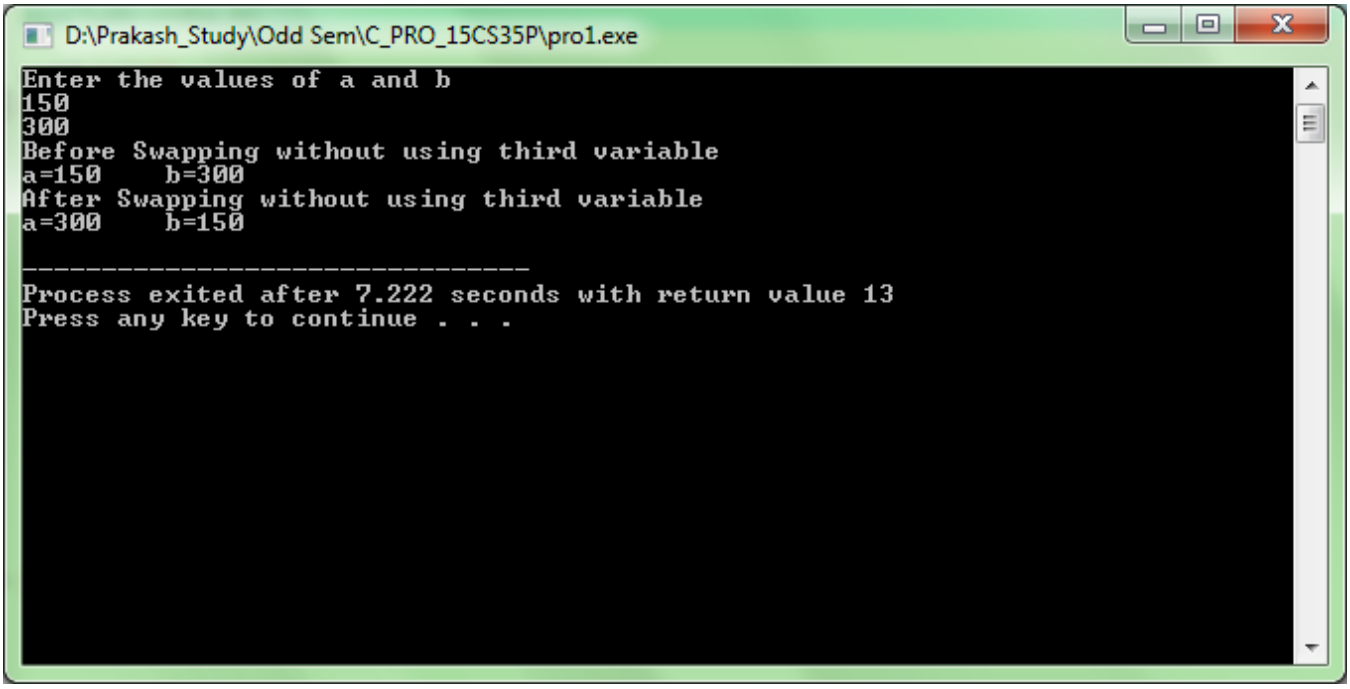
1. **WAP to swap the values of two variables without using third variable.**

