Practical Set 1 to 10

CE143 COMPUTER CONCEPTS & PROGRAMMING

RUSHIK RATHOD

20DCS103 DEPSTAR CSE

Set - 1

Practical - 1

Aim :

Write a C program that will output this passage by Michael Singer. Make sure your output looks exactly as shown here (including spacing, line breaks, punctuation, and the title and author). Use Required Escape Sequence and ASCII Value. There are three shapes in the output: Smiling Face, Diamond & Heart.

The ASCII Value for Smiling face is 1.

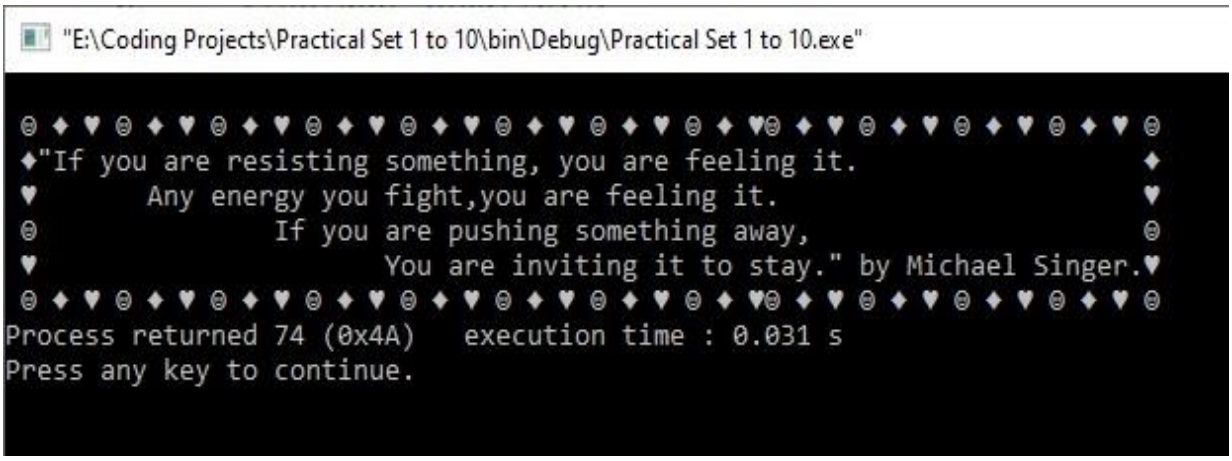
The ASCII Value for Diamond is is 4.

The ASCII Value for Heart is is 3.

Source Code :

```
#include <stdio.h>

#include <stdlib.h>

void main()

{

    printf("\n %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c
%c %c %c %c %c %c %c %c %c %c %c %c %c %c
%c”,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1);

    printf("\n %c\"If you are resisting something, you are feeling it.\t\t\t%c",4,4);

    printf("\n %c \t Any energy you fight,you are feeling it.\t\t\t%c",3,3);

    printf("\n %c\t\t If you are pushing something away,\t\t\t%c",1,1);

    printf("\n %c\t\t\tYou are inviting it to stay.\" by Michael Singer.%c",3,3);

    printf("\n %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c %c
%c %c%c %c %c %c %c %c %c %c %c %c %c %c
%c",1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1,4,3,1);

}
```

Output :



Practical - 2

Aim :

In a town, the percentage of men is 52. The percentage of total literacy is 48. If total percentage of literate men is 35 of the total population, write a program to find the total number of illiterate men and women if the population of the town is 80,000. Write Algorithms and Flowchart of this program.

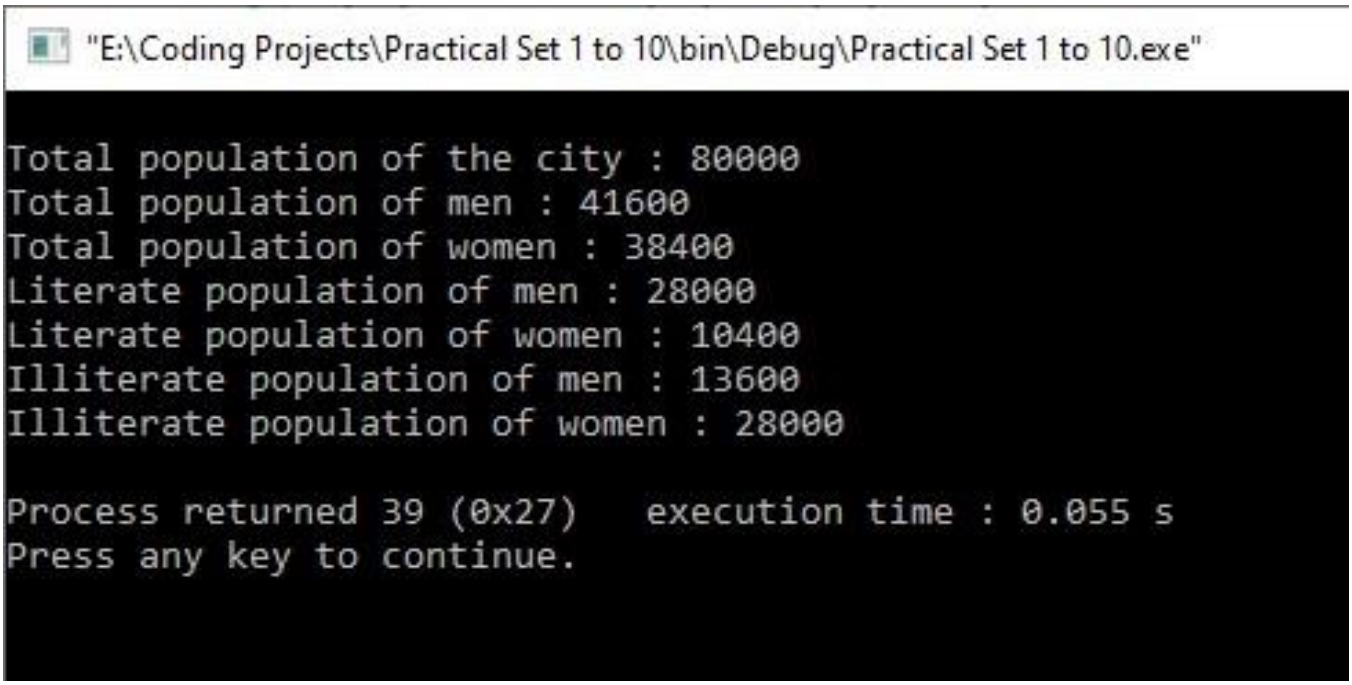
Source Code :

```
#include<stdio.h>

void main ()

{

 int pop, popmen, popwomen, poplit, poilit, litmen, ilitmen, litwomen, ilitwomen;

 pop = 80000;
```

```
//population of men
popmen = (52 * pop) / 100;

//population of women
popwomen = pop - popmen;

//literate population
poplit = (48 * pop) / 100;

//literate population of men
litmen = (35 * pop) / 100;

//literate population of women
litwomen = poplit - litmen;

//iliterate population of men
ilitmen = popmen - litmen;

//iliterate population of women
ilitwomen = popwomen - litwomen;

printf("\n");
printf("Total population of the city : %d\n",pop);
printf("Total population of men : %d\n",popmen);
printf("Total population of women : %d\n",popwomen);
printf("Literate population of men : %d\n",litmen);
printf("Literate population of women : %d\n",litwomen);
printf("Illiterate population of men : %d\n",ilitmen);
```

```
 printf("Illiterate population of women : %d\n",ilitwomen);
}
```

Output :



Practical - 3

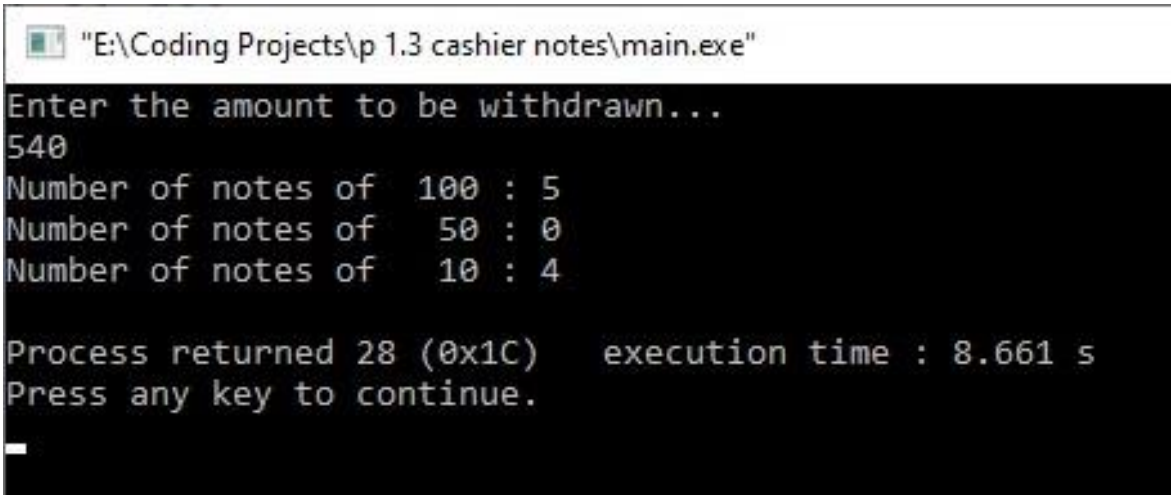
Aim :

A cashier has currency notes of denominations 10, 50 and 100. If the amount to be withdrawn is input through the keyboard in hundreds, find the total number of currency notes of each denomination the cashier will have to give to the withdrawer.

Source Code :

```
#include <stdio.h>
#include <stdlib.h>
void main()
{
   int amount;
   printf("Enter the amount to be withdrawn...\n");
   scanf("%d",&amount);

   printf("Number of notes of  100 : %d\n",amount/100);
   printf("Number of notes of   50 : %d\n",(amount%100)/50);
   printf("Number of notes of   10 : %d\n",((amount%100)%50)/10);
}
```

Output :



Set - 2

Practical - 1

Aim :

Write a program to calculate Net Salary. User has to input Basic Salary and Output should be:

Enter Basic Salary: 5000 (e.g. 5000)

Allowances:

DA = 70% of Basic Salary

HRA = 7% of Basic Salary

MA = 2% of Basic Salary

TA = 4% of Basic Salary Deduction:

PF = 12% of Basic Salary

IT = any value (e.g. 500)

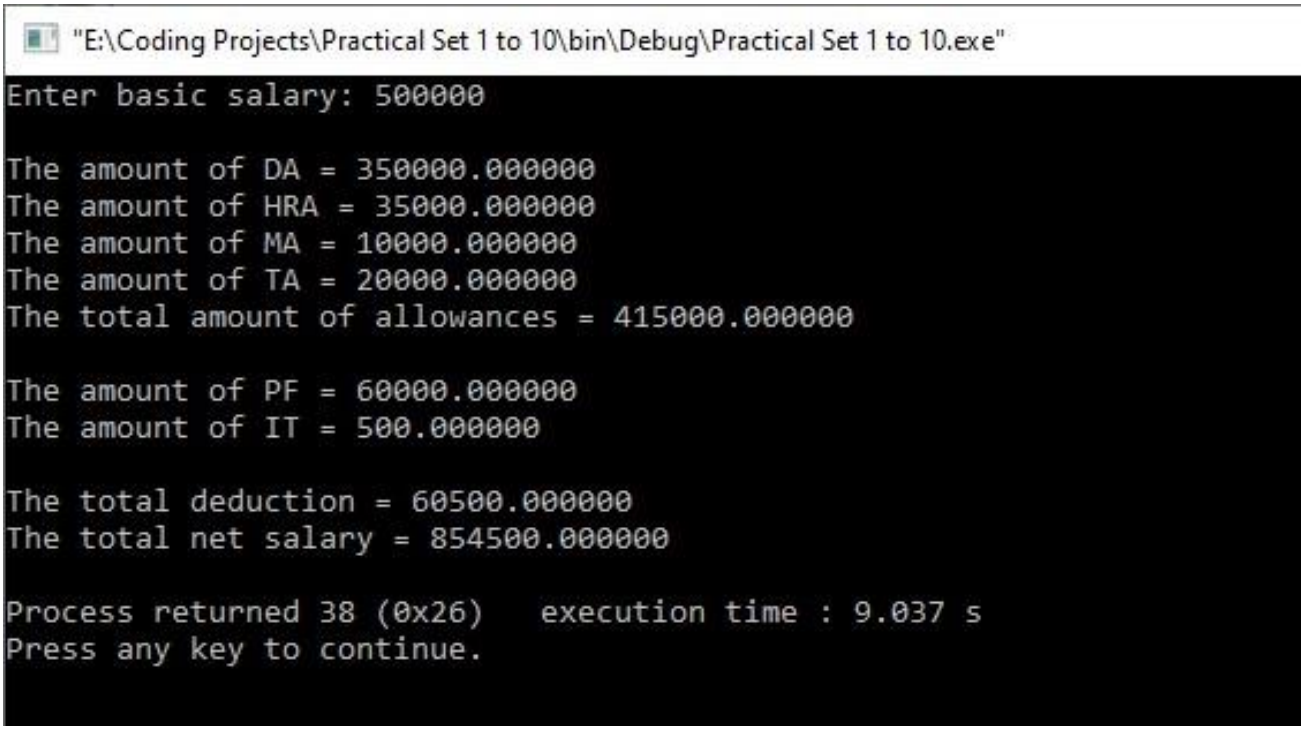
Net Salary = Basic Salary + Allowances – Deduction

