf("Enter the number of terms: ");
    scanf("%d", &n);

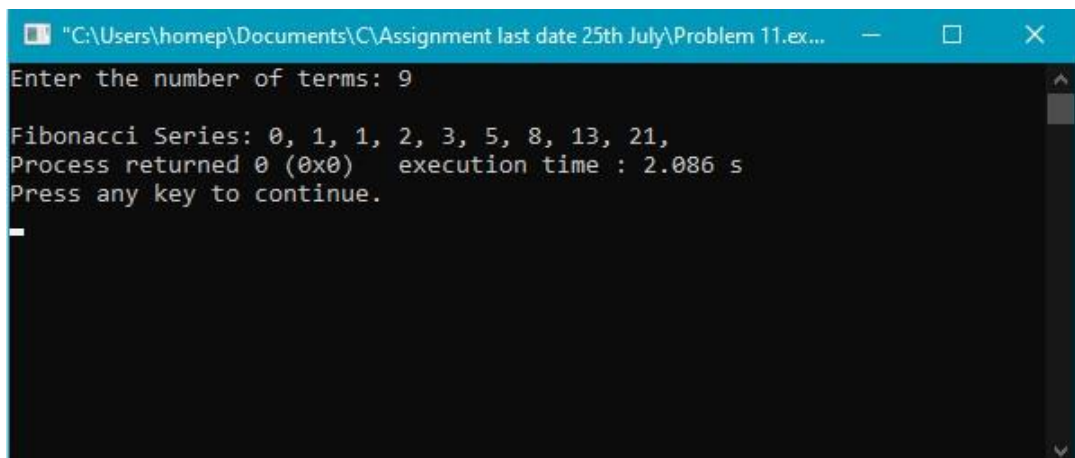
    printf("\nFibonacci Series: %d, %d, ", t1, t2);

    for (i = 3; i <= n; ++i)
    {
        printf("%d, ", nextTerm);

        t1 = t2;
        t2 = nextTerm;
        nextTerm = t1 + t2;
    }

    return 0;
}
```

#### RESULT :

A screenshot of a Windows command prompt window titled "C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 11.ex...". The window shows the execution of a C program. The user is prompted to "Enter the number of terms: 9". The program then displays the "Fibonacci Series: 0, 1, 1, 2, 3, 5, 8, 13, 21,". Below this, it shows "Process returned 0 (0x0) execution time : 2.086 s" and "Press any key to continue." followed by a cursor on a new line.

```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 11.ex..."
Enter the number of terms: 9

Fibonacci Series: 0, 1, 1, 2, 3, 5, 8, 13, 21,
Process returned 0 (0x0) execution time : 2.086 s
Press any key to continue.
```

## Problem-12 :

### ➤ C Program to Find GCD of two Numbers

#### CODE :

```
/*
To Find GCD of two Numbers.
*/
#include <stdio.h>
int main()
{
    int n1;
    int n2;

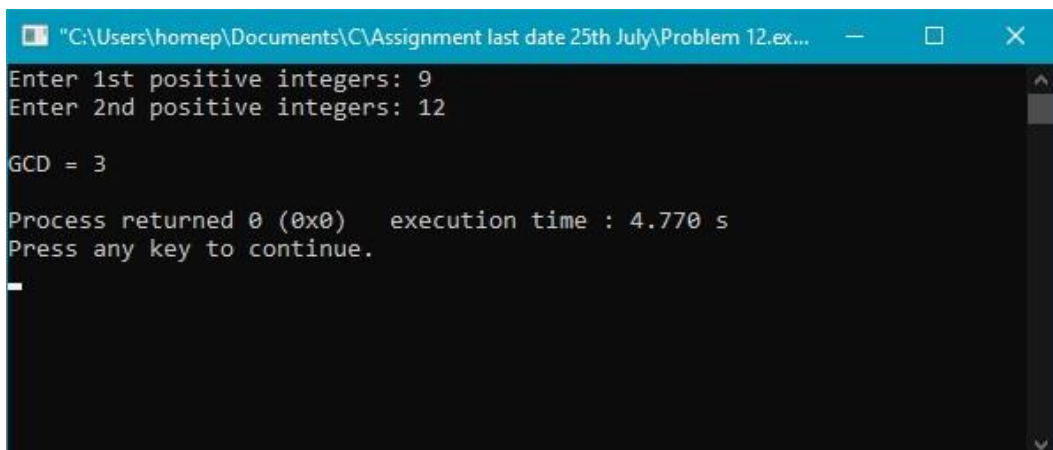
    printf("Enter 1st positive integers: ");
    scanf("%d",&n1);

    printf("Enter 2nd positive integers: ");
    scanf("%d",&n2);

    while(n1!=n2)
    {
        if(n1 > n2)
            n1 -= n2;
        else
            n2 -= n1;
    }
    printf("\nGCD = %d\n",n1);

    return 0;
}
```

#### RESULT :



The screenshot shows a Windows command prompt window titled "C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 12.ex...". The program prompts for two positive integers. The first input is 9, and the second input is 12. The program then outputs "GCD = 3". At the bottom, it shows "Process returned 0 (0x0) execution time : 4.770 s" and "Press any key to continue." with a cursor on a new line.

```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 12.ex..."
Enter 1st positive integers: 9
Enter 2nd positive integers: 12

GCD = 3

Process returned 0 (0x0)   execution time : 4.770 s
Press any key to continue.
_
```

### Problem-13 :

#### ➤ C Program to Find LCM of two Numbers

##### CODE :

```
/*  
To Find LCM of two Numbers  
*/  
#include <stdio.h>  
int main()  
{  
    int n1;  
    int n2;  
    int max;  
  
    printf("Enter 1st positive integer: ");  
    scanf("%d",&n1);  
  
    printf("Enter 2nd positive integer: ");  
    scanf("%d",&n2);  
  
    max = (n1 > n2) ? n1 : n2;  
  
    while (1)  
    {  
        if (max%n1 == 0 && max % n2 == 0)  
        {  
            printf("\nThe LCM of %d and %d is %d.\n", n1, n2, max);  
            break;  
        }  
        ++max;  
    }  
  
    return 0;  
}
```

##### RESULT :

A screenshot of a Windows command prompt window. The title bar shows the file path "C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 13.ex...". The command prompt displays the following text:  
Enter 1st positive integer: 55  
Enter 2nd positive integer: 12  
  
The LCM of 55 and 12 is 660.  
  
Process returned 0 (0x0) execution time : 3.609 s  
Press any key to continue.

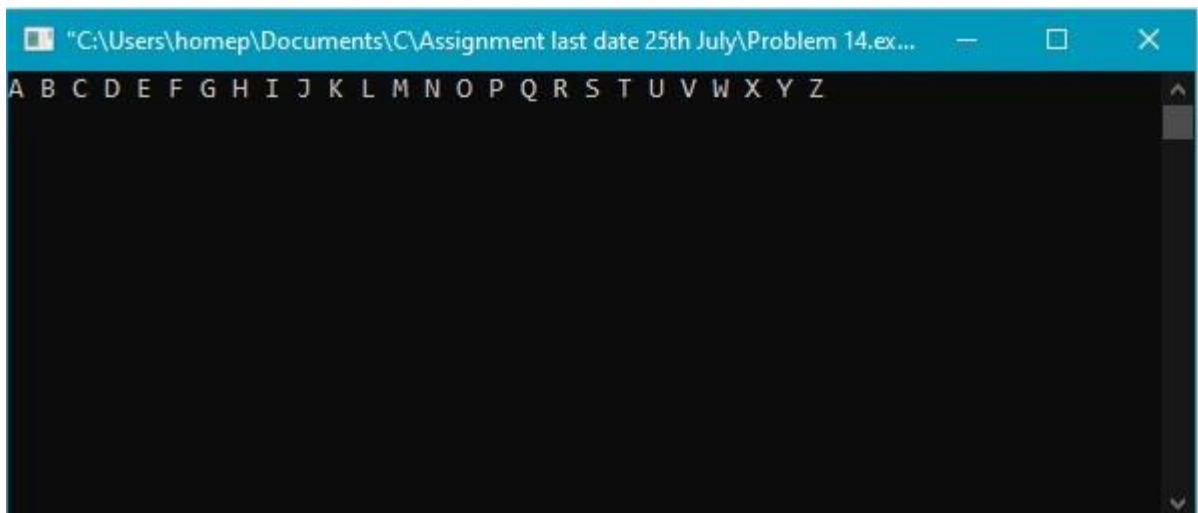
## Problem-14 :

### ➤ C Program to Display Characters from A to Z Using Loop

#### CODE :

```
/*  
To Display Characters from A to Z Using Loop  
*/  
#include <stdio.h>  
int main()  
{  
    char c;  
  
    for (c = 'A'; c <= 'Z'; ++c)  
        printf("%c ", c);  
  
    getchar();  
}
```

#### RESULT :



## Problem-15 :

### ➤ C Program to Count Number of Digits in an Integer

#### CODE :

```
/*
To Count Number of Digits in an Integer
*/
#include <stdio.h>
int main()
{
    long long n;
    int count = 0;

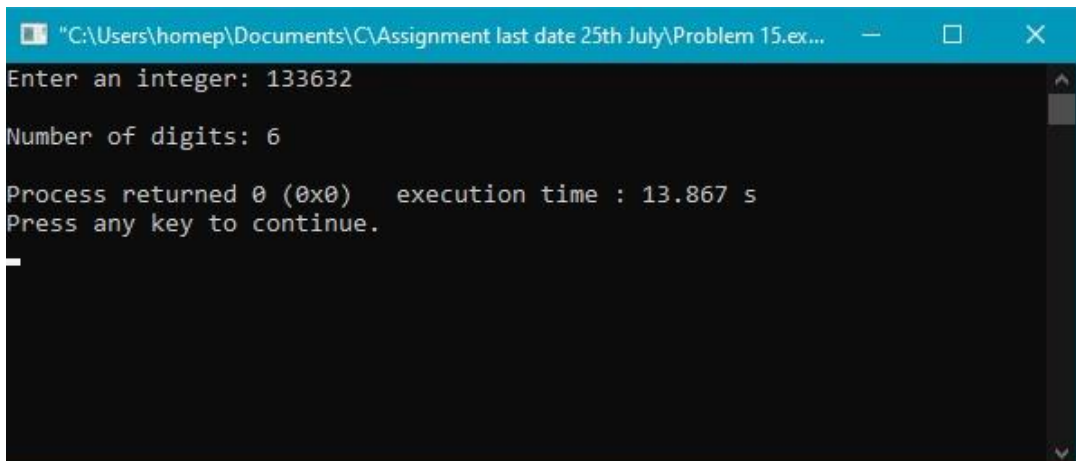
    printf("Enter an integer: ");
    scanf("%lld",&n);