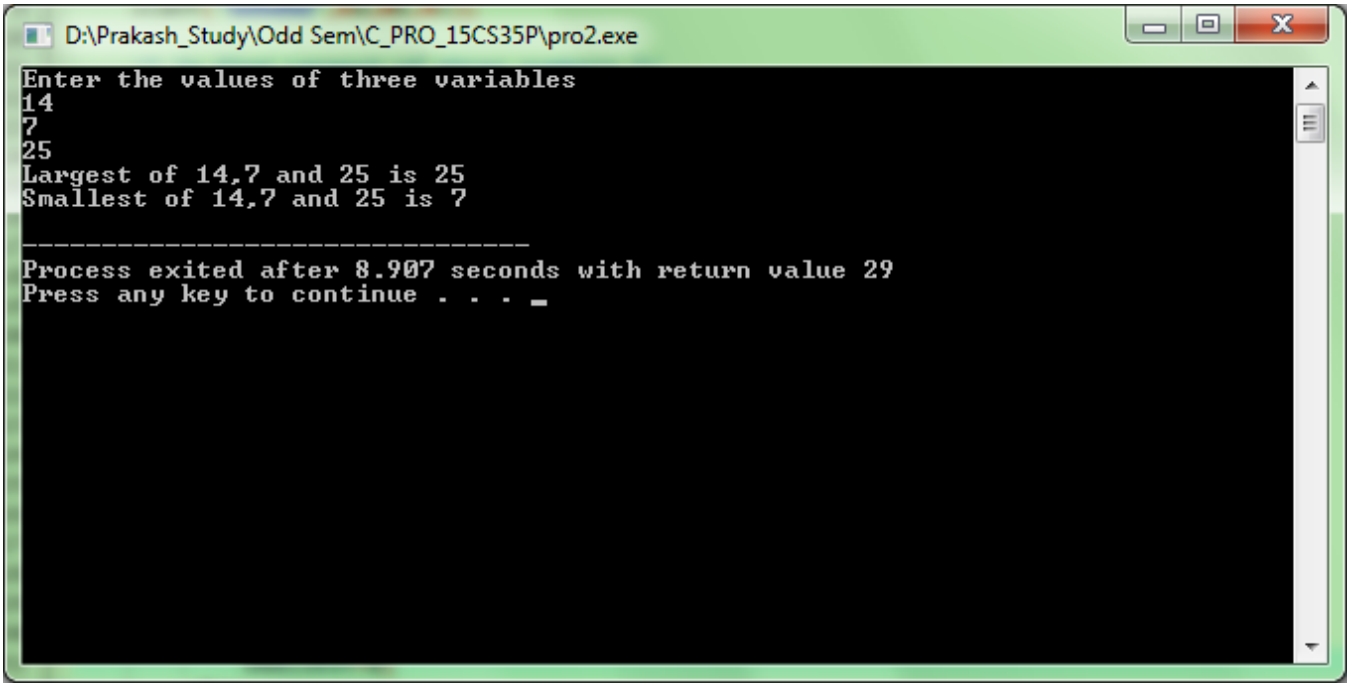
```
#include<stdio.h>
#include<conio.h>
  void main()
  {
        int x,y;

        printf("Enter the values of a and b\n");
        scanf("%d%d",&x,&y);

        printf("Before Swapping without using third variable\n");
        printf("a=%d\t b=%d\n",x,y);

        /* swap the values without using third variable */
        x=x+y;
        y=x-y;
        x=x-y;

        printf("After Swapping without using third variable\n");
        printf("a=%d\t b=%d\n",x,y);
        getch();
  }
```



2. WAP to find the largest/smallest of 3 numbers (if-else).

```
#include<stdio.h>
#include<conio.h>
void main()
   {
        int a,b,c;
        int largest,smallest;

        printf("Enter the values of three variables\n");
        scanf("%d%d%d",&a,&b,&c);

        /* To find largest of three numbers */
        if(a>b)
        {
             if(a>c)
                  largest=a;
             else
                  largest=c;
         }
        else
        {
             if(b>c)
                  largest=b;
             else
                  largest=c;
        }

        /* To find smallest of three numbers */
        if(a<b)
        {
             if(a<c)
                  smallest=a;
             else
                  smallest=c;
         }
        else
        {
             if(b<c)
                  smallest=b;
             else
                  smallest=c;
        }
        printf("Largest of %d,%d and %d is %d\n",a,b,c,largest);
        printf("Smallest of %d,%d and %d is %d\n",a,b,c,smallest);
        getch();
```

```
}
```