Source Code :

```
#include <stdio.h>
#include <stdlib.h>
void main()
{
   float bs, da, hra, ma, ta, allowance, deduction, pf, it, net;
   printf ("Enter basic salary: ",bs);
   scanf ("%f",&bs);

   da=(70*bs)/100;
   hra=(7*bs)/100;
   ma=(2*bs)/100;
```

```
    ta=(4*bs)/100;
    allowance=da+hra+ma+ta;

    printf("\nThe amount of DA = %f",da);
    printf("\nThe amount of HRA = %f",hra);
    printf("\nThe amount of MA = %f",ma);
    printf("\nThe amount of TA = %f",ta);
    printf("\nThe total amount of allowances = %f\n",allowance);

    it=500;
    pf=(12*bs)/100;
    deduction = pf+it;
    printf("\nThe amount of PF = %f",pf);
    printf("\nThe amount of IT = %f\n",it);
    printf("\nThe total deduction = %f",deduction);
    net=bs+allowance-deduction;
    printf("\nThe total net salary = %f\n",net);
}
```

Output :



Practical - 2

Aim :

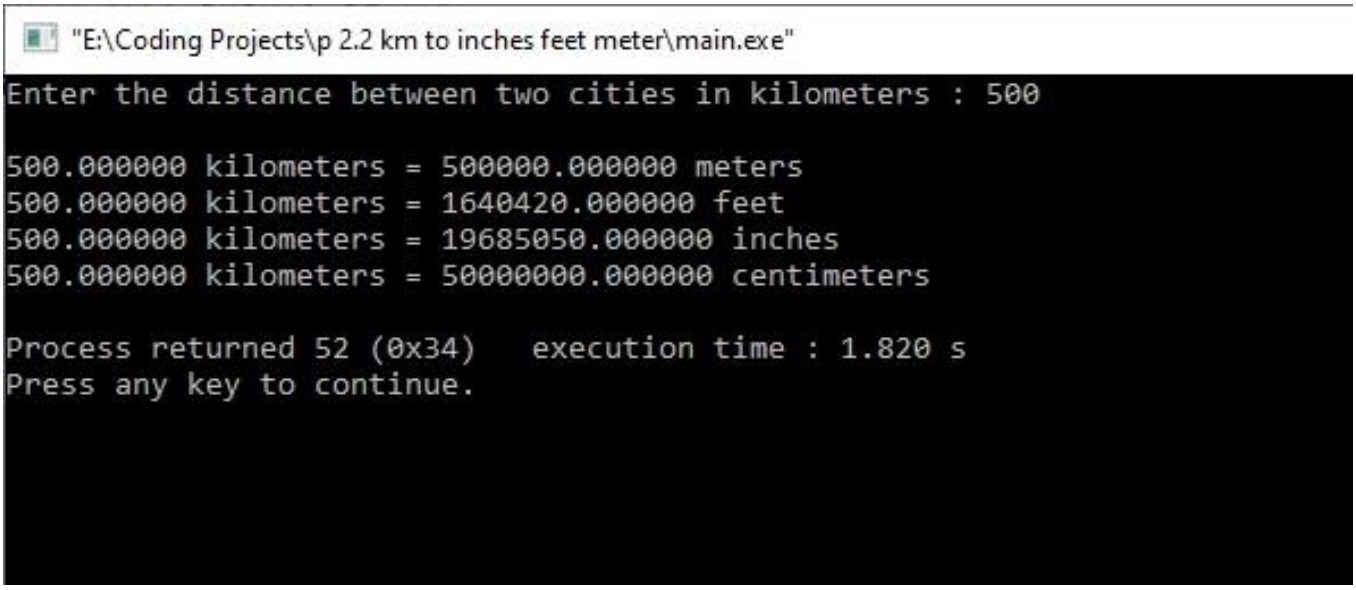
The distance between two cities (in km) is input through the keyboard. Write a program to convert and print its distance in meters, feet, inches and centimeters.

Source Code :

```
#include <stdio.h>

#include <stdlib.h>

void main()

{
```

```
    float x;
    printf("Enter the distance between two cities in kilometers : ");
    scanf("%f",&x);
    printf("\n%f kilometers = %f meters\n",x,x*1000);
    printf("%f kilometers = %f feet\n",x,x*3280.84);
    printf("%f kilometers = %f inches\n",x,x*39370.1);
    printf("%f kilometers = %f centimeters\n",x,x*100000);
}
```

Output :



Set - 3

Practical - 1

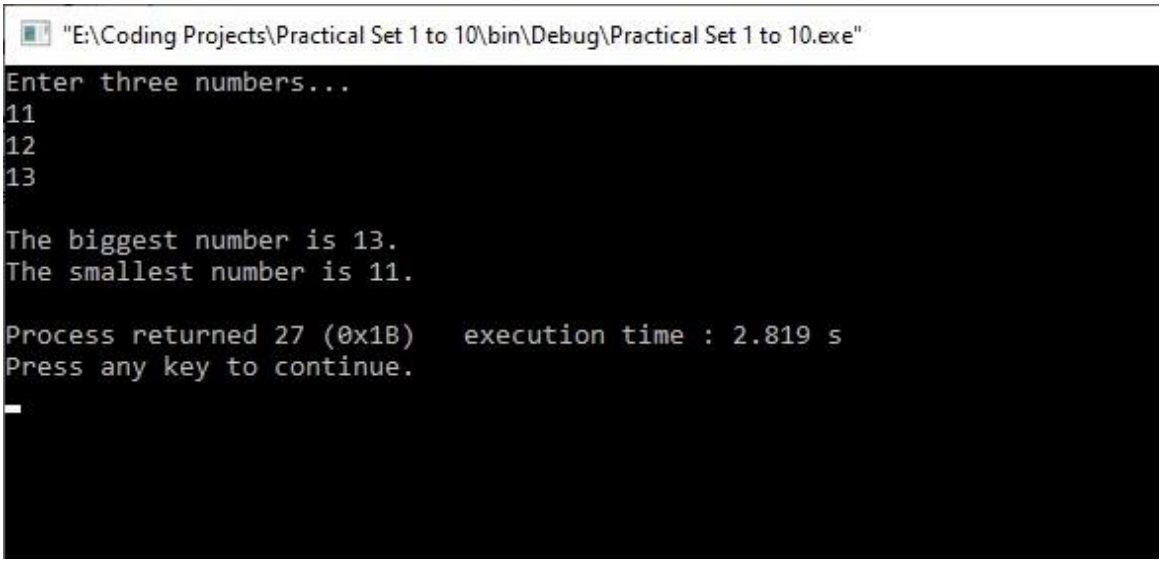
Aim :

Write a program to find the greatest of the three numbers entered through the keyboard using conditional operators.

Source Code :

```
#include <stdio.h>
#include <stdlib.h>
void main()
{
   int x,y,z,max,min;
   printf("Enter three numbers...\n");
   scanf("%d %d %d",&x,&y,&z);
   if(x == y && y == z)
   {
      printf("\nEntered numbers are same.\n");
   }
   else
   {
      max = x>y ? (x>z ? x : z) : (y>z ? y : z);
      min = x<y ? (x<z ? x : z) : (y<z ? y : z);
      printf("\nThe biggest number is %d.\n",max);
      printf("The smallest number is %d.\n",min);
   }
}
```

Output :



Practical - 2

Aim :

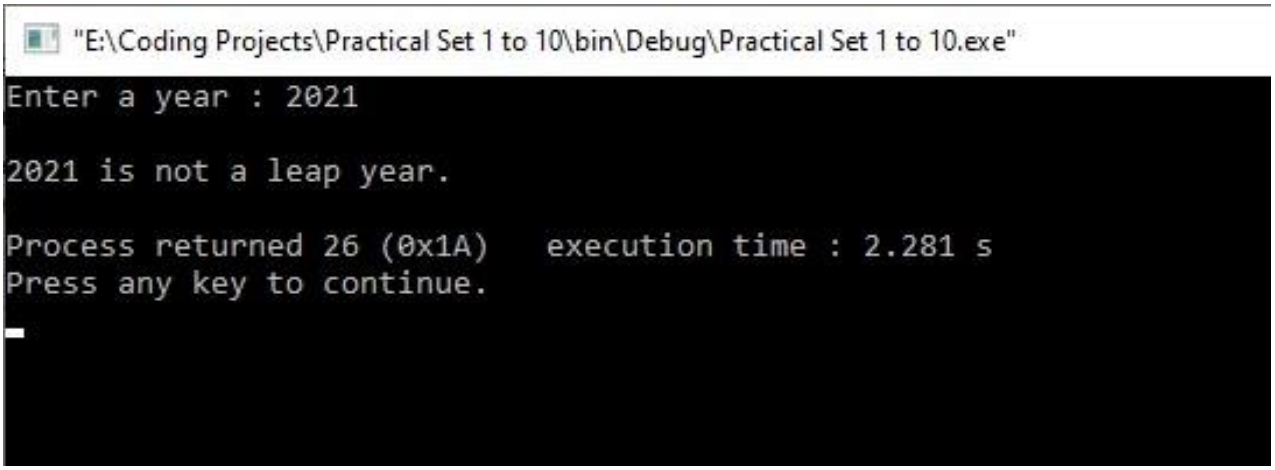
Any year is input through the keyboard. Write a program to determine whether the year is a leap year or not. Use the logical operators && and ||.

Source Code :

```
#include <stdio.h>

#include <stdlib.h>

void main()

{

   int year;

   printf("Enter a year : ");

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

```
    if ( (year%400==0) || (year%100!=0 && year%4==0) )
        {
          printf("\n%d is a leap year.\n",year);
        }
    else
        {
          printf ("\n%d is not a leap year.\n",year);
        }
}
```

Output :



Set - 4

Practical - 1

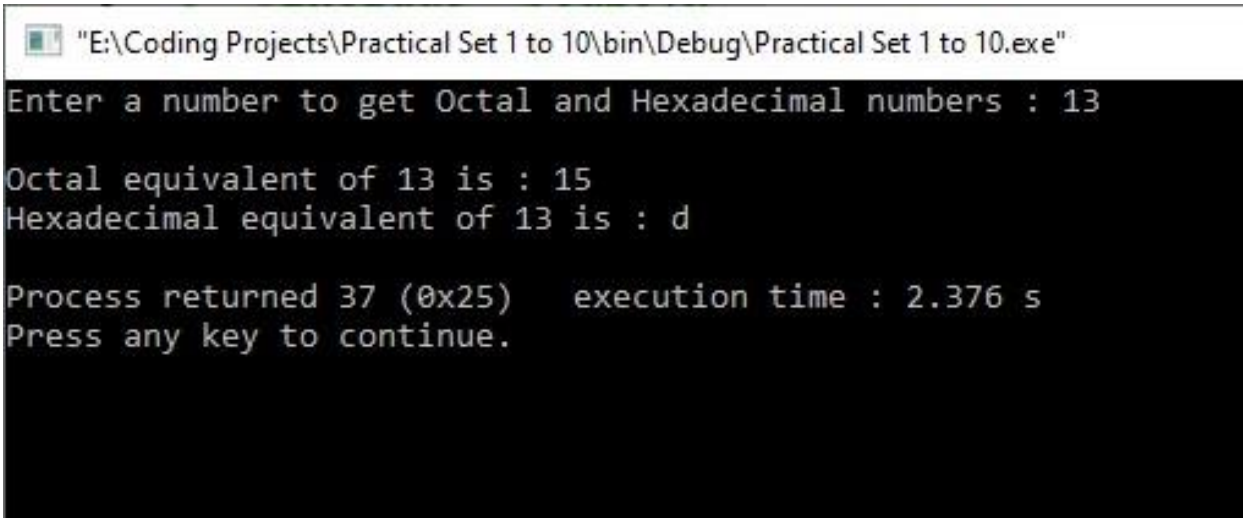
Aim :

Write a program to convert the decimal number into octal and hexadecimal format. Hint: %o and %x

Source Code :

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

void main()
{
   int x;
   printf("Enter a number to get Octal and Hexadecimal numbers : ");
   scanf("%d",&x);
   printf("\nOctal equivalent of %d is : %o",x,x);
   printf("\nHexadecimal equivalent of %d is : %x\n",x,x);
}
```

Ouput :



Practical - 2

Aim :