    // iterate at least once, then until n becomes 0
    // remove last digit from n in each iteration
    // increase count by 1 in each iteration
    do
    {
        n /= 10;
        ++count;
    }
    while (n != 0);

    printf("\nNumber of digits: %d\n",count);

    return 0;
}
```

#### RESULT:



The screenshot shows a Windows command prompt window titled "C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 15.ex...". The prompt displays the following text:

```
Enter an integer: 133632
Number of digits: 6
Process returned 0 (0x0)   execution time : 13.867 s
Press any key to continue.
```

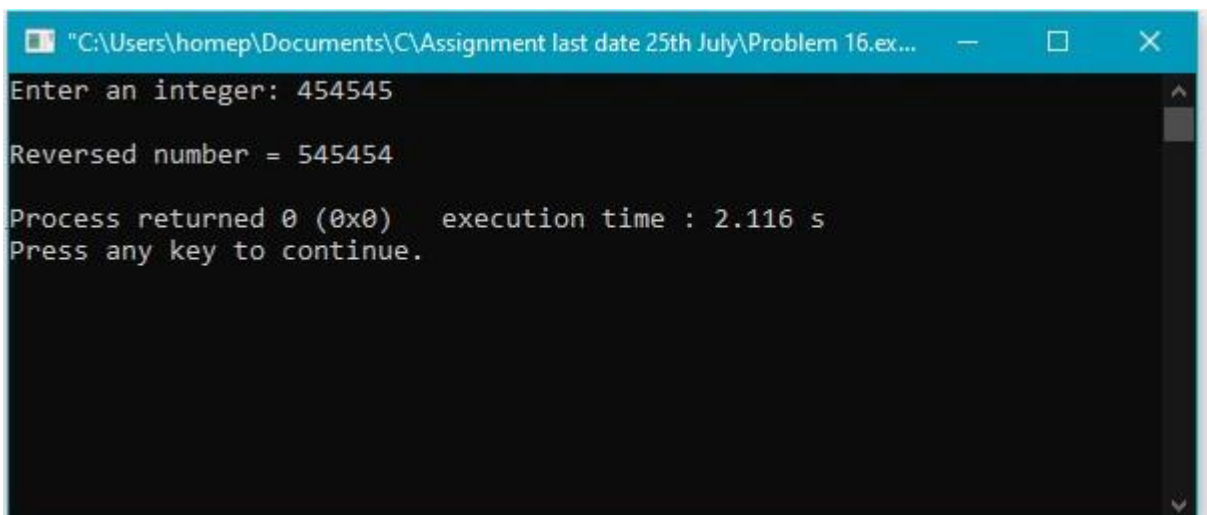
## Problem-16 :

### ➤ C Program to Reverse a Number.

#### CODE :

```
/*  
C Program to Reverse a Number  
*/  
#include <stdio.h>  
int main()  
{  
    int n,reverse = 0,remainder;  
  
    printf("Enter an integer: ");  
    scanf("%d",&n);  
  
    while (n != 0)  
    {  
        remainder = n % 10;  
        reverse = reverse * 10 + remainder;  
        n /= 10;  
    }  
  
    printf("\nReversed number = %d\n",reverse);  
  
    return 0;  
}
```

#### RESULT :



The screenshot shows a Windows command prompt window titled "C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 16.ex...". The window has a black background with white text. The text displayed is as follows:

```
Enter an integer: 454545  
  
Reversed number = 545454  
  
Process returned 0 (0x0)   execution time : 2.116 s  
Press any key to continue.
```

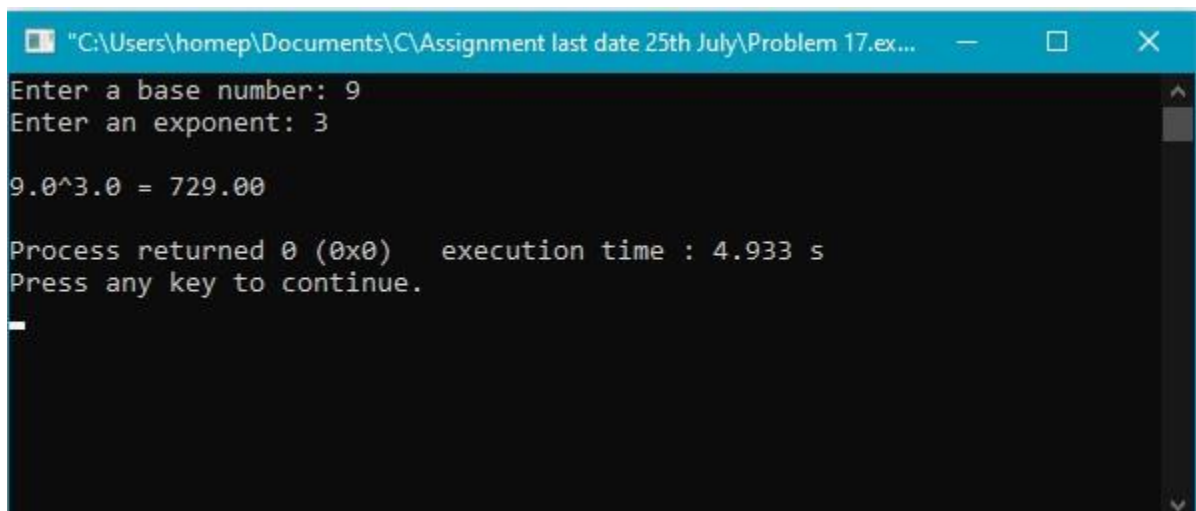
## Problem-17 :

### ➤ To Calculate the Power of a Number

#### CODE :

```
/*  
To Calculate the Power of a Number  
*/  
#include <math.h>  
#include <stdio.h>  
int main()  
{  
    double base,exp,result;  
  
    printf("Enter a base number: ");  
    scanf("%lf", &base);  
  
    printf("Enter an exponent: ");  
    scanf("%lf", &exp);  
  
    // power formula  
    result = pow(base,exp);  
  
    printf("\n%.1lf^%.1lf = %.2lf\n", base, exp, result);  
    return 0;  
}
```

#### RESULT :



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 17.ex...  
Enter a base number: 9  
Enter an exponent: 3  
  
9.0^3.0 = 729.00  
  
Process returned 0 (0x0)   execution time : 4.933 s  
Press any key to continue.  
_
```

## Problem-18 :

### ➤ C Program to Check Whether a Number is Palindrome or Not.

#### CODE :

```
/*
To Check Whether a Number is Palindrome or Not
*/
#include <stdio.h>
int main()
{
    int n,reversed = 0,remainder,original;

    printf("Enter an integer: ");
    scanf("%d", &n);

    original = n;

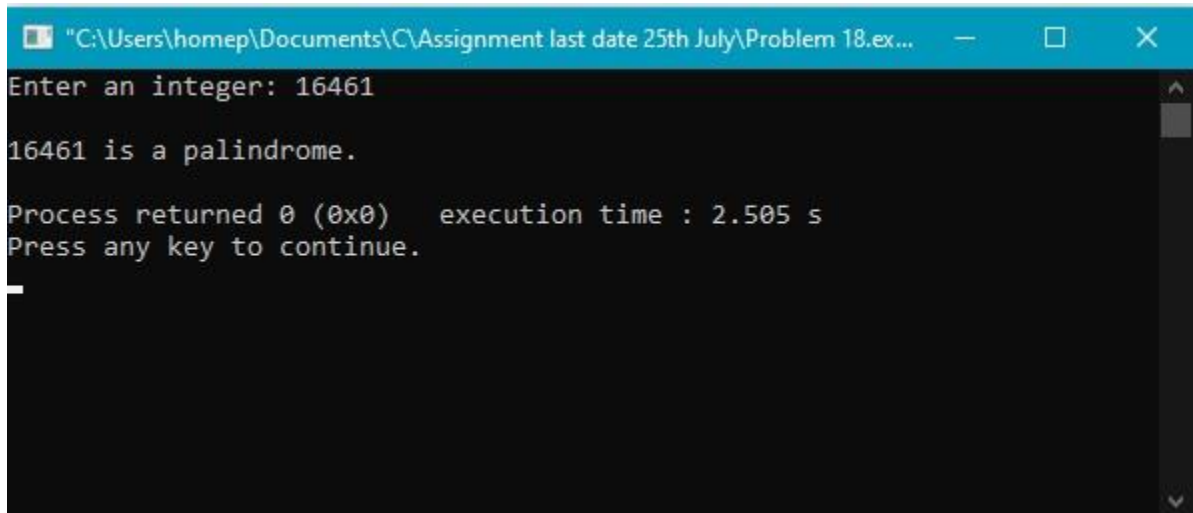
    // reversed integer is stored in reversed variable
    while (n != 0)
    {
        remainder = n % 10;
        reversed = reversed * 10 + remainder;
        n /= 10;
    }
    // palindrome if original and reversed are equal
    if (original == reversed)
        printf("\n%d is a palindrome.\n",original);
    else
        printf("\n%d is not a palindrome.\n",original);

    return 0;
}
```