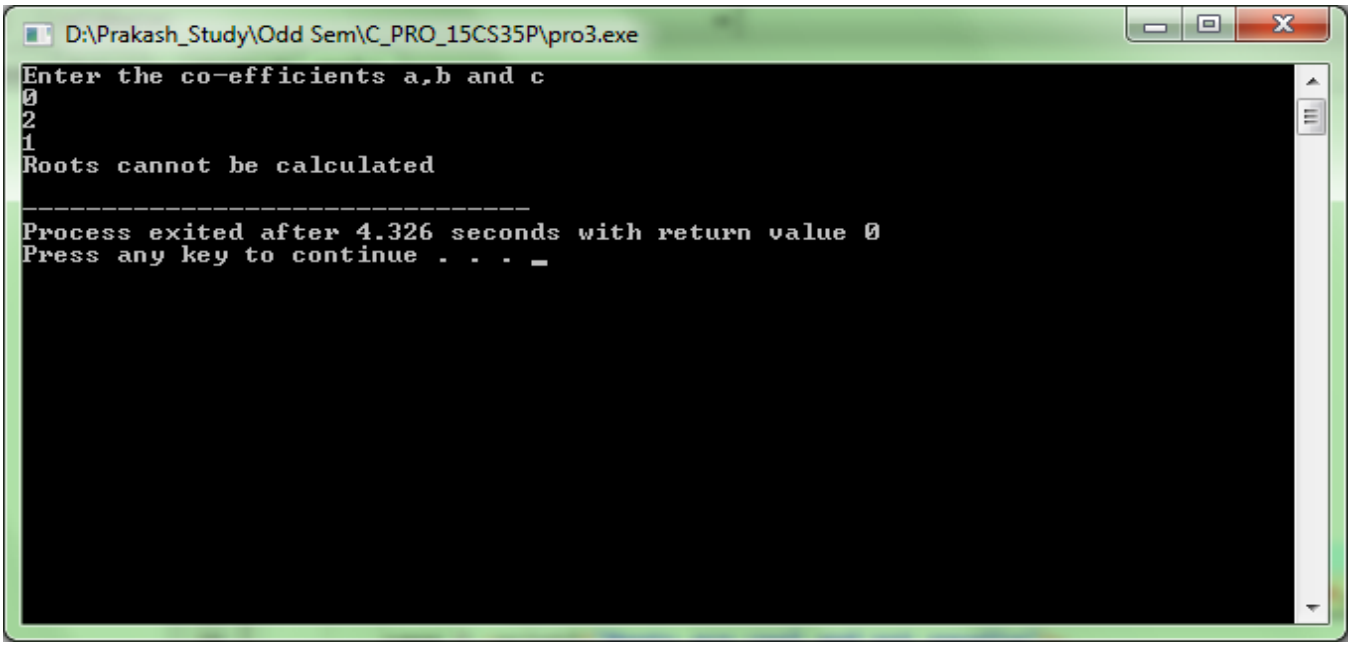

D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro2.exe
Enter the values of three variables
14
7
25
Largest of 14,7 and 25 is 25
Smallest of 14,7 and 25 is 7

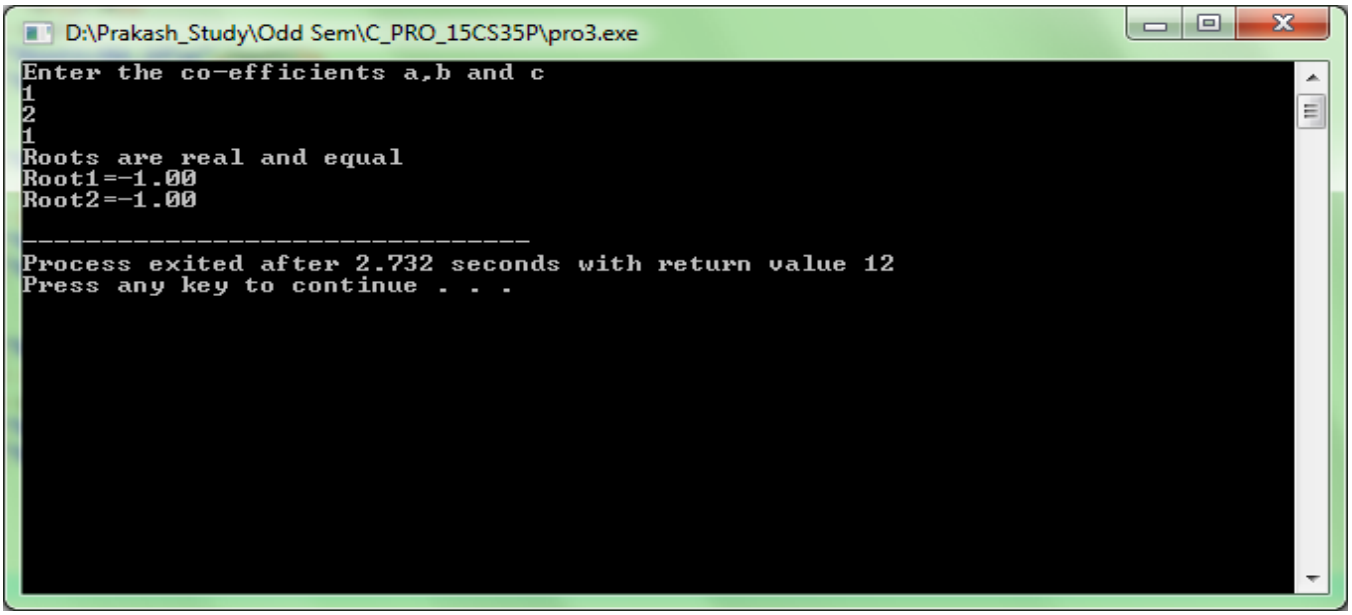
Process exited after 8.907 seconds with return value 29
Press any key to continue . . .