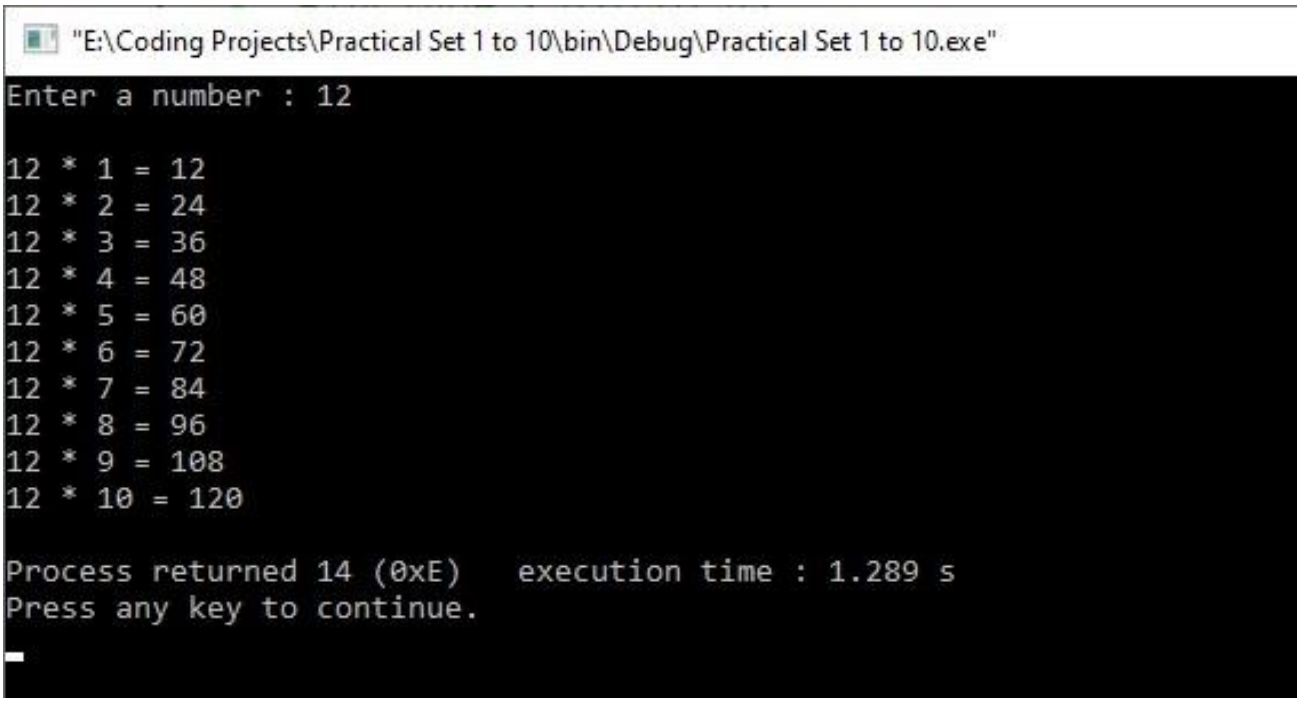
Write a C Program to print multiplication table of number entered by user.

Source Code :

```
#include <stdio.h>
#include <stdlib.h>
void main()
{
   int x;
   printf("Enter a number : ");
   scanf("%d",&x);
   printf("\n%d * %d = %d\n",x,1,x*1);
```

```
    printf("%d * %d = %d\n",x,2,x*2);
    printf("%d * %d = %d\n",x,3,x*3);
    printf("%d * %d = %d\n",x,4,x*4);
    printf("%d * %d = %d\n",x,5,x*5);
    printf("%d * %d = %d\n",x,6,x*6);
    printf("%d * %d = %d\n",x,7,x*7);
    printf("%d * %d = %d\n",x,8,x*8);
    printf("%d * %d = %d\n",x,9,x*9);
    printf("%d * %d = %d\n",x,10,x*10);
}
```

Output :



Set - 5

Practical - 1

Aim :

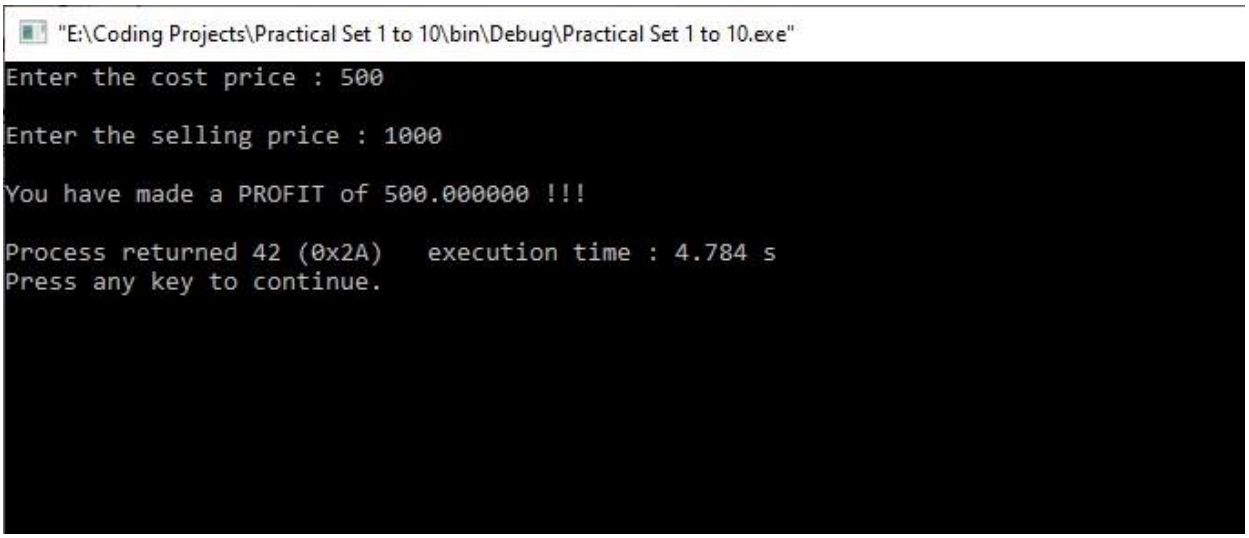
If the cost price and selling price of an item is input through the keyboard, write a program to determine whether the seller has made profit or incurred loss. Also determine how much profit he made or loss he incurred.

Source Code :

```
#include <stdio.h>
#include <stdlib.h>
//x=cost price and y=selling price
void main()
{
   float x,y;
   printf("Enter the cost price : ");
   scanf("%f",&x);
   printf("\nEnter the selling price : ");
   scanf("%f",&y);

   if (x<y)
   {
      printf("\nYou have made a PROFIT of %f !!!\n",y-x);
   }
   else
   {
      printf("\nYou have incurred a LOSS of %f.\n",x-y);
```

```
    }
}
```

Output :



Practical - 2

Aim :

If the ages of Ram, Shyam and Ajay are input through the keyboard, write a program to determine the youngest of the three. (Hint: Use Nested Switch Statement)

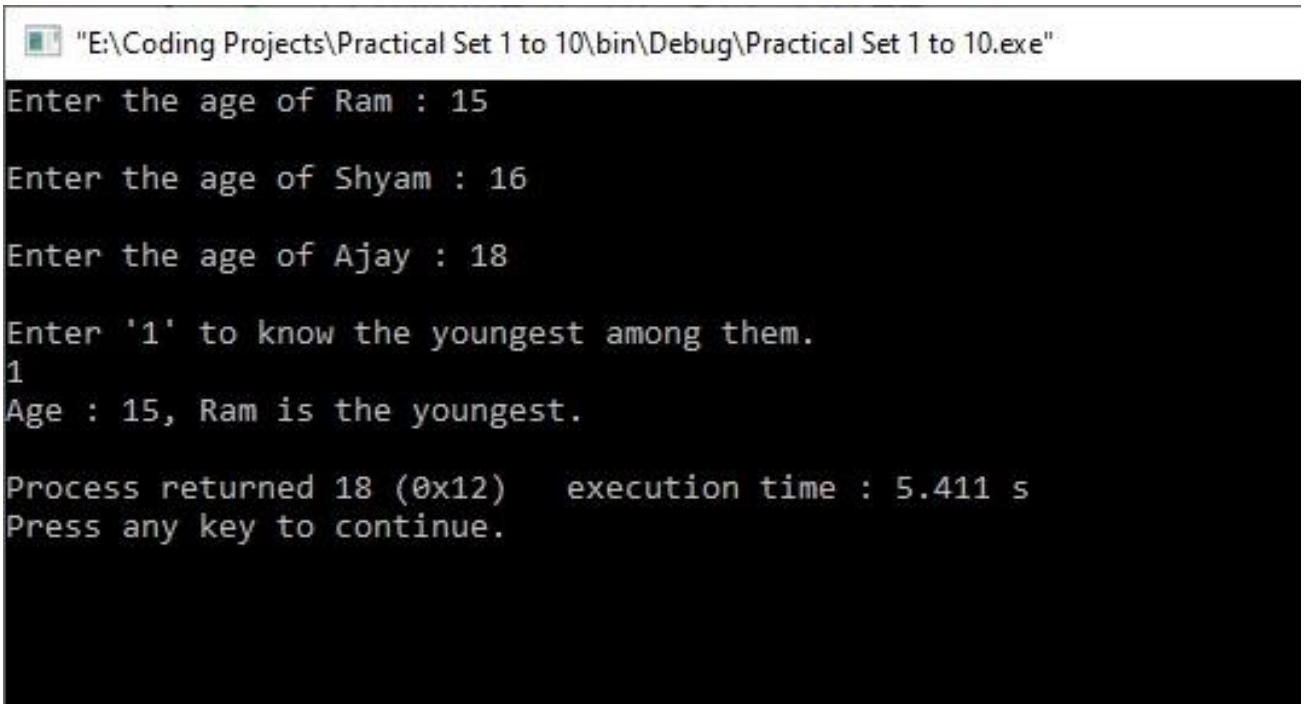
Source Code :

```
#include <stdio.h>
#include <stdlib.h>
void main()
{
```

```
int n,Ram,Shyam,Ajay;
printf("Enter the age of Ram : ",Ram);
scanf("%d",&Ram);
printf("\nEnter the age of Shyam : ",Shyam);
scanf("%d",&Shyam);
printf("\nEnter the age of Ajay : ",Ajay);
scanf("%d",&Ajay);

printf("\nEnter '1' to know the youngest among them.\n");
scanf("%d",&n);
switch(1)
{
case 1:
   if ((Shyam>=Ram)&&(Ajay>=Ram))
       printf("Age : %d, Ram is the youngest.\n",Ram);

      switch(2)
       {
          case 2:
              if ((Ram>=Shyam)&&(Ajay>=Shyam))
              printf("Age : %d, Shyam is the youngest.\n",Shyam);
       }
             switch(3)
               {
                  case 3:
                      if ((Ram>=Ajay)&&(Shyam>=Ajay))
                      printf("Age : %d, Ajay is the youngest.\n",Ajay);
                }
```

```
    }
}
```

Output :



Practical - 3

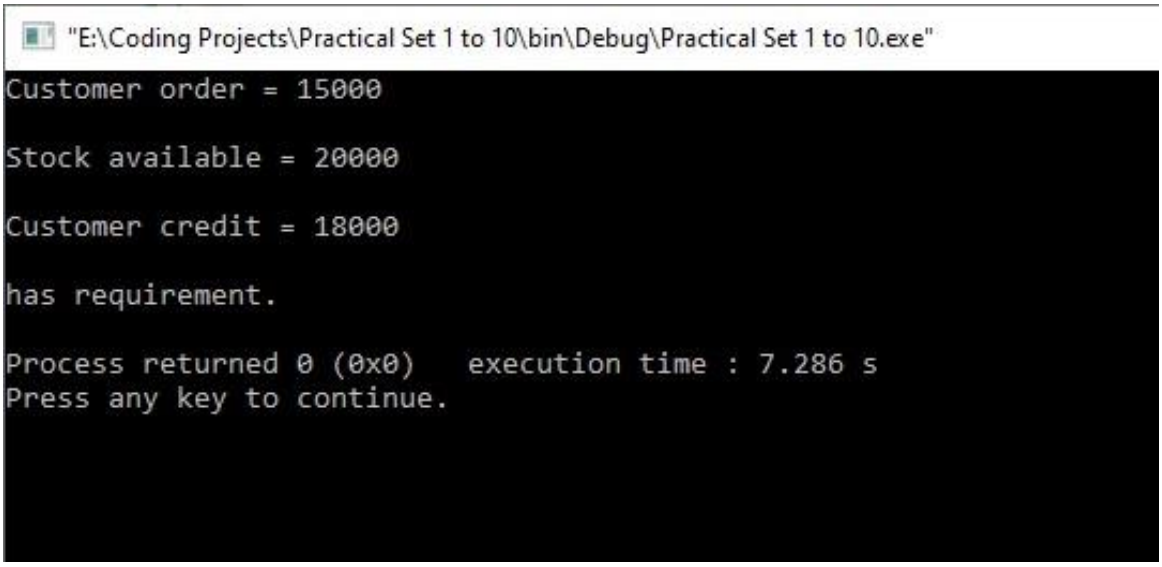
Aim :

The policy followed by a company to process customer orders is given by the following rules: a) If a customer order is less than or equal to that in stock and ‘has credit’ is OK, supply ‘has requirements’. b) If ‘has credit’ is not OK do not supply. Send him intimation. c) If ‘has credit’ is OK but the item in stock is less than ‘has ordered’, supply what is in stock and Intimate him that the balance will be refunded. Write a C program to implement the company policy.