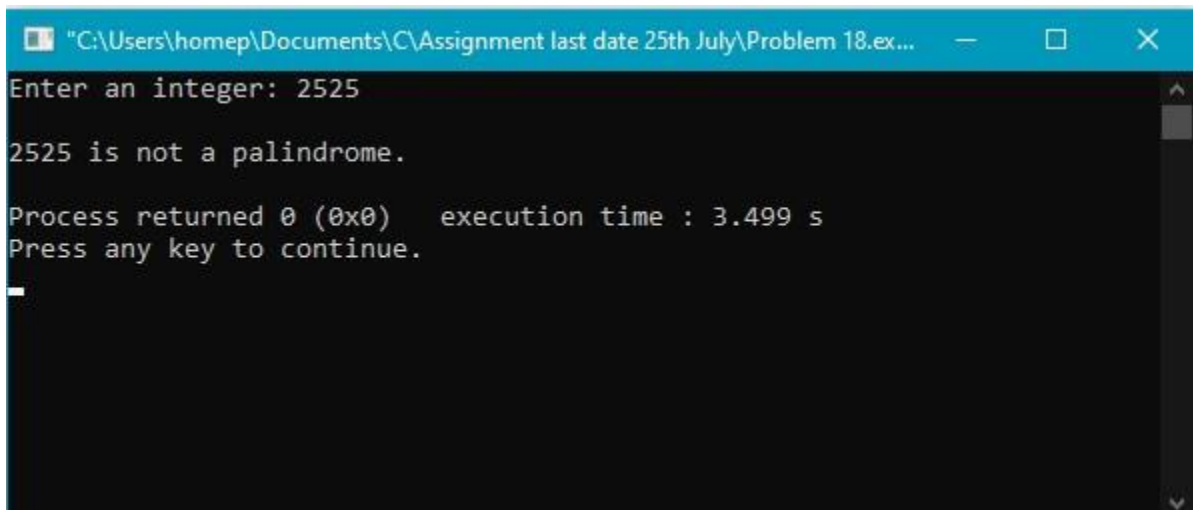
## RESULT:

### Case-1



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 18.ex...  
Enter an integer: 16461  
16461 is a palindrome.  
Process returned 0 (0x0)   execution time : 2.505 s  
Press any key to continue.  
_
```

### Case-2



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 18.ex...  
Enter an integer: 2525  
2525 is not a palindrome.  
Process returned 0 (0x0)   execution time : 3.499 s  
Press any key to continue.  
_
```

## Problem-19 :

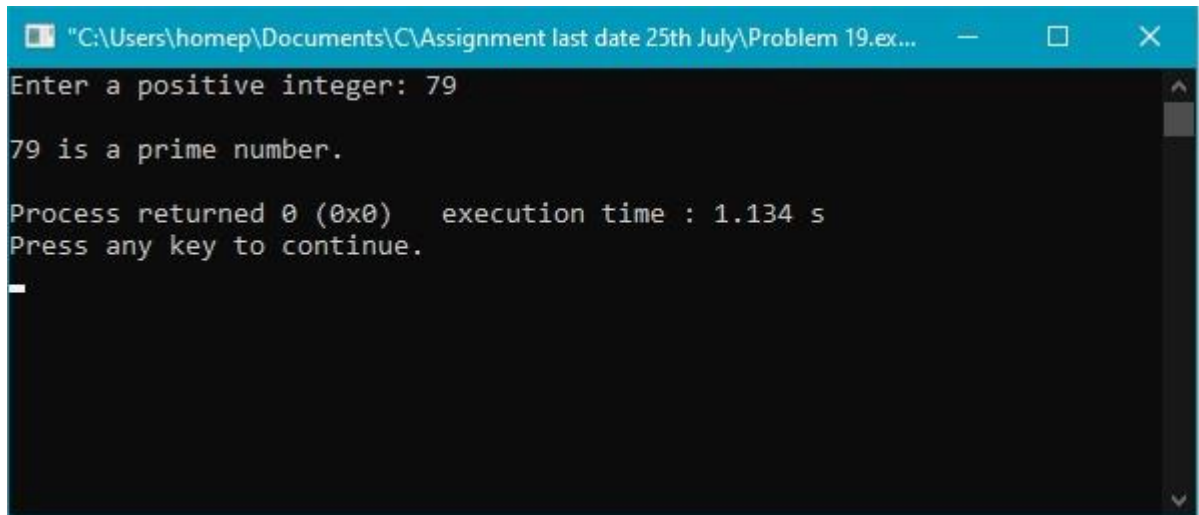
➤ C Program to Check Whether a Number is Prime or Not.

CODE :

```
/*  
To Check Whether a Number is Prime or Not  
*/  
#include <stdio.h>  
  
int main() {  
  
    int n,i,flag = 0;  
  
    printf("Enter a positive integer: ");  
    scanf("%d", &n);  
  
    // 0 and 1 are not prime numbers  
    // change flag to 1 for non-prime number  
    if (n == 0 || n == 1)  
        flag = 1;  
  
    for (i = 2; i <= n / 2; ++i)  
    {  
  
        // if n is divisible by i, then n is not prime  
        // change flag to 1 for non-prime number  
        if (n % i == 0)  
        {  
            flag = 1;  
            break;  
        }  
    }  
    // flag is 0 for prime numbers  
    if (flag == 0)  
        printf("\n%d is a prime number.\n", n);  
  
    else  
        printf("\n%d is not a prime number.\n", n);  
  
    return 0;  
}
```

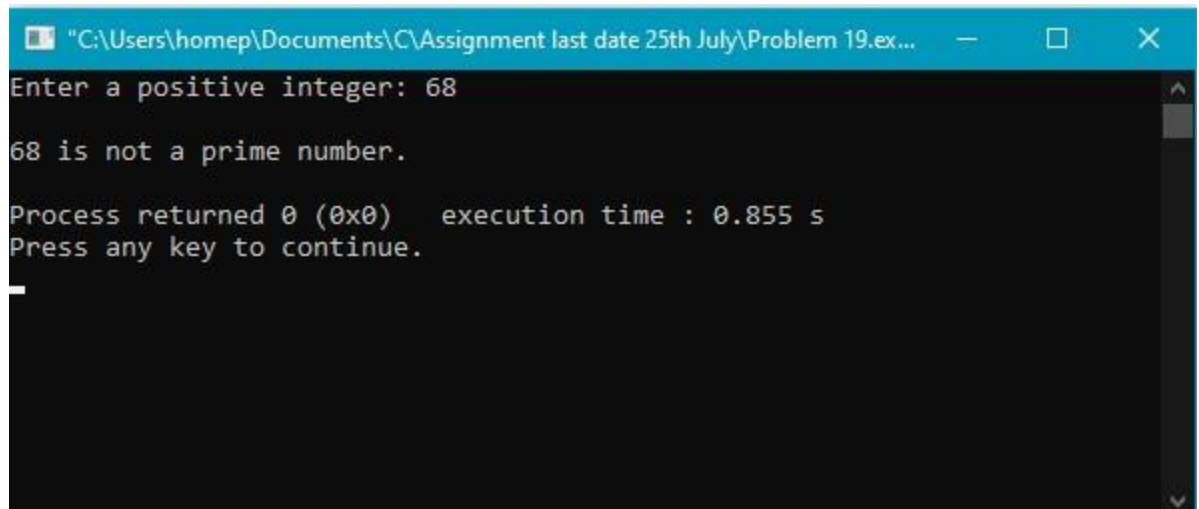
## RESULT:

### Case-1



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 19.ex...  
Enter a positive integer: 79  
79 is a prime number.  
Process returned 0 (0x0)   execution time : 1.134 s  
Press any key to continue.  
_
```

### Case-2



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 19.ex...  
Enter a positive integer: 68  
68 is not a prime number.  
Process returned 0 (0x0)   execution time : 0.855 s  
Press any key to continue.  
_
```

## Problem-20 :

### ➤ C Program to Check Armstrong Number.

#### CODE :

```
/*
C Program to Check Armstrong Number
*/
#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;

    for (originalNum = num; originalNum != 0; ++n)
    {
        originalNum /= 10;
    }

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

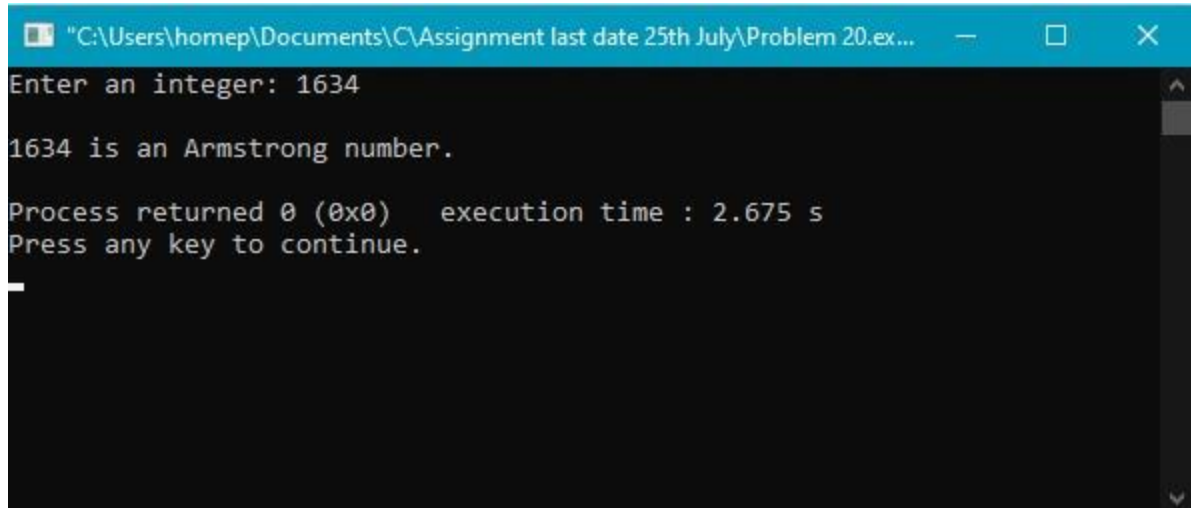
        result += pow(remainder, n);
    }

    if ((int)result == num)
        printf("\n%d is an Armstrong number.\n", num);

    else
        printf("\n%d is not an Armstrong number.\n", num);

    return 0;
}
```