3. WAP to calculate the roots of a quadratic equation (using switch).

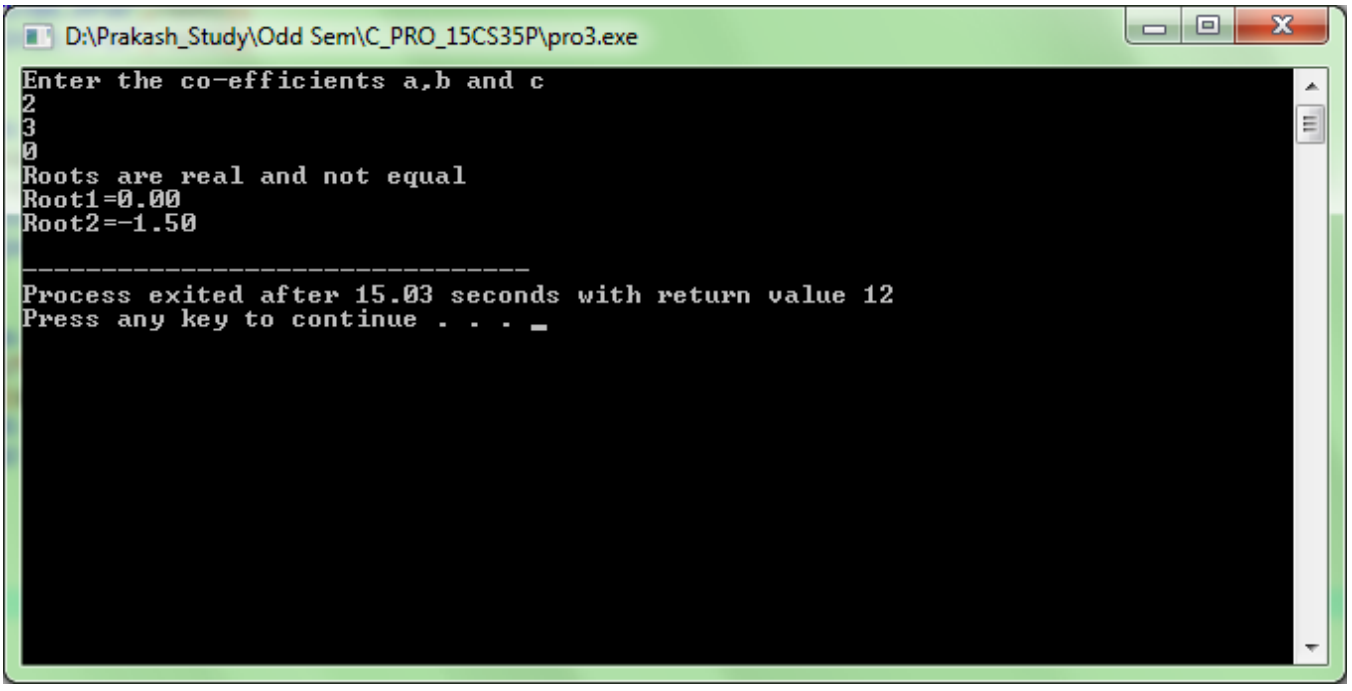
```
#include<stdio.h>
#include<math.h>
#include<conio.h>
void main()
{
     float a,b,c;
     float root1,root2,disc;
     int option;

     printf("Enter the co-efficients a,b and c\n");
     scanf("%f%f%f",&a,&b,&c);

     if(a == 0.0)
     {
          printf("Roots cannot be calculated \n");
     }
     else
     {
          disc = (b*b)-(4*a*c);
          if(disc == 0.0)
               option = 1;
          else if(disc > 0.0)
               option = 2;
          else
               option = 3;

          switch(option)
          {
          case 1 :  printf("Roots are real and equal\n");
                    root1 = -b/(2*a);
                    root2 = root1;
                    printf("Root1=%4.2f\n",root1);
                    printf("Root2=%4.2f\n",root2);
                    break;

          case 2 :  printf("Roots are real and not equal\n");
                    root1 = (-b+sqrt(disc))/(2*a);
                    root2 = (-b-sqrt(disc))/(2*a);
                    printf("Root1=%4.2f\n",root1);
                    printf("Root2=%4.2f\n",root2);
                    break;

          case 3 :  printf("Roots are imaginary\n");
                    root1 = -b/(2*a);
```

```
                        root2 = sqrt(abs(disc))/(2*a);
                        printf("Root1 = %4.2f +i
%4.2f\n",root1,root2);
                        printf("Root2 = %4.2f -i
%4.2f\n",root1,root2);
                        break;
                }
        }
`       getch();
}
```