Source Code :

```
#include<stdio.h>
#include<stdlib.h>
void main()
{
   int CO,HS,HC,CC,CD=14000;
   //CO=customer order,  HS=stock of company,  HC=customer credit
   //CC=customer credit,  CD=company credit.
   printf("Customer order = ");
   scanf("%d",&CO);
   printf("\nStock available = ");
   scanf("%d",&HS);
   printf("\nCustomer credit = ");
   scanf("%d",&CC);
   if( CO<=HS && CC >= CD )
   {
      printf("\nhas requirement.\n");
   }
   else if( CO>HS && CC >= CD )
      {
         printf("balance will be refunded !\n");
      }
   else if(CC<CD)
   {
      printf("credit not sufficient !\n");
   }
}
```

Output :


"E:\Coding Projects\Practical Set 1 to 10\bin\Debug\Practical Set 1 to 10.exe"
Customer order = 15000
Stock available = 20000
Customer credit = 18000
has requirement.
Process returned 0 (0x0) execution time : 7.286 s
Press any key to continue.

Set - 6

Practical - 1

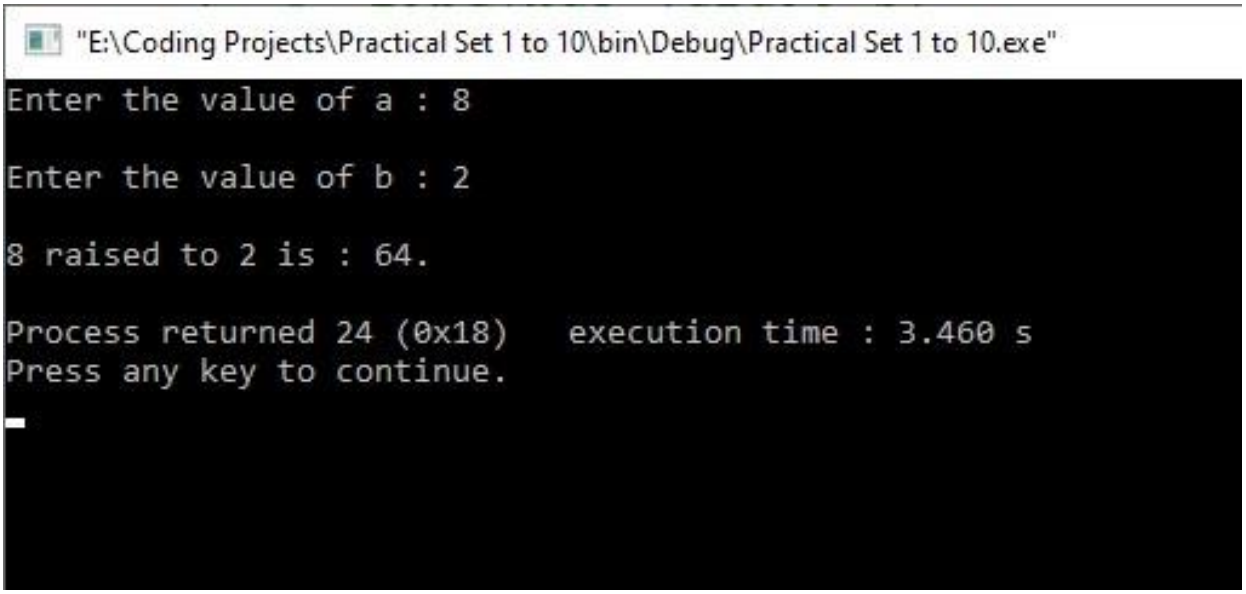
Aim :

Two numbers are entered through the keyboard. Write a program to find the value of one number raised to the power of another. (Use While loop)

Source Code :

```
#include <stdio.h>
#include <stdlib.h>
void main()
{
   int a,b,c=1,x;
   printf("Enter the value of a : ");
   scanf("%d",&a);                    //   3
   printf("\nEnter the value of b : ");
   scanf("%d",&b);                    //   4
   x=b;
   while(b!=0)
   {
      c=c*a;   // c=3  9  27  81
      b--;     // b=3  2   1   0
   }
   printf("\n%d raised to %d is : %d.\n",a,x,c);
}
```

Output :



Practical - 2

Aim :

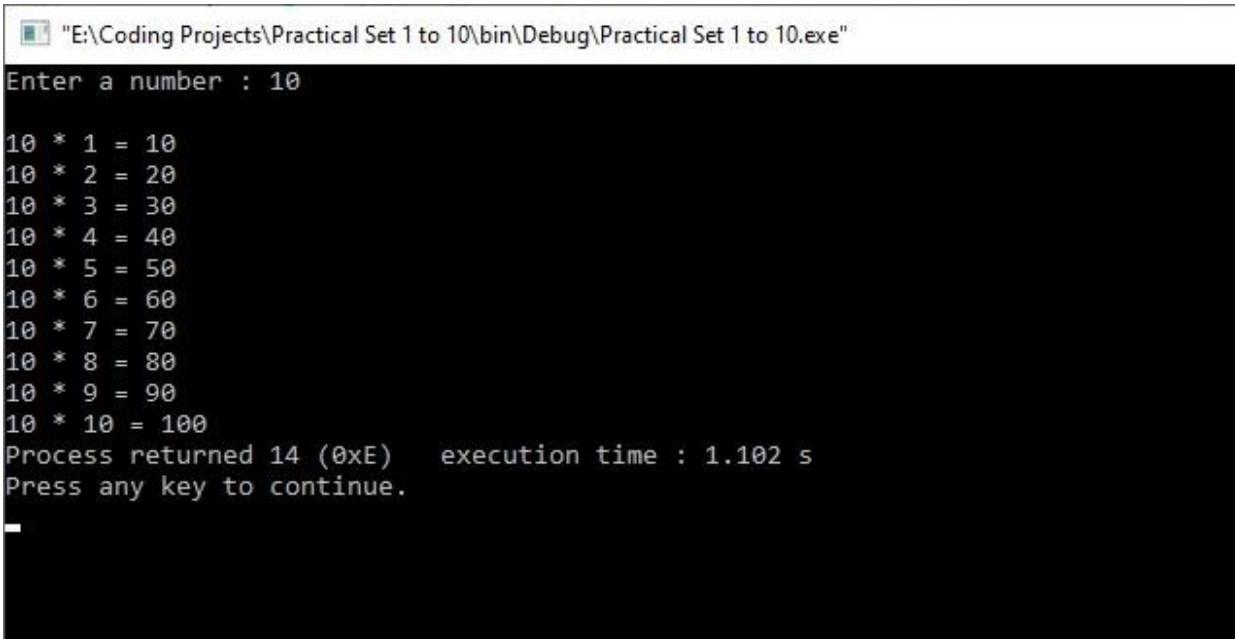
Write a program to print the multiplication table of the number entered from the keyboard. The table should get displayed in the following form: 12*1=12 12*2=24 …

Source Code :

```
#include <stdio.h>
#include <stdlib.h>
void main()
{
   int n,x;
   printf("Enter a number : ");
```

```
    scanf("%d",&n);
    for(x=1;x<=10;x++)
    {
        printf("\n%d * %d = %d",n,x,n*x);
    }
}
```

Output :



Practical - 3

Aim :

Write a menu driven program which has following options: 1. Prime or not 2. Perfect number or not 3. Factorial of a number 4. Exit Use do...while statement so that the menu is displayed at least once. Also use Switch statement.

Source Code :

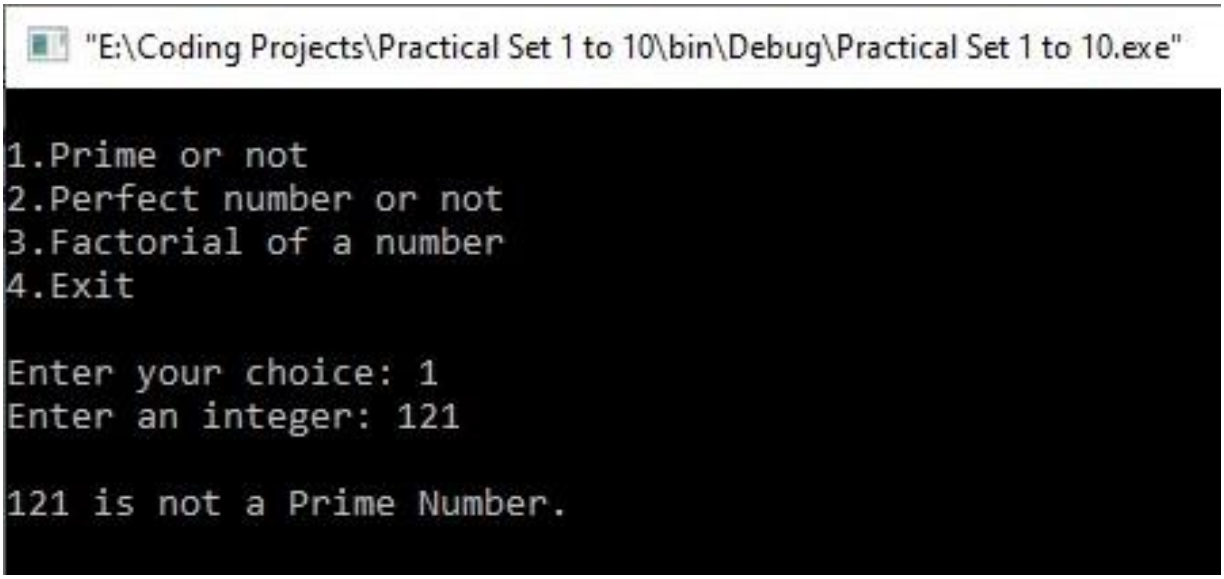
```
#include<stdio.h>
int main()
{
 int c=0, num, n, flag=0, i , rem , sum=0;
 unsigned long long fact ;
 while(c!=4)
 {
 //display menu
 printf("\n1.Prime or not \n2.Perfect number or not \n3.Factorial of a number\n4.Exit\n");
 //display choice option to the user

 printf("\nEnter your choice: ");
 scanf("%d", &c);
 //write case statement for Four options
 switch(c)
 {
 // Prime or Not
 case 1:

 printf("Enter an integer: ");
 scanf("%d", &num);
 n=num;
 for(i=2;i<=n/2;i++)
 {
 if(num%i==0)
```

```
{
flag=1;
break;
}
}
//for number "1" it's neither prime nor composite
if(num==1)
printf("\n1 is neither prime nor composite");

else
{
if(flag==0)
printf("\n%d is Prime Number.\n\n", n);
else
printf("\n%d is not a Prime Number.\n\n", n);
}
break;

// Perfect Number or Not

case 2:

printf("Enter a Number\n");
scanf("%d", &num);
for (i = 1; i < num; i++){
rem = num % i;
if (rem == 0)
```

```
{
sum = sum + i;
}
}
if (sum == num){
printf("Entered Number is perfect number\n");
}
else{
printf("Entered Number is not a perfect number\n");

}
break;

// Factorial of a number

case 3:

fact = 1;
printf("\nEnter an integer: ");
scanf("%d", &num);

if (num < 0)
{
printf("Error! Factorial of a negative number doesn't exist.");
}
else
{
```

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

fact = fact * i;
}
printf("\nFactorial of %d = %lld\n", num, fact);