## RESULT:



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 20.ex...  
Enter an integer: 1634  
1634 is an Armstrong number.  
Process returned 0 (0x0)   execution time : 2.675 s  
Press any key to continue.  
_
```

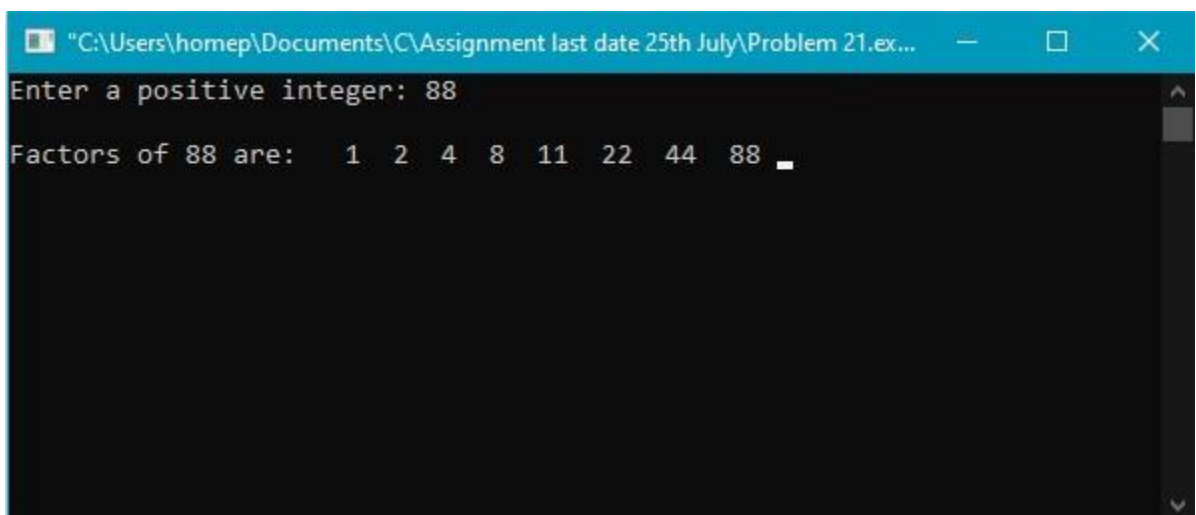
## Problem-21 :

### ➤ C Program to Display Factors of a Number.

#### CODE :

```
/*  
To Display Factors of a Number.  
*/  
#include <stdio.h>  
int main() {  
    int num,i;  
  
    printf("Enter a positive integer: ");  
    scanf("%d", &num);  
  
    printf("\nFactors of %d are: ", num);  
    for (i = 1; i <= num; ++i) {  
        if (num % i == 0)  
        {  
            printf(" %d ", i);  
        }  
    }  
    getch();  
}
```

#### RESULT :



The screenshot shows a Windows command prompt window titled "C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 21.ex...". The prompt displays the text "Enter a positive integer: 88". Below this, it shows the output "Factors of 88 are: 1 2 4 8 11 22 44 88" followed by a cursor. The window has a standard Windows title bar with minimize, maximize, and close buttons.

## Problem-22 :

### ➤ C Program to Make a Simple Calculator Using switch...case

#### CODE :

```
/*
To Make a Simple Calculator Using switch...case
*/
#include <stdio.h>

int main() {

    char op;
    double first,second;

    printf("Enter an operator (+, -, *, /): ");
    scanf("%c", &op);

    printf("Enter 1st operands: ");
    scanf("%lf",&first);

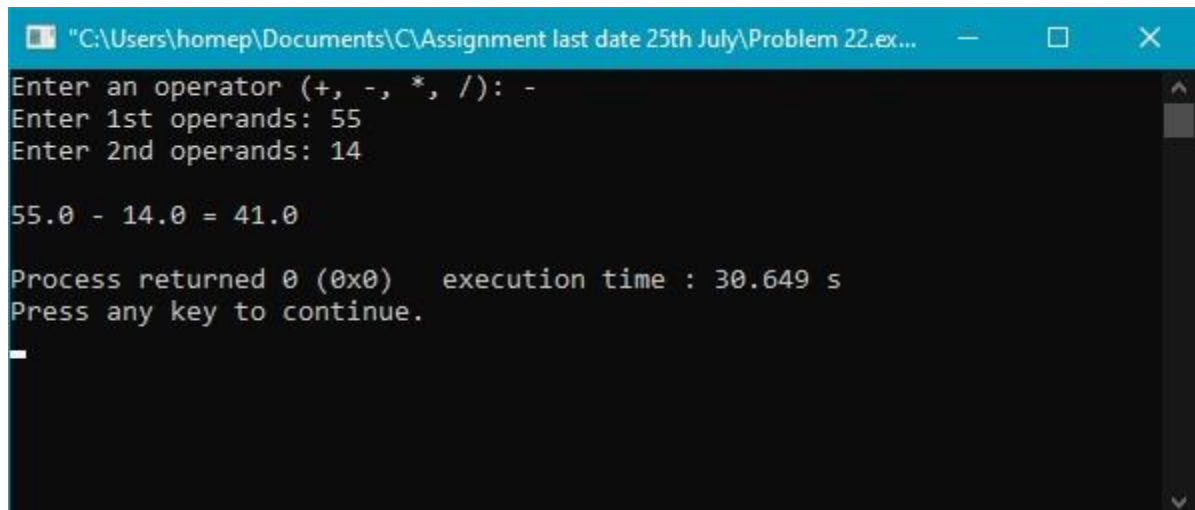
    printf("Enter 2nd operands: ");
    scanf("%lf",&second);

    switch (op)
    {
        case '+':
            printf("\n%.1lf + %.1lf = %.1lf\n", first, second, first + second);
            break;
        case '-':
            printf("\n%.1lf - %.1lf = %.1lf\n", first, second, first - second);
            break;
        case '*':
            printf("\n%.1lf * %.1lf = %.1lf\n", first, second, first * second);
            break;
        case '/':
            printf("\n%.1lf / %.1lf = %.1lf\n", first, second, first / second);
            break;

        default:
            printf("Error! operator is not correct");
    }

    return 0;
}
```

## RESULT:



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 22.ex...  
Enter an operator (+, -, *, /): -  
Enter 1st operands: 55  
Enter 2nd operands: 14  
  
55.0 - 14.0 = 41.0  
  
Process returned 0 (0x0)   execution time : 30.649 s  
Press any key to continue.  
_
```



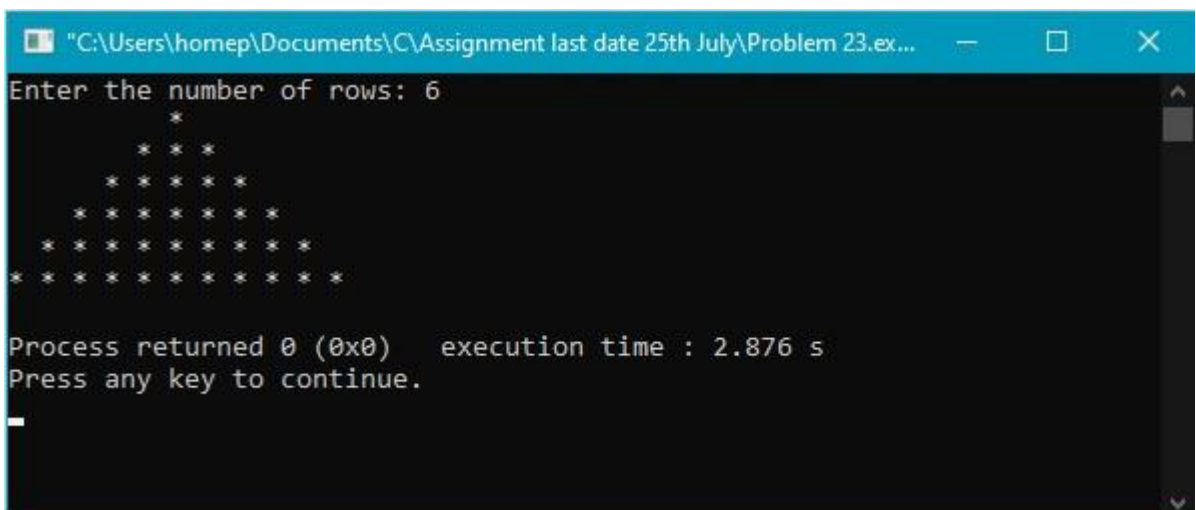
## Problem-23 :

### ➤ C Program to Print Pyramids and Patterns

#### CODE :

```
/*
C Program to Print Pyramids and Patterns.
*/
//Full Pyramid of *//
#include <stdio.h>
int main() {
    int i, space, rows, k = 0;
    printf("Enter the number of rows: ");
    scanf("%d", &rows);
    for (i = 1; i <= rows; ++i, k = 0)
    {
        for (space = 1; space <= rows - i; ++space)
        {
            printf(" ");
        }
        while (k != 2 * i - 1)
        {
            printf("* ");
            ++k;
        }
        printf("\n");
    }
    return 0;
}
```

#### RESULT:



The screenshot shows a Windows command prompt window titled "C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 23.ex...". The prompt displays the output of the C program. It first asks "Enter the number of rows: 6". Below this, it prints a full pyramid of asterisks with 6 rows. The first row has 1 asterisk, the second has 3, the third has 5, the fourth has 7, the fifth has 9, and the sixth has 11. At the bottom, it shows "Process returned 0 (0x0) execution time : 2.876 s" and "Press any key to continue." followed by a cursor.

## Problem-24 :

- Write a C program to accept a coordinate point in an XY coordinate system and determine which quadrant the coordinate point lies.

### CODE :

```
/*
Write a C program to accept a coordinate point in an XY coordinate system and
determine which quadrant the coordinate point lies.
*/
#include <stdio.h>
int main()
{
    int x,y;
    printf("Input the coordinate(x,y): ");
    scanf("%d %d",&x,&y);

    if (x>0 && y>0)
        printf("\nThe coordinate point (%d,%d) lies in First Quadrant.\n",x,y);

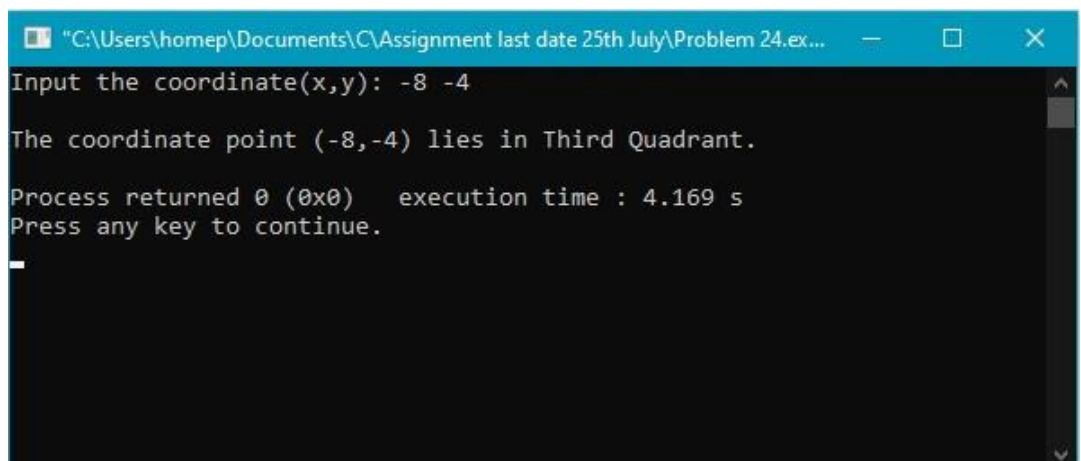
    else if (x<0 && y>0)
        printf("\nThe coordinate point (%d,%d) lies in second Quadrant.\n",x,y);

    else if (x<0 && y<0)
        printf("\nThe coordinate point (%d,%d) lies in Third Quadrant.\n",x,y);

    else if (x>0 && y<0)
        printf("\nThe coordinate point (%d,%d) lies in Forth Quadrant.\n",x,y);

    return 0;
}
```