}
break ;
//For Exit block
case 4:
printf("\nExit");
break;
}
}
}
```

Output :



Practical - 4

Aim :

Write a program for a match-stick game between the computer and a user. Your Program should ensure that the computer always wins. Rules for the games are as follows:

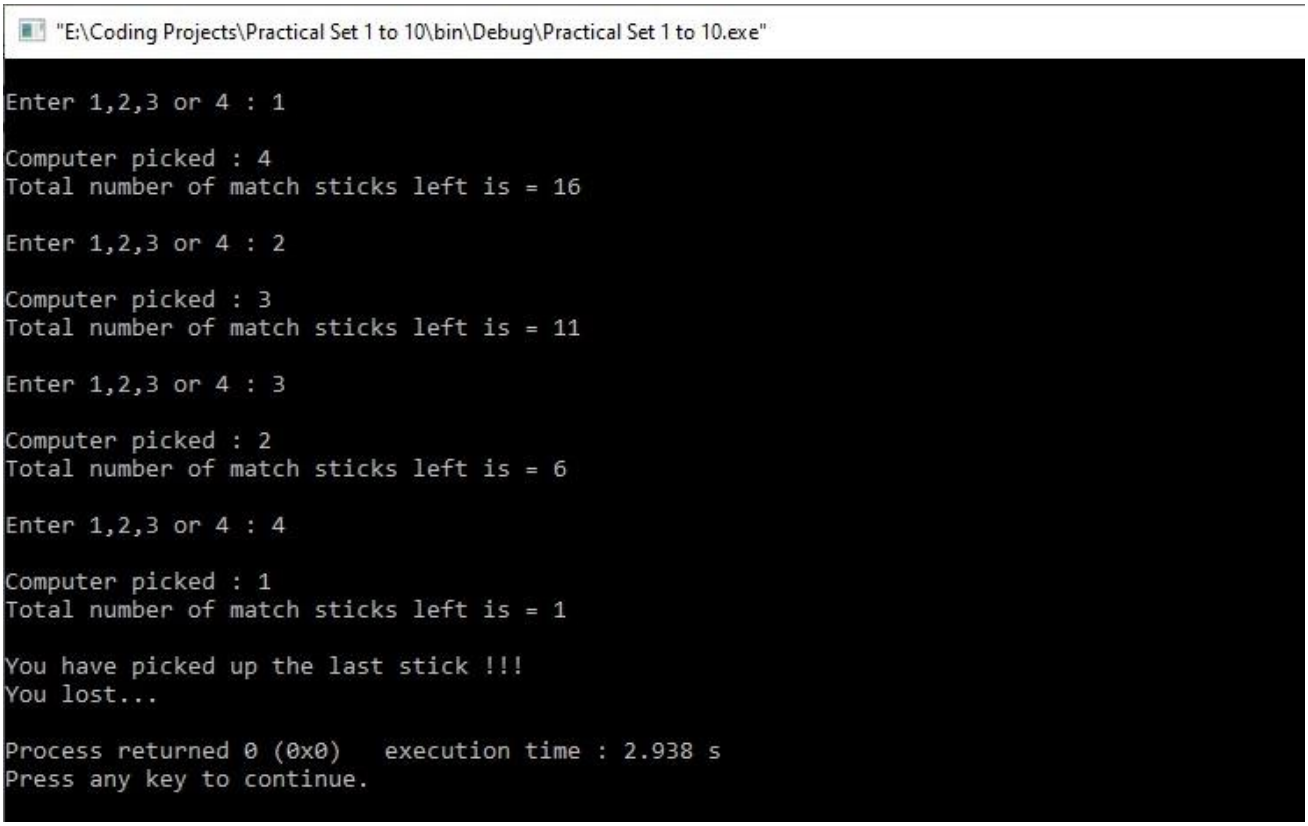
- There are 21 match-sticks.
- The computer asks the player to pick 1, 2, 3, or 4 match-sticks.
- After the person picks, the computer does its picking.
- Whoever is forced to pick up the last match-stick loses the game. Use while loop, break and Continue Statements.

Source Code :

```
#include<stdio.h>
#include <stdlib.h>
void main()
{
   int m=21,p,c;
   while(1)
   {
   printf("\nEnter 1,2,3 or 4 : ");
   scanf("%d",&p);
   if(p <1 || p>4 )
      {
         continue;
      }
   c=5-p;
   printf("\nComputer picked : %d\n",c);
   m=m-5;
```

```
    printf("Total number of match sticks left is = %d\n",m);
    if(m==1)
      {
        printf("\nYou have picked up the last stick !!!\nYou lost...\n");
        break;
      }
    }
}
```

Output :



Set - 7

Practical - 1

Aim :

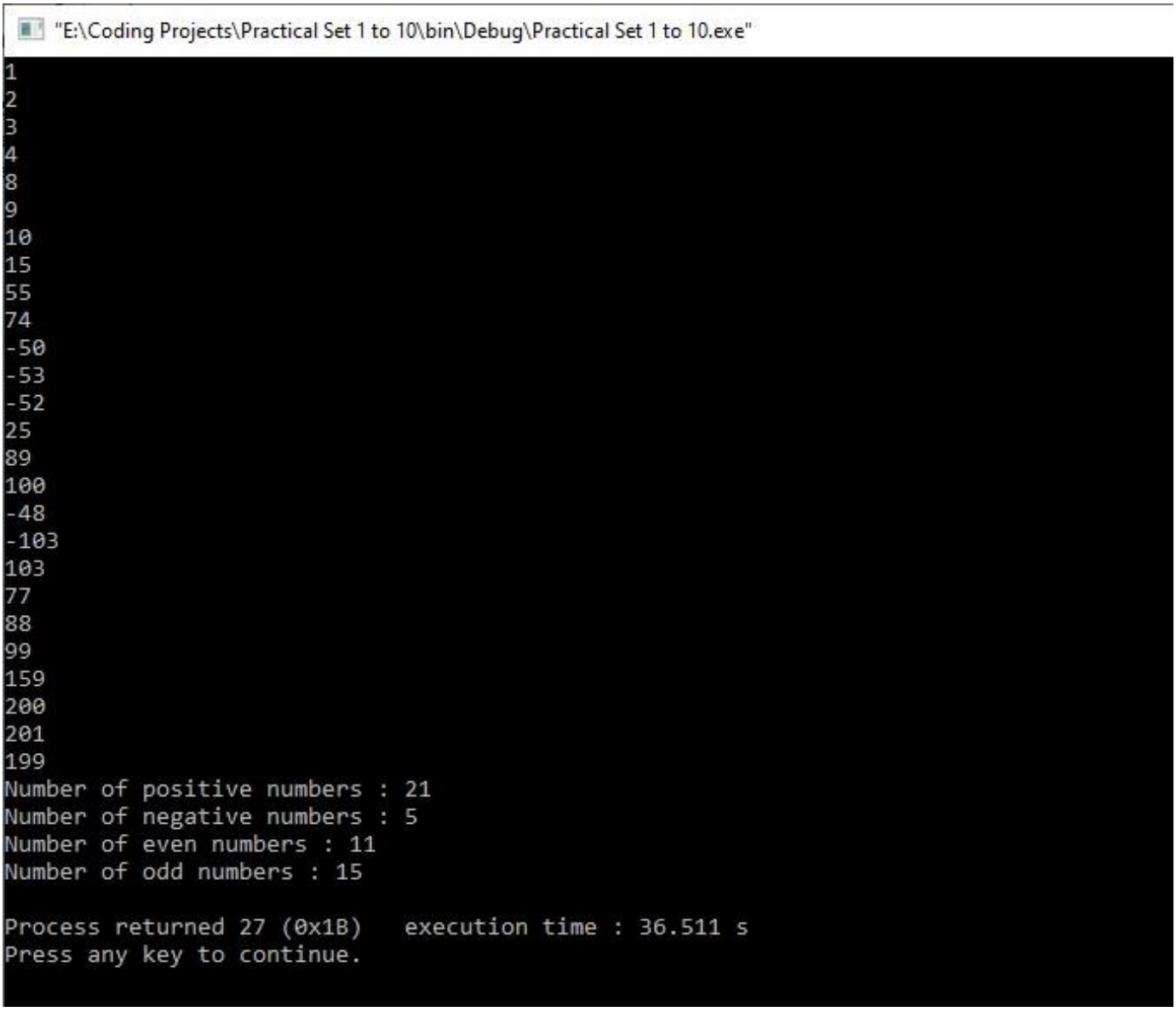
Twenty five numbers are entered from the keyboard into an array. Write a program to find out how many of them are positive, negative, how many are even and odd.

Source Code :

```
#include <stdio.h>
#include <stdlib.h>
void main()
{
   int a[25],i,p=0,n=0,even=0,odd=0;
   for (i=0;i<=25;i++)
   {
      scanf("%d",&a[i]);
   }
   for (i=0;i<=25;i++)
   {
      if (a[i]>0)
         p++;
      else
         n++;
      if (a[i]%2 == 0)
         even++;
      else
         odd++;
```

```
    }
    printf("Number of positive numbers : %d\n",p);
    printf("Number of negative numbers : %d\n",n);
    printf("Number of even numbers : %d\n",even);
    printf("Number of odd numbers : %d\n",odd);
}
```

Output :



Practical - 2

Aim :

Write a program for creating two arrays of different size and merge both arrays into one by sorting those arrays in ascending order. [Merge by sorting]

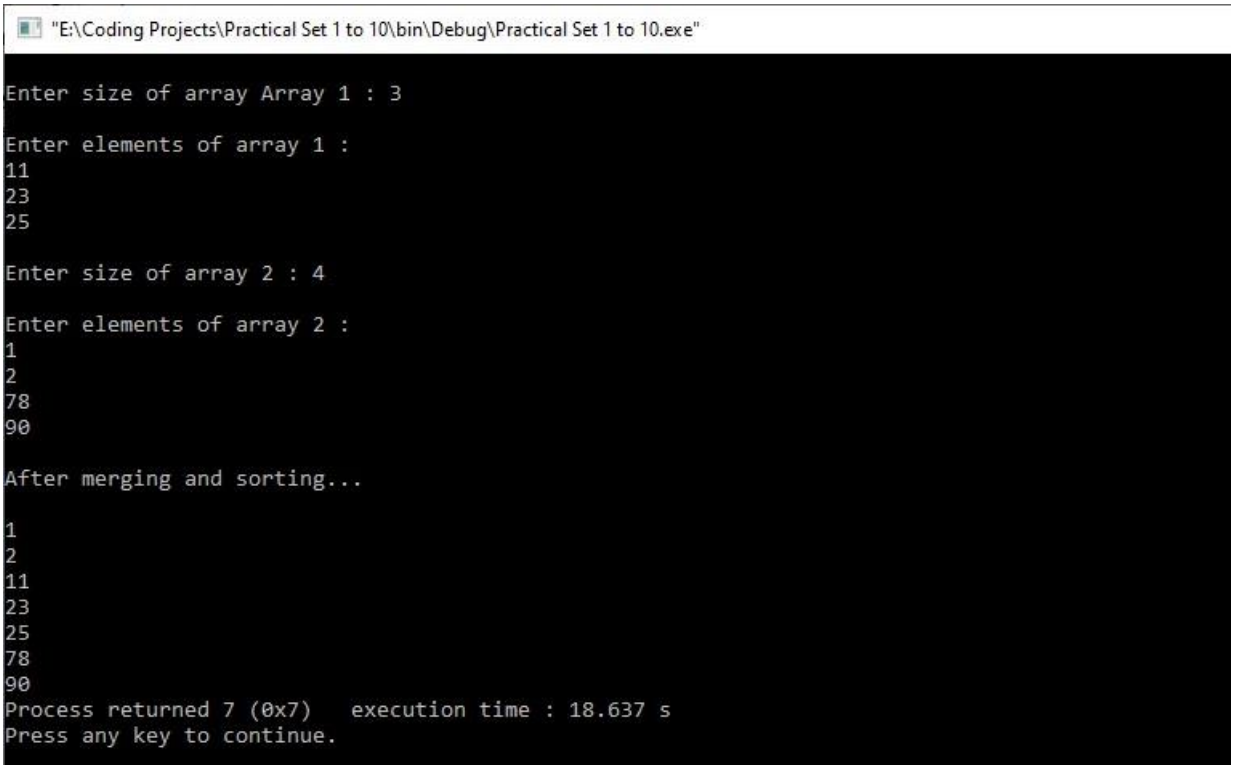
Source Code :

```
#include <stdio.h>
void main()
{

   int array1[50], array2[50], array3[100], m, n, i, j, k = 0;
   printf("\nEnter size of array Array 1 : ");
   scanf("%d", &m);

   printf("\nEnter elements of array 1 :\n");
   for (i = 0; i < m; i++)
   {
      scanf("%d", &array1[i]);
   }

   printf("\nEnter size of array 2 : ");
   scanf("%d", &n);
   printf("\nEnter elements of array 2 :\n");
   for (i = 0; i < n; i++)
   {
      scanf("%d", &array2[i]);
   }
```

```
i = 0;
j = 0;
while (i < m && j < n)
{
   if (array1[i] < array2[j])
   {
      array3[k] = array1[i];
      i++;
   }
   else
   {
      array3[k] = array2[j];
      j++;
   }
   k++;
}
if (i >= m)
{
   while (j < n)
   {
      array3[k] = array2[j];
      j++;
      k++;
   }
}
if (j >= n)
{
   while (i < m)
```

```
        {
            array3[k] = array1[i];
            i++;
            k++;
        }
    }
    printf("\nAfter merging and sorting... \n");
    for (i = 0; i < m + n; i++)
    {
        printf("\n%d", array3[i]);
    }
}
```

Output :



Practical - 3

Aim :

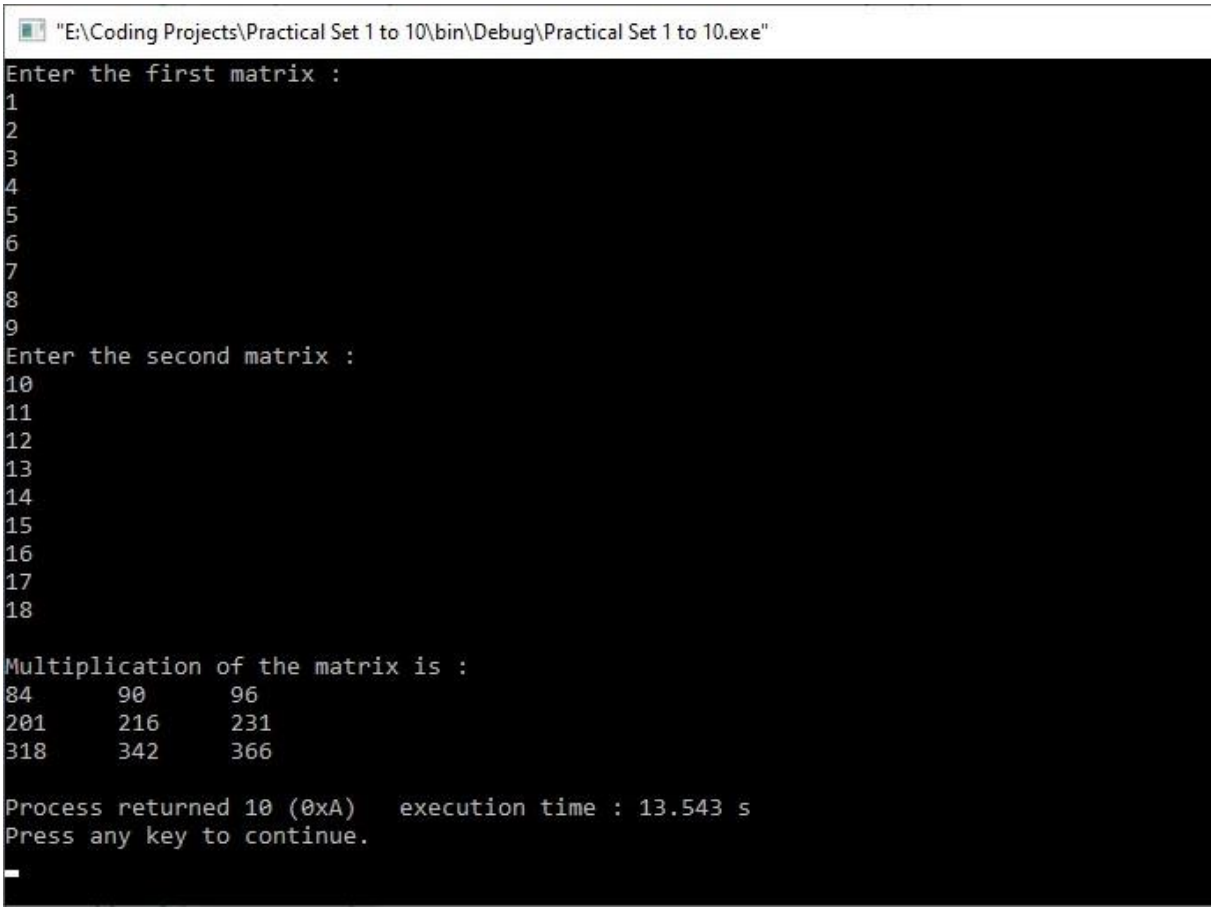
Write a Program to multiply any two 3*3 Matrices.

Source Code :

```
#include<stdio.h>
#include <stdlib.h>
void main()
{
   int i,j,a[3][3],b[3][3],mul[3][3],k;
   printf("Enter the first matrix :\n");
   for(i=0;i<3;i++)
      {
         for(j=0;j<3;j++)
         {
            scanf("%d",&a[i][j]);
         }
      }
   printf("Enter the second matrix :\n");
   for(i=0;i<3;i++)
      {
         for(j=0;j<3;j++)
         {
            scanf("%d",&b[i][j]);
         }
      }
   for(i=0;i<3;i++)
```

```
    {
      for(j=0;j<3;j++)
      {
        mul[i][j]=0;
        for(k=0;k<3;k++)
          {
            mul[i][j]+=a[i][k]*b[k][j];
          }
      }
    }
  printf("\nMultiplication of the matrix is :\n");
  for(i=0;i<3;i++)
    {
      for(j=0;j<3;j++)
      {
        printf("%d\t",mul[i][j]);
      }
      printf("\n");
    }
}
```

Output :


"E:\Coding Projects\Practical Set 1 to 10\bin\Debug\Practical Set 1 to 10.exe"
Enter the first matrix :
1
2
3
4
5
6
7
8
9
Enter the second matrix :
10
11
12
13
14
15
16
17
18
Multiplication of the matrix is :
84 90 96
201 216 231
318 342 366
Process returned 10 (0xA) execution time : 13.543 s
Press any key to continue.

Set - 8

Practical - 1

Aim :

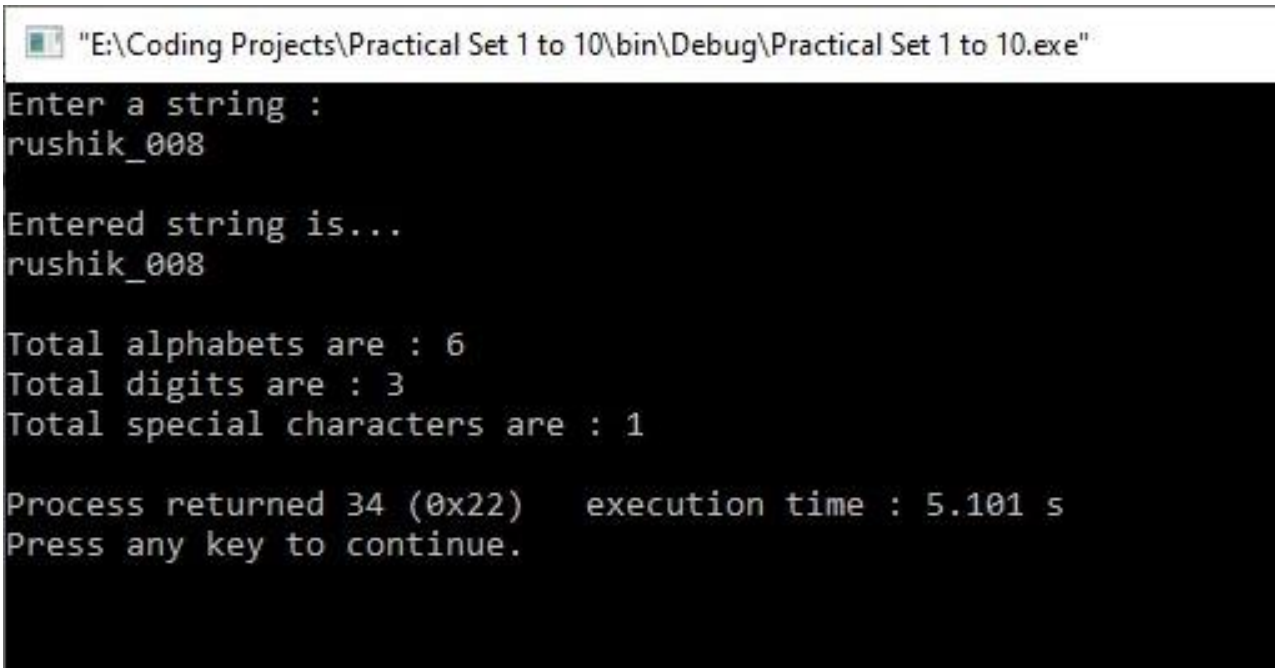
Take a user input for a string and calculate the number of alphabets, digits and special characters from the given input.

Source Code :

```
#include<stdio.h>
#include <stdlib.h>
void main()
{
   char str[100],al=0,di=0,sp=0,i;
   puts("Enter a string :");
   gets(str);
   puts("\nEntered string is...");
   puts(str);
   for(i=0;i<=str[i]!='\0';i++)
     {
       if(str[i]>='A' && str[i]<= 'Z' || str[i]>='a' && str[i]<='z')
          {
             al++;
          }
       else if(str[i]>='0' && str[i]<='9' )
          {
             di++;
          }
```

```
        else
          {
          sp++;
          }
      }
   printf("\nTotal alphabets are : %d",al);
   printf("\nTotal digits are : %d",di);
   printf("\nTotal special characters are : %d\n",sp);
}
```

Output :



Practical - 2

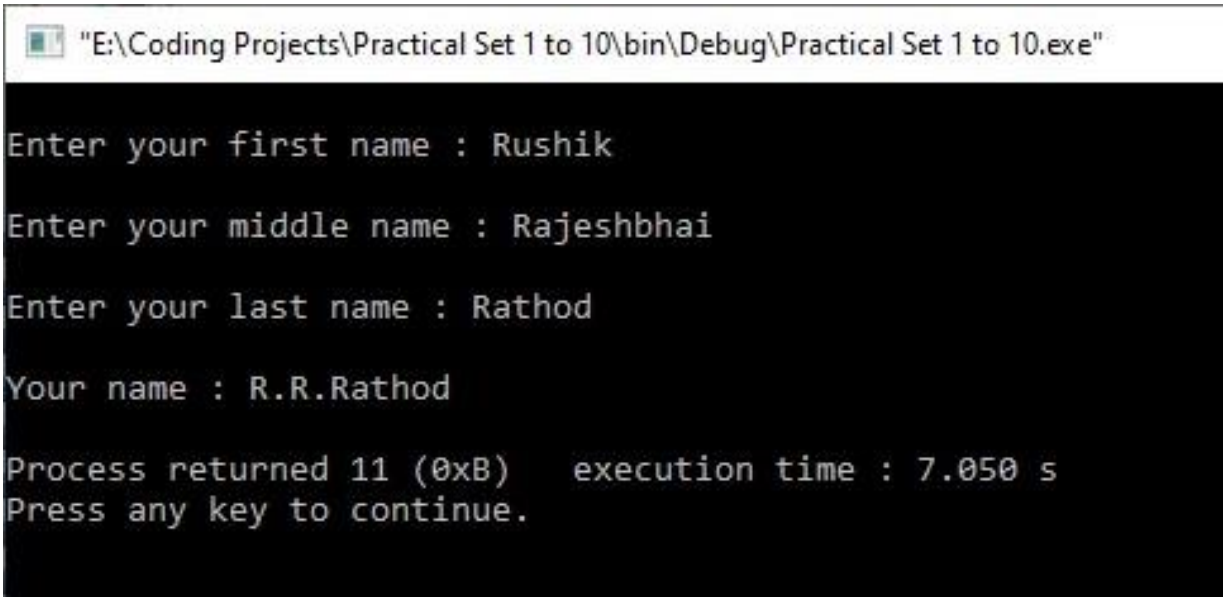
Aim :

Write a program that takes a set of names of individuals and abbreviates the first, middle and other names except the last name by their first letter.

Source Code :

```
#include<stdio.h>
#include<string.h>
void main()
{
 int i,h;
 char a[15],b[15],c[15];
 printf("\nEnter your first name : ");
 scanf("%s",&a);
 printf("\nEnter your middle name : ");
 scanf("%s",&b);
 printf("\nEnter your last name : ");
 scanf("%s",&c);
 printf("\nYour name : ");
 printf("%c.%c.%s\n",a[0],b[0],c);
}
```

Output :



Practical - 3

Aim :

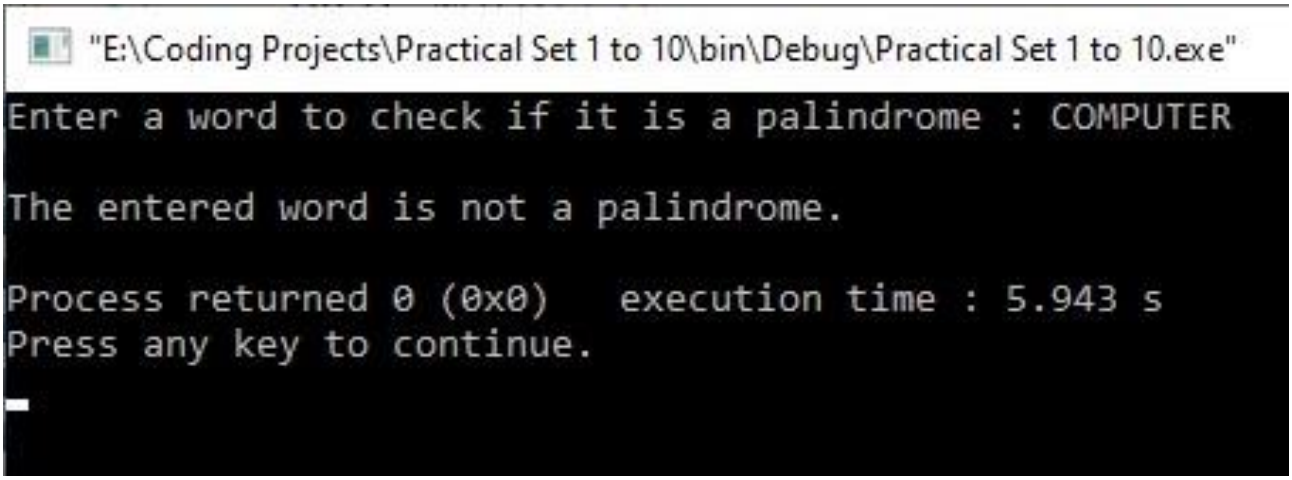
Write a C program to check if the user inputed string is palindrome or not using recursion.

Source Code :

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

void check(char [], int);
void main()
```

```
{
    char word[15];
    printf("Enter a word to check if it is a palindrome\n");
    scanf("%s", word);
    check(word, 0);
 }

void check(char word[], int index)
{
    int len = strlen(word) - (index + 1);
    if (word[index] == word[len])
    {
        if (index + 1 == len || index == len)
        {
            printf("The entered word is a palindrome\n");
            return;
        }
        check(word, index + 1);
    }
    else
    {
        printf("The entered word is not a palindrome\n");
    }
}
```

Output :


"E:\Coding Projects\Practical Set 1 to 10\bin\Debug\Practical Set 1 to 10.exe"
Enter a word to check if it is a palindrome : COMPUTER
The entered word is not a palindrome.
Process returned 0 (0x0)   execution time : 5.943 s
Press any key to continue.