### RESULT :



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 24.ex...
Input the coordinate(x,y): -8 -4
The coordinate point (-8,-4) lies in Third Quadrant.
Process returned 0 (0x0)   execution time : 4.169 s
Press any key to continue.
_
```

## Problem-25 :

- Write a program in C to read n number of values in an array and display it in reverse order.

CODE :

```
/*
Write a program in C to read n number of values in an
array and display it in reverse order.
*/
#include <stdio.h>
int main()
{
    int i,n,a[100];

    printf("\n\nRead n number of values in an array and display it in reverse order:\n");

    printf("\nInput the number of elements to store in the array : ");
    scanf("%d",&n);

    printf("\nInput %d number of elements in the array :\n",n);
    for(i=0;i<n;i++)
    {
        printf("element - %d : ",i);
        scanf("%d",&a[i]);
    }
    printf("\nThe values store into the array are : \n");

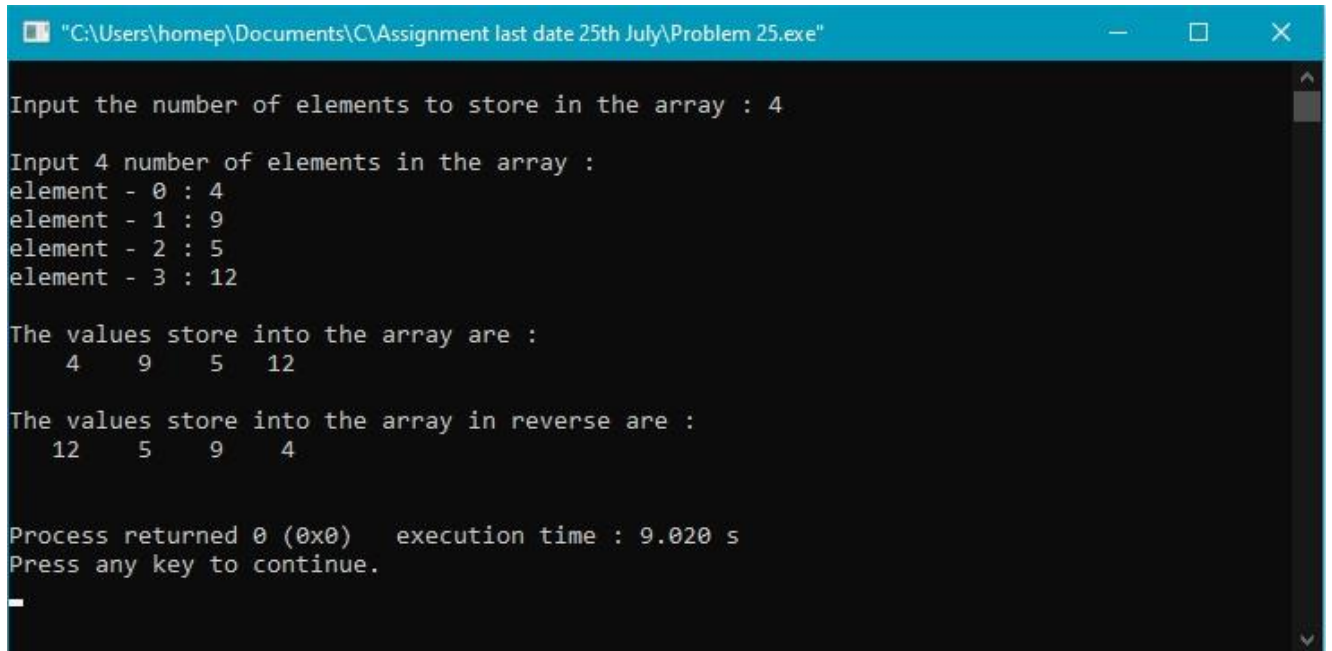
    for(i=0;i<n;i++)
    {
        printf("% 5d",a[i]);
    }

    printf("\n\nThe values store into the array in reverse are :\n");

    for(i=n-1;i>=0;i--)
    {
        printf("% 5d",a[i]);
    }

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

## RESULT :



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 25.exe"

Input the number of elements to store in the array : 4

Input 4 number of elements in the array :
element - 0 : 4
element - 1 : 9
element - 2 : 5
element - 3 : 12

The values store into the array are :
    4    9    5   12

The values store into the array in reverse are :
   12    5    9    4

Process returned 0 (0x0)   execution time : 9.020 s
Press any key to continue.
_
```

## Problem-26 :

➤ Write a program in C to find the sum of all elements of the array.

### CODE :

```
/*
Write a program in C to find the sum of all elements of the array.
*/
#include<stdio.h>
int main()
{
    //let's assume the maximum array size as 100.
    int arr[100],size,i,sum = 0;

    //Get size input from user
    printf("Enter array size = ");
    scanf("%d",&size);

    //Get all elements using for loop and store it in array
    printf("Enter array elements = ");

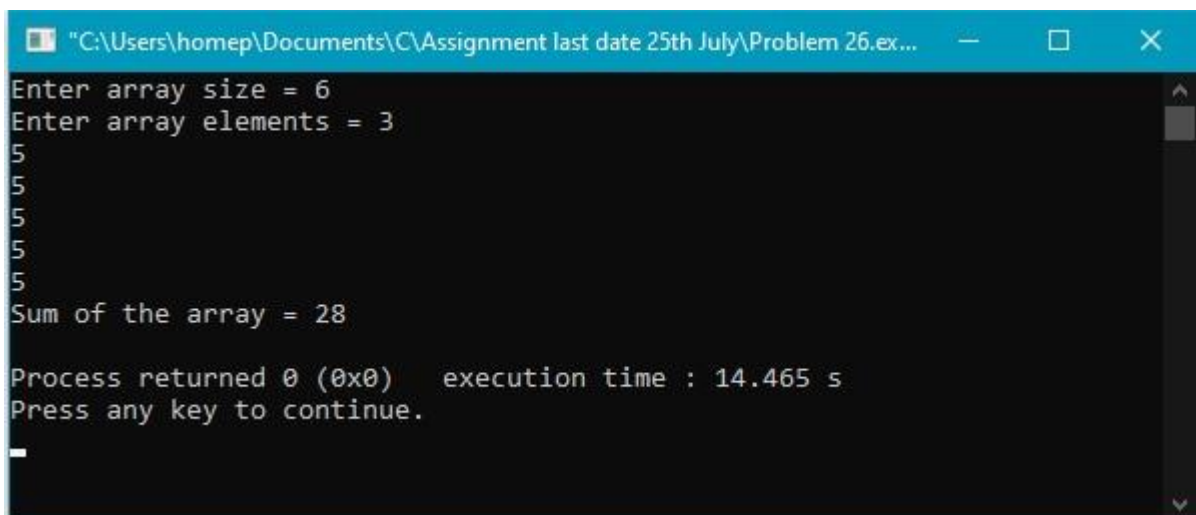
    for(i = 0; i < size; i++)
        scanf("%d",&arr[i]);

    //add all elements to the variable sum.
    for(i = 0; i < size; i++)
        sum = sum + arr[i]; // same as sum += arr[i];

    //print the result
    printf("Sum of the array = %d\n",sum);

    return 0;
}
```

### RESULT :



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 26.ex...
Enter array size = 6
Enter array elements = 3
5
5
5
5
5
5
Sum of the array = 28

Process returned 0 (0x0)   execution time : 14.465 s
Press any key to continue.
_
```

## Problem-27 :

- Write a program in C to count the total number of duplicate elements in an array

CODE :

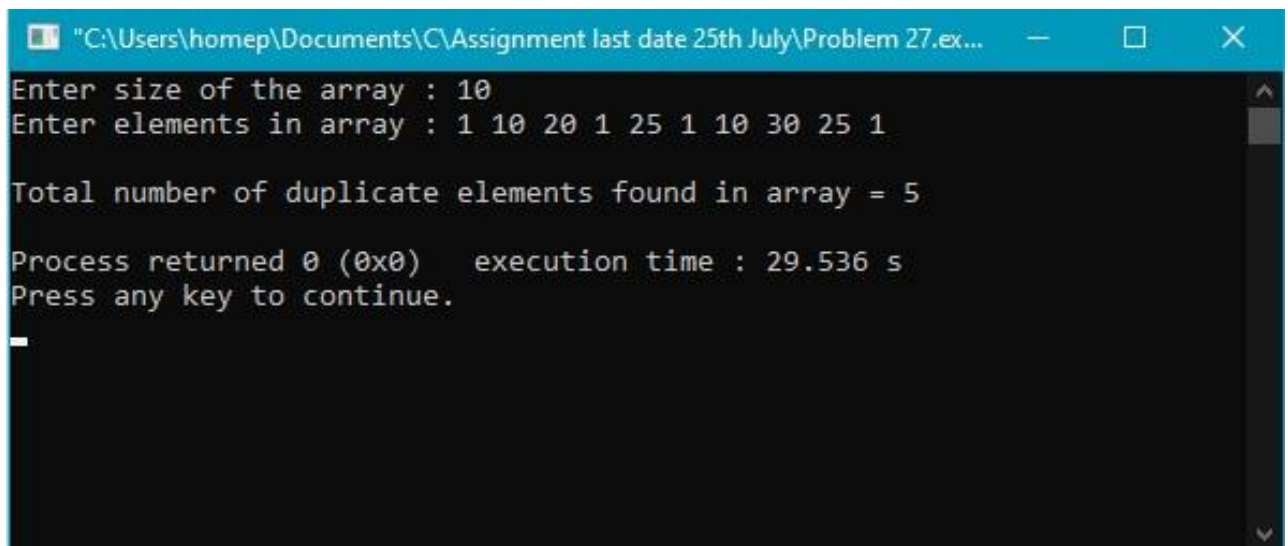
```
/*
C program to count total number of duplicate elements in an array
*/
#include <stdio.h>
#define MAX_SIZE 100 // Maximum array size
int main()
{
    int arr[MAX_SIZE];
    int i,j,size,count = 0;

    // Input size of array //
    printf("Enter size of the array : ");
    scanf("%d", &size);

    //Input elements in array //
    printf("Enter elements in array : ");
    for(i=0; i<size; i++)
    {
        scanf("%d", &arr[i]);
    }
    //Find all duplicate elements in array//
    for(i=0; i<size; i++)
    {
        for(j=i+1; j<size; j++)
        {
            //If duplicate found then increment count by 1 //
            if(arr[i] == arr[j])
            {
                count++;
                break;
            }
        }
    }
    printf("\nTotal number of duplicate elements found in array = %d\n", count);

    return 0;
}
```

## RESULT:



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 27.ex...  
Enter size of the array : 10  
Enter elements in array : 1 10 20 1 25 1 10 30 25 1  
  
Total number of duplicate elements found in array = 5  
  
Process returned 0 (0x0)   execution time : 29.536 s  
Press any key to continue.  
_
```

## Problem-28 :

➤ Write a program in C to find the sum of all elements of the array.

CODE :

```
/*
Write a program in C to find an array's maximum and minimum elements.
*/
#include <stdio.h>
void main()
{
    int arr1[100];
    int i, mx, mn, n;

    printf("Find maximum and minimum element in an array :\n");

    printf("Input the number of elements to be stored in the array = ");
    scanf("%d",&n);

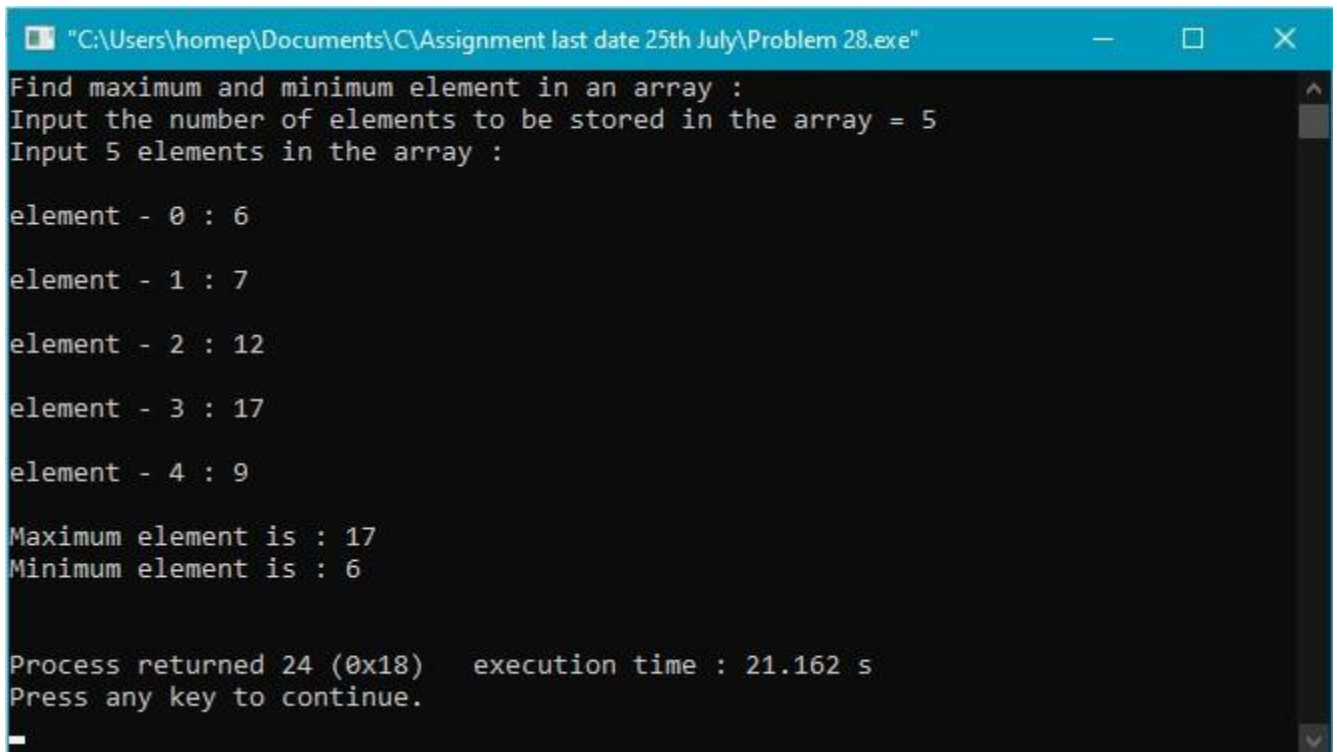
    printf("Input %d elements in the array :\n",n);
    for(i=0;i<n;i++)
    {
        printf("\nelement - %d : ",i);
        scanf("%d",&arr1[i]);
    }
    mx = arr1[0];
    mn = arr1[0];

    for(i=1; i<n; i++)
    {
        if(arr1[i]>mx)
        {
            mx = arr1[i];
        }

        if(arr1[i]<mn)
        {
            mn = arr1[i];
        }
    }
    printf("\nMaximum element is : %d\n", mx);
    printf("Minimum element is : %d\n\n", mn);
}
```



## RESULT:



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 28.exe"
Find maximum and minimum element in an array :
Input the number of elements to be stored in the array = 5
Input 5 elements in the array :

element - 0 : 6
element - 1 : 7
element - 2 : 12
element - 3 : 17
element - 4 : 9

Maximum element is : 17
Minimum element is : 6

Process returned 24 (0x18)    execution time : 21.162 s
Press any key to continue.
_
```

## Problem-29 :

➤ Write a program in C to insert a new value in the array (unsorted list )

### CODE :

```
/*
Write a program in C to insert a new value in the array-unsorted list
*/
#include <stdio.h>
int main()
{
    int arr1[100],i,n,p,x;

    printf("Insert New value in the unsorted array : ");
    printf("Input the size of array : ");
    scanf("%d", &n);
    /* Stored values into the array*/
    printf("Input %d elements in the array in ascending order:\n",n);
    for(i=0;i<n;i++)
    {
        printf("\nelement - %d : ",i);
        scanf("%d",&arr1[i]);
    }

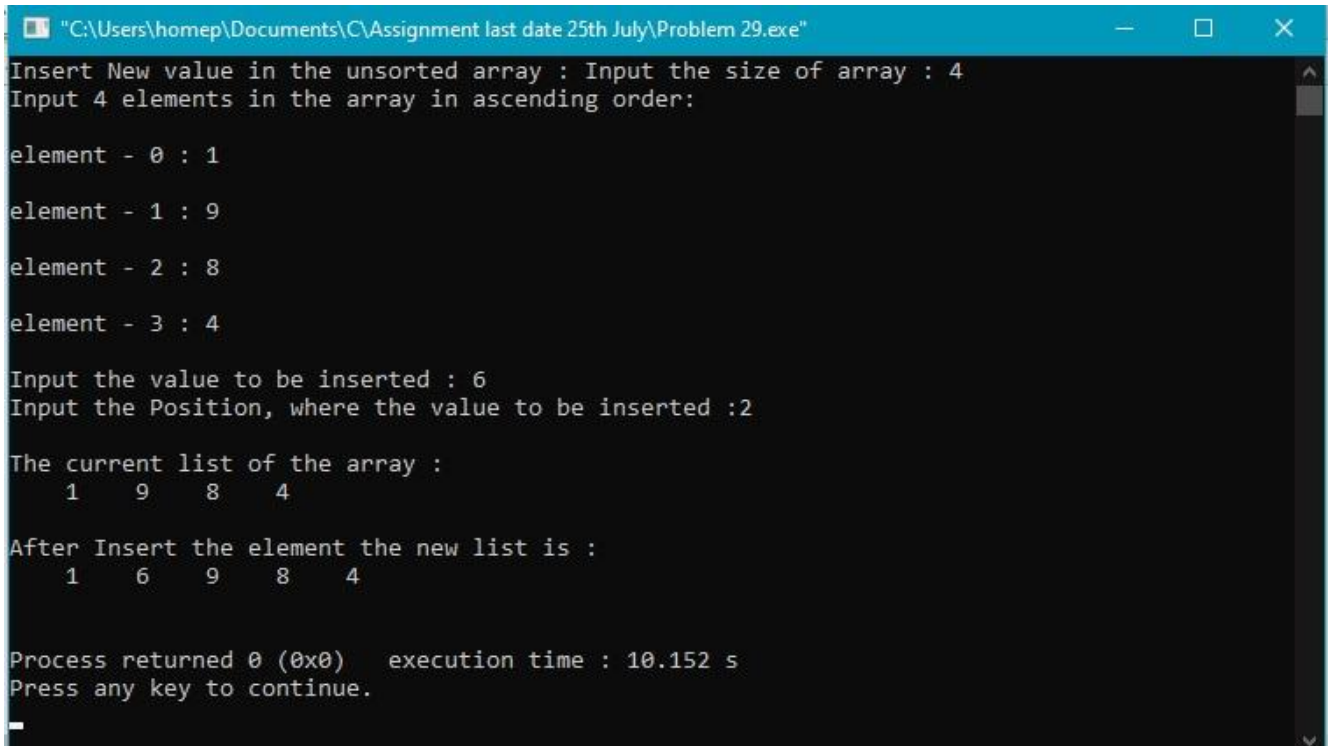
    printf("\nInput the value to be inserted : ");
    scanf("%d",&x);
    printf("Input the Position, where the value to be inserted :");
    scanf("%d",&p);

    printf("\nThe current list of the array :\n");
    for(i=0;i<n;i++)
        printf("% 5d",arr1[i]);
    /* Move all data at right side of the array */
    for(i=n;i>=p;i--)
        arr1[i]= arr1[i-1];
    /* insert value at given position */
    arr1[p-1]=x;

    printf("\n\nAfter Insert the element the new list is :\n");

    for(i=0;i<=n;i++)
        printf("% 5d",arr1[i]);
    printf("\n\n");
}
```