Set - 9

Practical - 1

Aim :

Write a C program to check if the entered number is prime or not by using types of user defined functions

(i) No arguments passed and no return value
(ii) No arguments passed but a return value
(iii) Argument passed but no return value
(iv) Argument passed and a return value

Source Code :

No arguments passed and no return value

```
#include <stdio.h>
void checkPrimeNumber();
int main()
{
 checkPrimeNumber(); // argument is not passed
 return 0;
}
// return type is void meaning doesn't return any value
void checkPrimeNumber()
{
 int n, i, flag = 0;
 printf("Enter a positive integer: ");
 scanf("%d",&n);
```

```
 for(i=2; i <= n/2; ++i)
 {
 if(n%i == 0)
 {
 flag = 1;
 }
 }
 if (flag == 1)
 printf("%d is not a prime number.", n);
 else
 printf("%d is a prime number.", n);
}
```

No arguments passed but a return value

```
#include <stdio.h>
int getInteger();
int main()
{
 int n, i, flag = 0;
 // no argument is passed
 n = getInteger();
 for(i=2; i<=n/2; ++i)
 {
 if(n%i==0){
 flag = 1;
 break;
 }
```

```
 }
 if (flag == 1)
 printf("%d is not a prime number.", n);
 else
 printf("%d is a prime number.", n);
 return 0;
}
```

Argument passed but no return value

```
#include <stdio.h>
void checkPrimeAndDisplay(int n);
int main()
{
 int n;
printf("Enter a positive integer: ");
 scanf("%d",&n);
 // n is passed to the function
 checkPrimeAndDisplay(n);
 return 0;
}
// return type is void meaning doesn't return any value
void checkPrimeAndDisplay(int n)
{
 int i, flag = 0;
 for(i=2; i <= n/2; ++i)
 {
 if(n%i == 0){
```

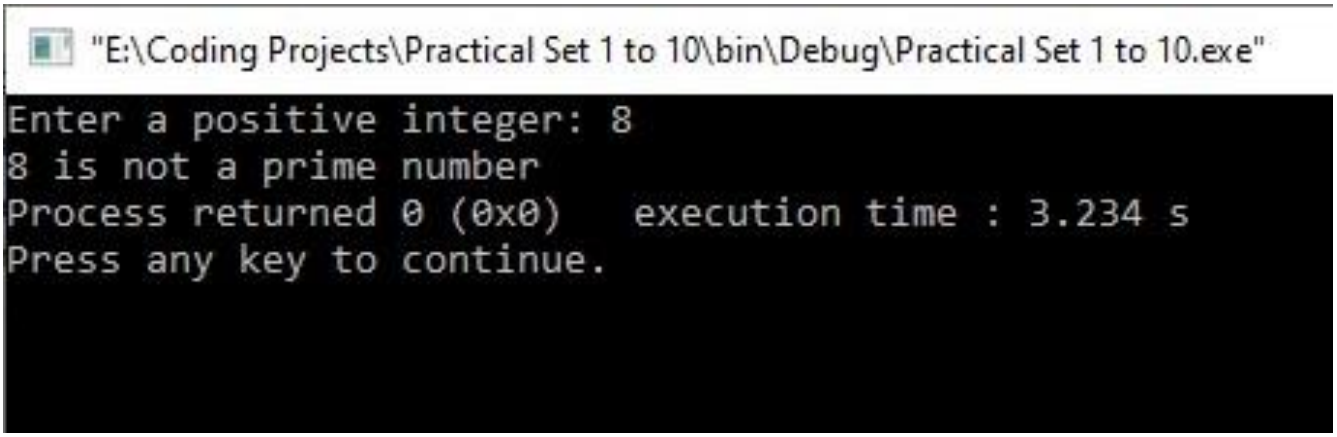
```
 flag = 1;
 break;
 }
 }
 if(flag == 1)
 printf("%d is not a prime number.",n);
 else
 printf("%d is a prime number.", n);
}
```

Argument passed and a return value

```
#include <stdio.h>
int checkPrimeNumber(int n);
int main()
{
 int n, flag;
 printf("Enter a positive integer: ");
 scanf("%d",&n);
 // n is passed to the checkPrimeNumber() function
 // the returned value is assigned to the flag variable
 flag = checkPrimeNumber(n);
 if(flag == 1)
 printf("%d is not a prime number",n);
 else
 printf("%d is a prime number",n);
 return 0;
}
```

```
// int is returned from the function

int checkPrimeNumber(int n)
{
 int i;
 for(i=2; i <= n/2; ++i)
 {
 if(n%i == 0)
 return 1;
 }
 return 0;
}
```

Output :



Practical - 2

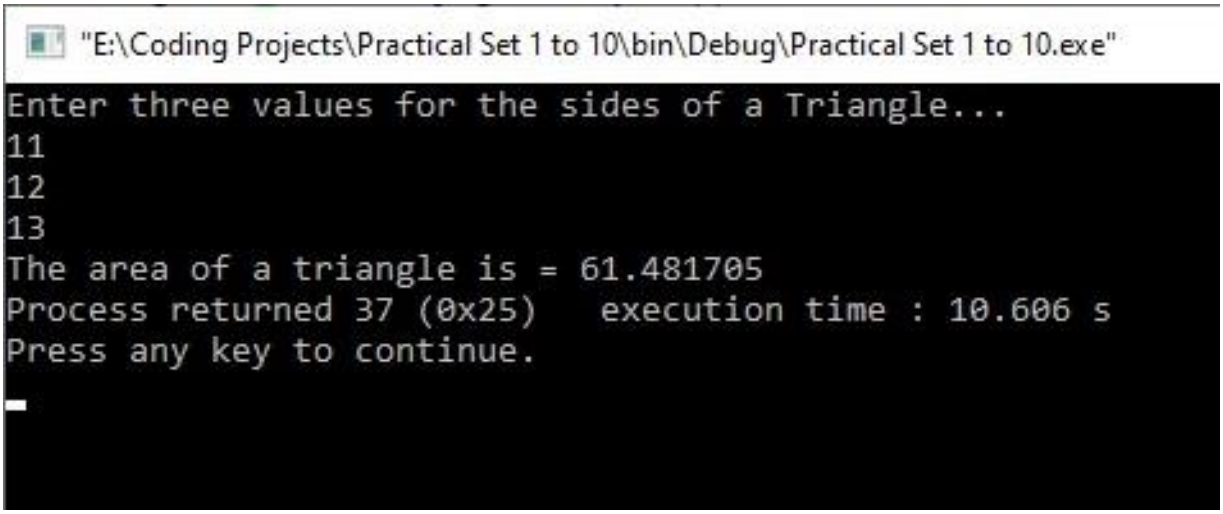
Aim :

If the length of the sides of a triangle are denoted by a, b and c, then write a program to calculate the area of a triangle.

Source Code :

```
#include<stdio.h>

#include<math.h>

void main()

{

   int p,q,r;

   float av,ans;

   printf("Enter three values for the sides of a Triangle...\n");

   scanf("%d %d %d",&p, &q, &r);

   av = (p+q+r)/2;

   ans = sqrt(av*(av-p)*(av-q)*(av-r));

   printf("The area of a triangle is = %f",ans);

}
```

Output :



Practical - 3

Aim :

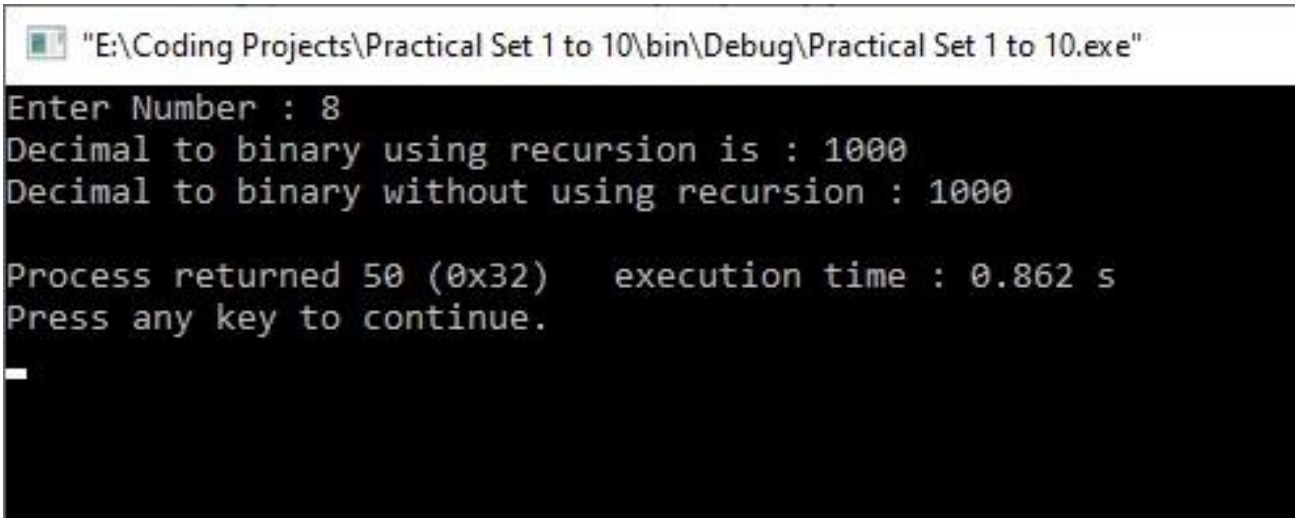
A positive integer is entered through the keyboard, write a function to find the binary equivalent of this number using recursion.

Source Code :

```
#include<stdio.h>

int non_rec_bin(int num)

{

 int x, res=0, pos=1;

 while (num!=0)

 {

 x = num%2;
```

```
 res = res + (x*pos);
 pos = 10*pos;
 num = num/2;
 }
 return res;
}
int rec_bin(int num)
{
 if(num==0)
 {
 return 0;
 }
 else
 {
 return ((num%2)+10*rec_bin(num/2));
 }
}
void main()

{
 int num;
 printf("Enter Number : ");
 scanf("%d", &num);
 printf("Decimal to binary using recursion is : %d", rec_bin(num));
 printf("\nDecimal to binary without using recursion : %d\n", non_rec_bin(num));
}
```

Output :


"E:\Coding Projects\Practical Set 1 to 10\bin\Debug\Practical Set 1 to 10.exe"
Enter Number : 8
Decimal to binary using recursion is : 1000
Decimal to binary without using recursion : 1000
Process returned 50 (0x32) execution time : 0.862 s
Press any key to continue.

Set - 10

Practical - 1

Aim :

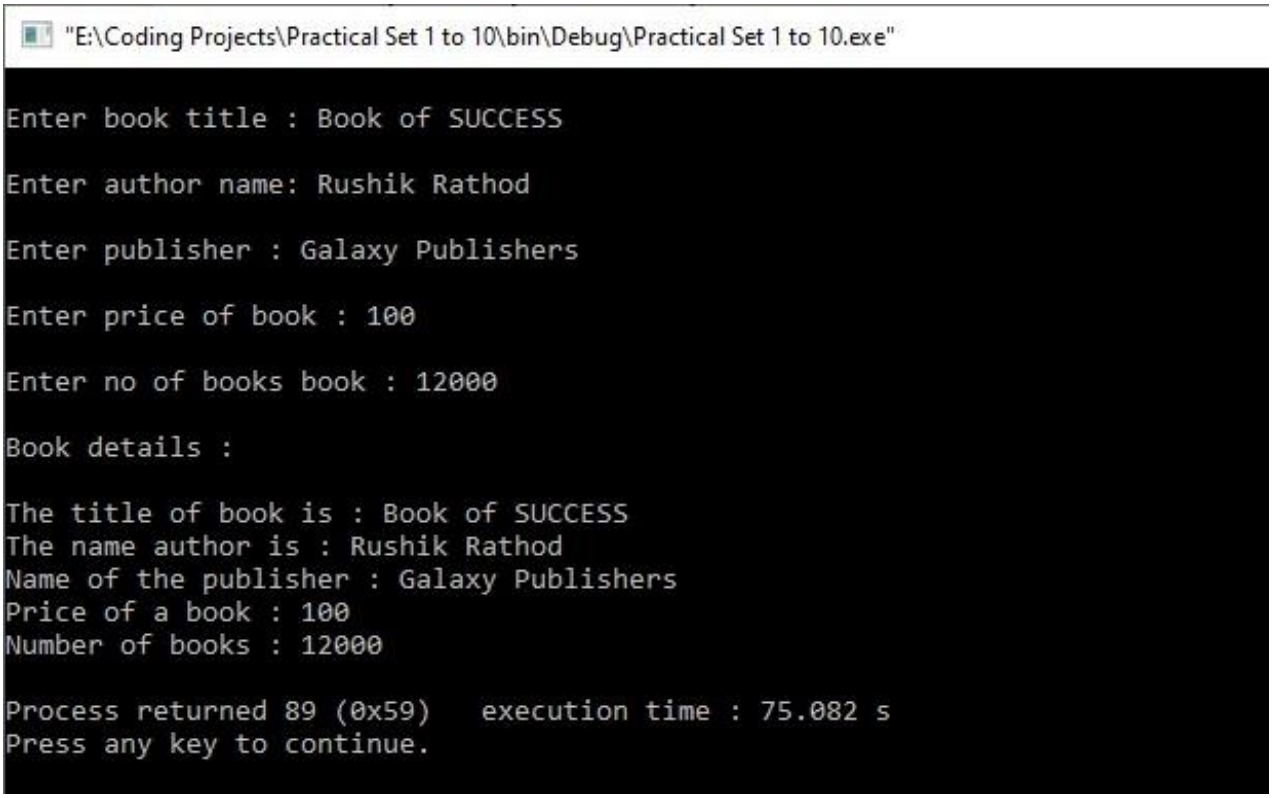
Write a C program to create a structure of Book Detail and display the details of the book in appropriate format by passing structure as function argument.

Source Code :

```
#include<stdio.h>
#include <stdlib.h>
#include<string.h>
struct Book_Detail
{
   char title[32];
   char author[32];
   char publisher[32];
   int price;
   int b_n;
}b1;

void main()
{
   printf("\nEnter book title : ");
   gets(b1.title);
   printf("\nEnter author name: ");
   gets(b1.author);
   printf("\nEnter publisher : ");
```

```
    gets(b1.publisher);
    printf("\nEnter price of book : ");
    scanf("%d",&b1.price);
    printf("\nEnter no of books book : ");
    scanf("%d",&b1.b_n);
    printf("\nBook details :\n");
    printf("\nThe title of book is : %s\nThe name author is : %s",b1.title,b1.author);
    printf("\nName of the publisher : %s\nPrice of a book : %d\nNumber of books :
%d\n",b1.publisher,b1.price,b1.b_n);
}
```

Output :



Practical - 2

Aim :

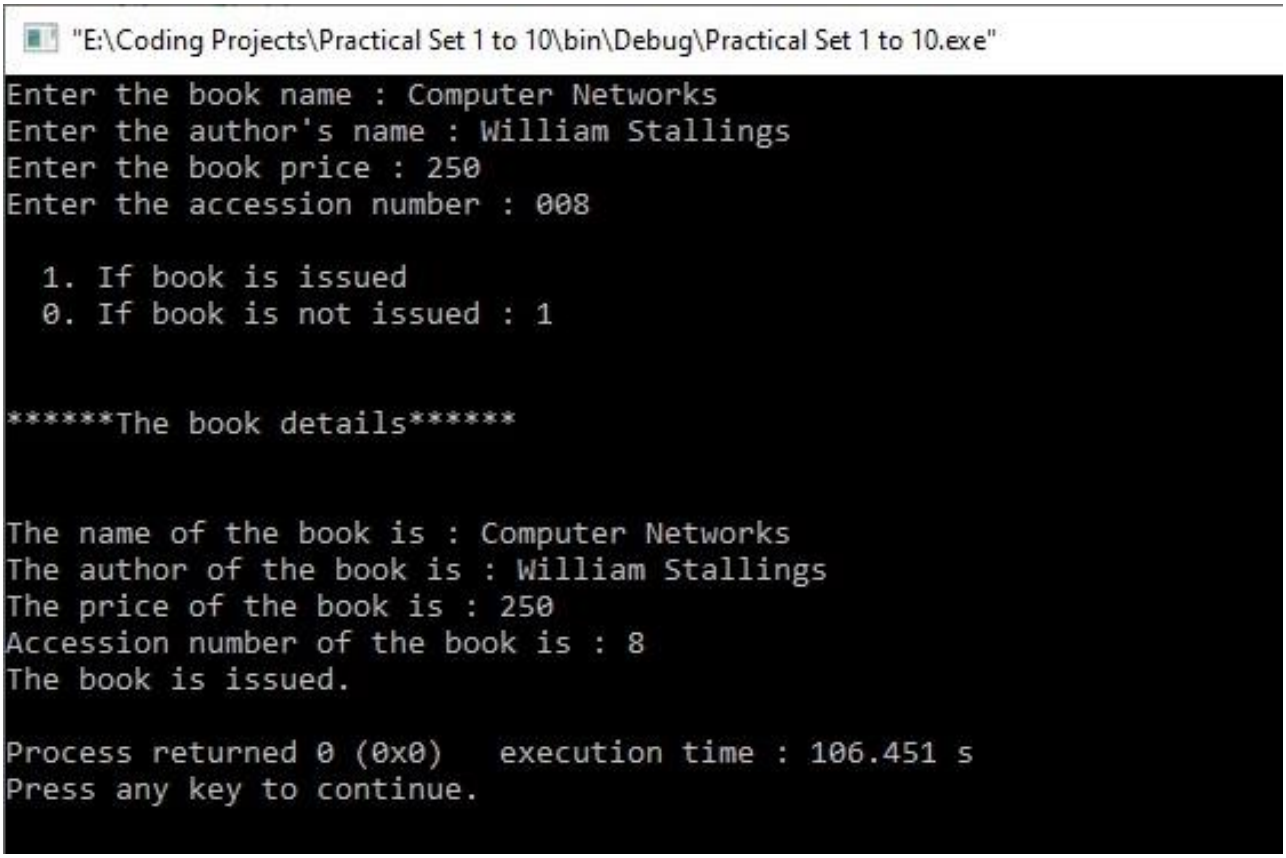
Create a Union called library to hold accession number, title of the book ,author name, price of the book and flag indicating whether the book is issued or not.(flag = 1 if the book is issued , flag = 0 otherwise). Write a program to enter data of one book and display the data.

Source Code :

```
#include<stdio.h>
#include <string.h>
union book_detail
{
 char title[32];
 char author[32];
 int price;
 int accession_no;
}b1;
void main()
{
 int flag;
 char t[32],a[32];
 int p,acc;
 printf("Enter the book name : ");
 gets(t);
 printf("Enter the author's name : ");
 gets(a);
 printf("Enter the book price : ");
 scanf("%d",&p);
```

```
 printf("Enter the accession number : ");
 scanf("%d",&acc);
 printf("\n  1. If book is issued");
 printf("\n  0. If book is not issued : ");
 scanf("\n%d",&flag);
 printf("\n\n******The book details******\n\n");
 strcpy(b1.title,t);
 printf("\nThe name of the book is : %s",b1.title);
 strcpy(b1.author,a);
 printf("\nThe author of the book is : %s",b1.author);
 b1.price=p;
 printf("\nThe price of the book is : %d",b1.price);
 b1.accession_no=acc;
 printf("\nAccession number of the book is : %d\n",b1.accession_no);
 if(flag)
 {
   printf("The book is issued.\n");
 }
 else
 {
   printf("The book is not issued.\n");
 }
}
```

Output :


"E:\Coding Projects\Practical Set 1 to 10\bin\Debug\Practical Set 1 to 10.exe"
Enter the book name : Computer Networks
Enter the author's name : William Stallings
Enter the book price : 250
Enter the accession number : 008
1. If book is issued
0. If book is not issued : 1
******The book details******
The name of the book is : Computer Networks
The author of the book is : William Stallings
The price of the book is : 250
Accession number of the book is : 8
The book is issued.
Process returned 0 (0x0) execution time : 106.451 s
Press any key to continue.

Practical - 3

Aim :

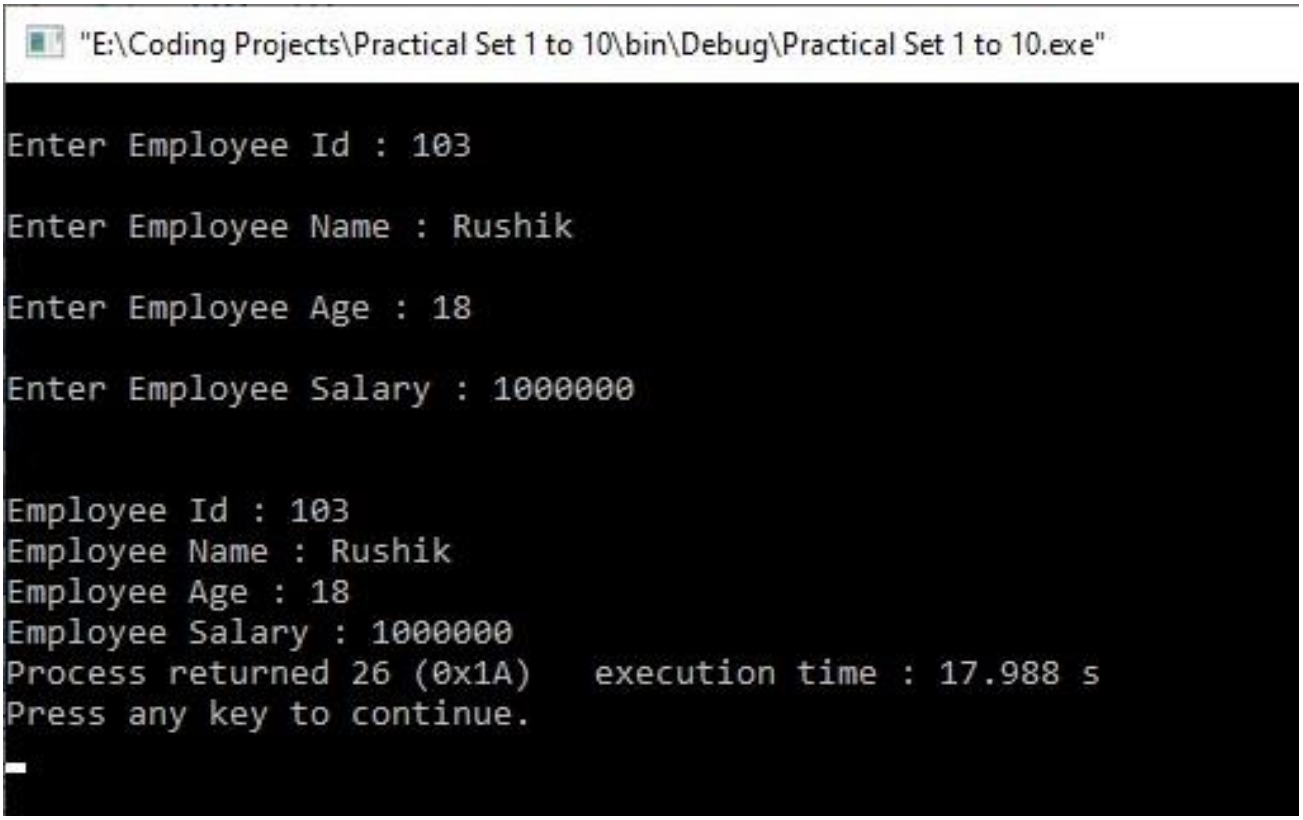
Write a C program for nested structure to display employee details such as, Age, Name, Address, Salary.

Source Code :

```
#include<stdio.h>


 struct Employee
 {
 int Id;
 char Name[25];
 int Age;
 long Salary;
 };
 void main()
 {


 struct Employee E; //Statement 1
 printf("\nEnter Employee Id : ");
 scanf("%d",&E.Id);
 printf("\nEnter Employee Name : ");
 scanf("%s",&E.Name);
 printf("\nEnter Employee Age : ");
 scanf("%d",&E.Age);
 printf("\nEnter Employee Salary : ");
```

```
scanf("%ld",&E.Salary);
printf("\n\nEmployee Id : %d",E.Id);
printf("\nEmployee Name : %s",E.Name);
printf("\nEmployee Age : %d",E.Age);
printf("\nEmployee Salary : %ld",E.Salary);
}
```

Output :



Thank you… :)