## RESULT :



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 29.exe"
Insert New value in the unsorted array : Input the size of array : 4
Input 4 elements in the array in ascending order:

element - 0 : 1
element - 1 : 9
element - 2 : 8
element - 3 : 4

Input the value to be inserted : 6
Input the Position, where the value to be inserted :2

The current list of the array :
    1    9    8    4

After Insert the element the new list is :
    1    6    9    8    4

Process returned 0 (0x0)   execution time : 10.152 s
Press any key to continue.
_
```

## Problem-30 :

- Write a program in C to delete an element at the desired position from an array.

CODE :

```
/*
Write a program in C to delete an element at the desired position from an array.
*/
#include <stdio.h>
int main()
{
    int arr1[50],i,pos,n;

    printf("Delete an element at desired position from an array :\n");
    printf("Input the size of array : ");
    scanf("%d", &n);
    //Stored values into the array//
    printf("Input %d elements in the array in ascending order:\n",n);
    for(i=0;i<n;i++)
    {
        printf("\nelement - %d : ",i);
        scanf("%d",&arr1[i]);
    }

    printf("\nInput the position where to delete: ");
    scanf("%d",&pos);
    //locate the position of i in the array//
    i=0;

    while(i!=pos-1)
        i++;
    /*the position of i in the array will be replaced by the
    value of its right */

    while(i<n)
    {
        arr1[i]=arr1[i+1];
        i++;
    }
    n--;
    printf("\nThe new list is : ");
    for(i=0;i<n;i++)
    {
        printf(" %d",arr1[i]);
    }
    printf("\n\n");
}
```

## RESULT :

"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 30.exe"

Delete an element at desired position from an array :

Input the size of array : 6

Input 6 elements in the array in ascending order:

element - 0 : 8

element - 1 : 6

element - 2 : 7

element - 3 : 1

element - 4 : 9

element - 5 : 6

Input the position where to delete: 1

The new list is : 6 7 1 9 6

Process returned 0 (0x0) execution time : 15.191 s

Press any key to continue.

## Problem-31 :

- Write a program in C to find an element in a given array (Linear search)

CODE :

```
/*
Write a program in C to find an element in a given array (Linear search)
*/
#include <stdio.h>
int main()
{
    int array[100], search, c, n;

    printf(">>Enter number of elements in array = ");
    scanf("%d", &n);

    printf("\n>>Enter %d integer(s)\n", n);

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

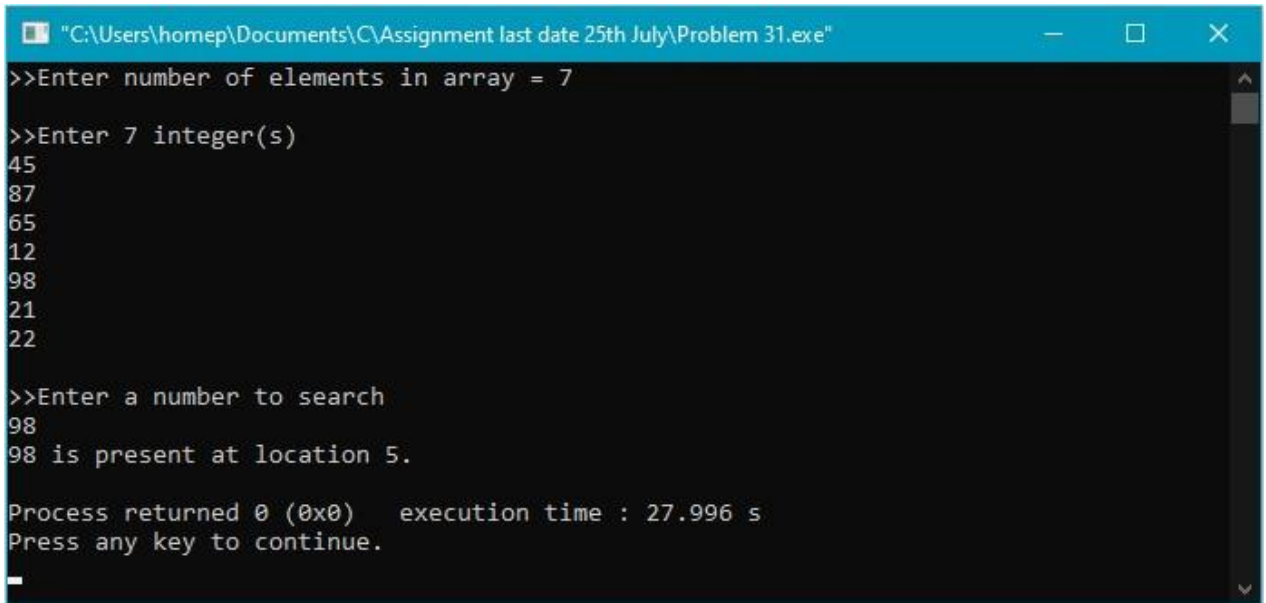
    printf("\n>>Enter a number to search\n");
    scanf("%d", &search);

    for (c = 0; c < n; c++)
    {
        if (array[c] == search)
        {
            printf("%d is present at location %d.\n", search, c+1);
            break;
        }
    }
    if (c == n)
        printf("%d isn't present in the array.\n", search);

    return 0;
}
```

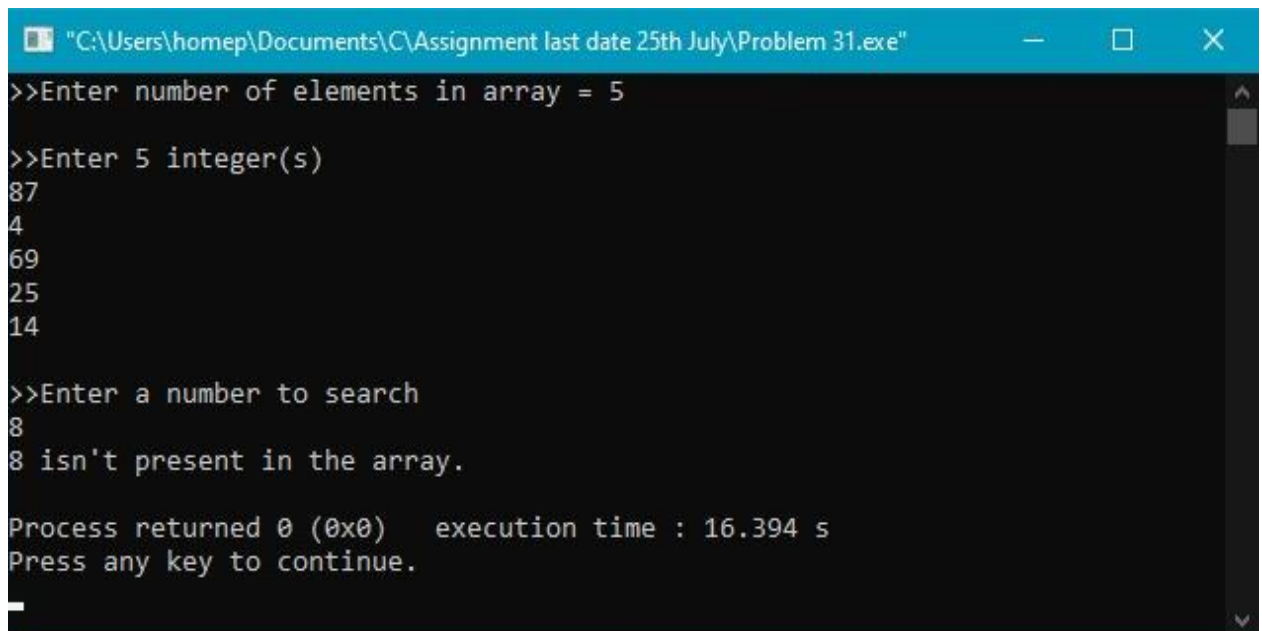
## RESULT :

### Case-1



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 31.exe"
>>Enter number of elements in array = 7
>>Enter 7 integer(s)
45
87
65
12
98
21
22
>>Enter a number to search
98
98 is present at location 5.
Process returned 0 (0x0)   execution time : 27.996 s
Press any key to continue.
```

### Case-2



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 31.exe"
>>Enter number of elements in array = 5
>>Enter 5 integer(s)
87
4
69
25
14
>>Enter a number to search
8
8 isn't present in the array.
Process returned 0 (0x0)   execution time : 16.394 s
Press any key to continue.
```

## Problem-32 :

- Write a program in C to display the sum of the series  
[  $1+x+x^2/2!+x^3/3!+....$  ]

### CODE :

```
/*
Write a program in C to display the sum of the series [  $1+x+x^2/2!+x^3/3!+....$  ].
*/
#include <stdio.h>
int main()
{
    float x,sum,no_row;
    int i,n;

    printf("Input the value of x :");
    scanf("%f",&x);

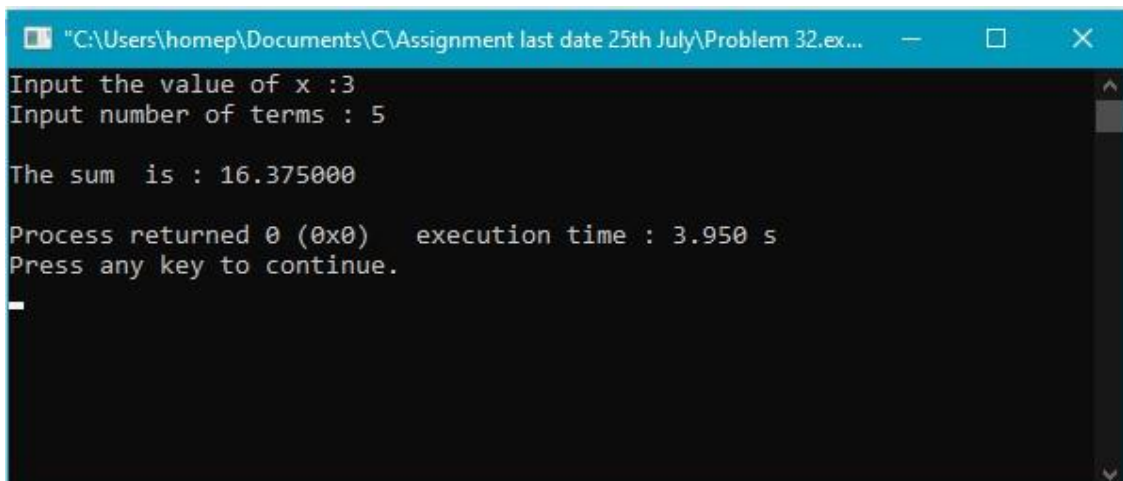
    printf("Input number of terms : ");
    scanf("%d",&n);

    sum =1; no_row = 1;

    for (i=1; i<n; i++)
    {
        no_row = no_row*x/(float)i;
        sum =sum+ no_row;
    }
    printf("\nThe sum is : %f\n",sum);

    return 0;
}
```

### RESULT :



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 32.ex...
Input the value of x :3
Input number of terms : 5

The sum is : 16.375000

Process returned 0 (0x0) execution time : 3.950 s
Press any key to continue.
_
```



### Problem-33 :

#### ➤ **C Program to Convert Binary Number to Decimal and vice-versa.**

CODE :

```
/*
C Program to Convert Binary Number to Decimal and vice-versa
*/
#include <stdio.h>
#include <math.h>
int main()
{
    int convert(long long);
    long long n;

    printf("Enter a binary number: ");
    scanf("%lld", &n);

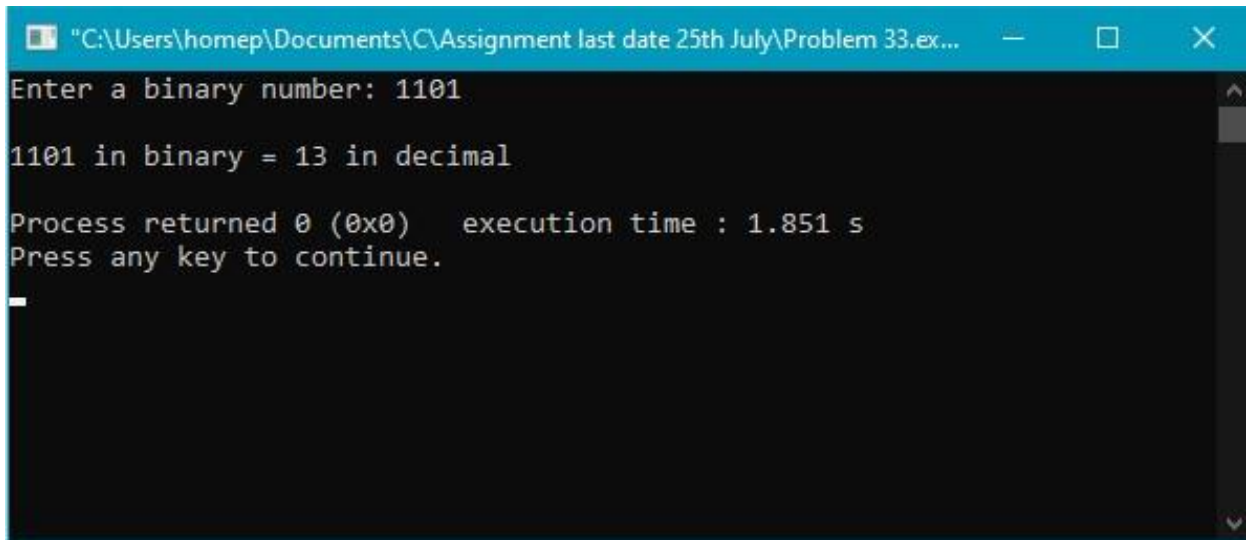
    printf("\n%lld in binary = %d in decimal\n",n,convert(n));
    return 0;
}

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;
}
```

## RESULT:



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 33.ex...  
Enter a binary number: 1101  
  
1101 in binary = 13 in decimal  
  
Process returned 0 (0x0)   execution time : 1.851 s  
Press any key to continue.  
_
```

## Problem-34 :

### ➤ C Program to Convert Octal Number to Decimal and vice-versa

CODE :

```
/*
C Program to Convert Octal Number to Decimal and vice-versa
*/
#include <stdio.h>
#include <math.h>
int main()
{
    long long convertOctalToDecimal(int octalNumber);
    int octalNumber;

    printf("Enter an octal number: ");
    scanf("%d", &octalNumber);

    printf("\n%d in octal = %lld in decimal\n", octalNumber,
convertOctalToDecimal(octalNumber));

    return 0;
}

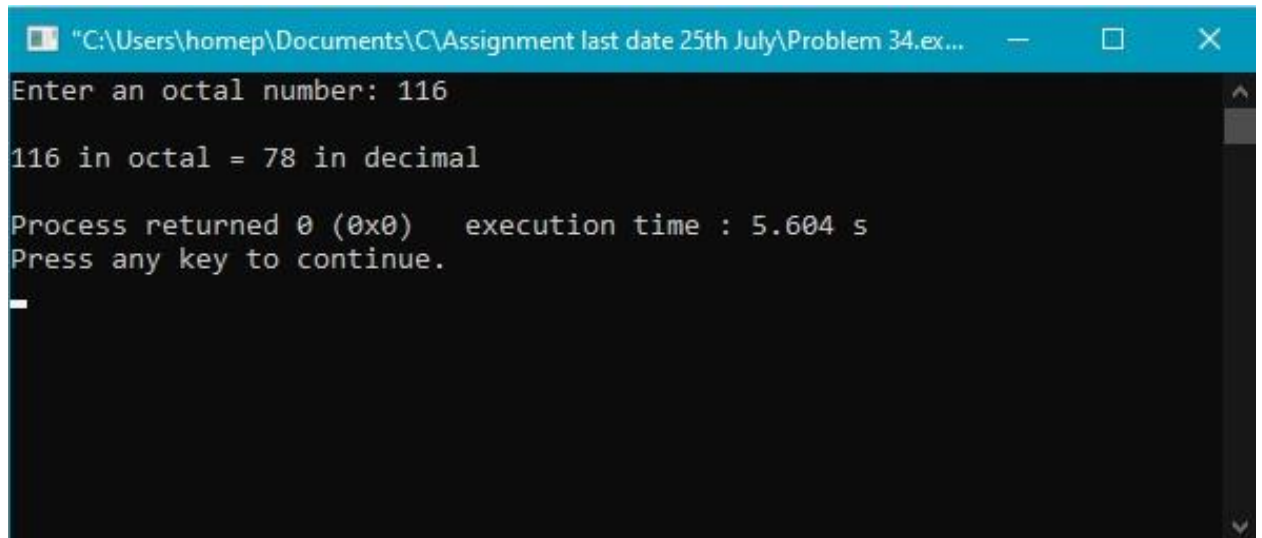
long long convertOctalToDecimal(int octalNumber)
{
    int decimalNumber = 0, i = 0;

    while(octalNumber != 0)
    {
        decimalNumber += (octalNumber%10) * pow(8,i);
        ++i;
        octalNumber/=10;
    }

    i = 1;

    return decimalNumber;
}
```

## RESULT:



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 34.ex...  —  □  ×  
Enter an octal number: 116  
116 in octal = 78 in decimal  
Process returned 0 (0x0)   execution time : 5.604 s  
Press any key to continue.  
_
```

## Problem-35 :

### ➤ C Program to Convert Binary Number to Octal and vice-versa

CODE :

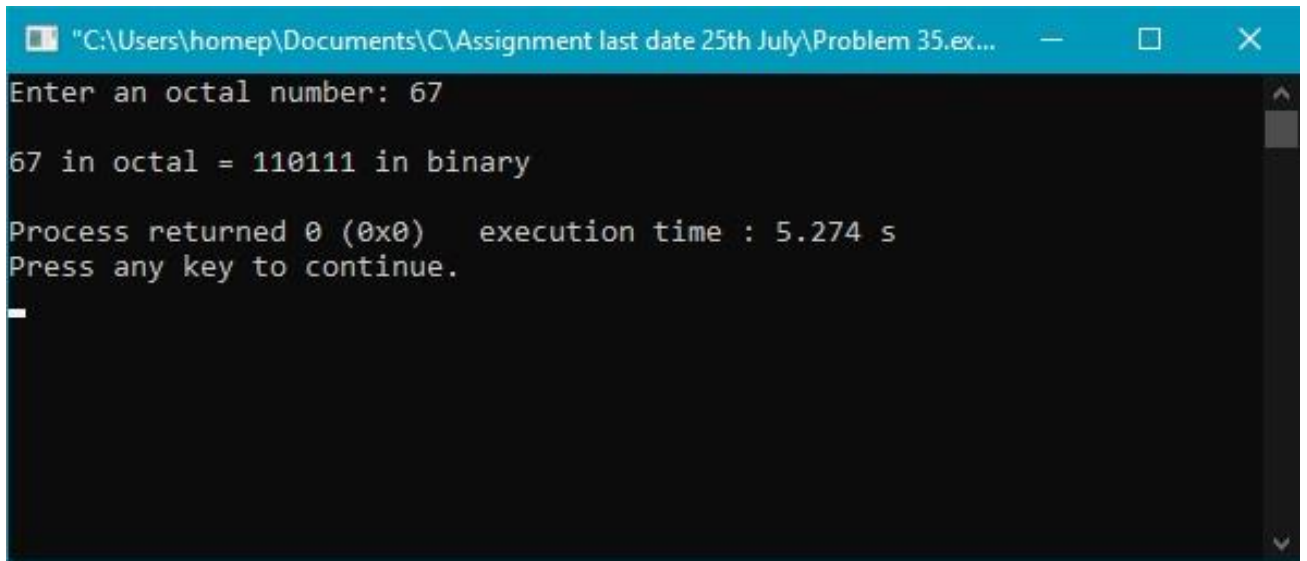
```
/*
C Program to Convert Binary Number to Octal and vice-versa
*/
#include <math.h>
#include <stdio.h>
int main()
{
    long long convert(int oct);
    int oct;
    printf("Enter an octal number: ");
    scanf("%d", &oct);
    printf("\n%d in octal = %lld in binary\n", oct, convert(oct));
    return 0;
}

long long convert(int oct) {
    int dec = 0, i = 0;
    long long bin = 0;

    while (oct != 0) {
        dec += (oct % 10) * pow(8, i);
        ++i;
        oct /= 10;
    }
    i = 1;

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

## RESULT:



```
"C:\Users\homep\Documents\C\Assignment last date 25th July\Problem 35.ex...  
Enter an octal number: 67  
67 in octal = 110111 in binary  
Process returned 0 (0x0)   execution time : 5.274 s  
Press any key to continue.  
_
```

.....ASSIGNMENT END.....