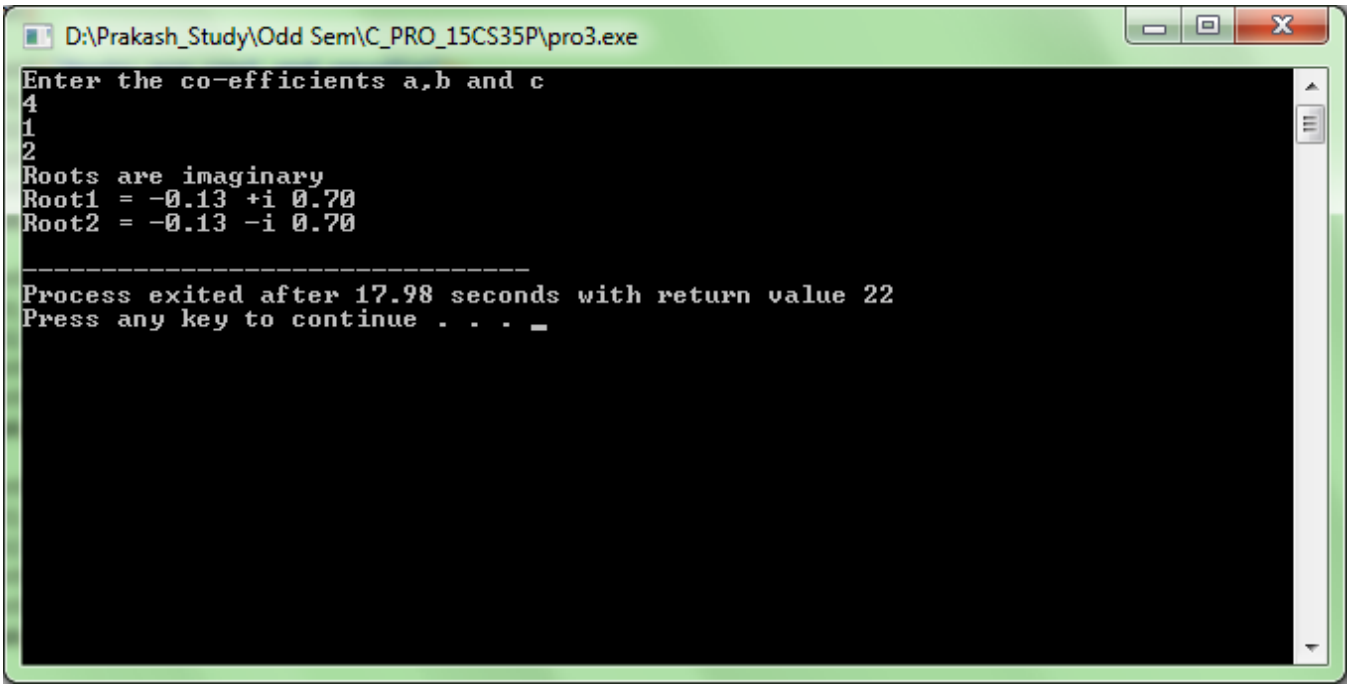



D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro3.exe
Enter the co-efficients a,b and c
1
2
1
Roots are real and equal
Root1=-1.00
Root2=-1.00

Process exited after 2.732 seconds with return value 12
Press any key to continue . . .


D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro3.exe
Enter the co-efficients a,b and c
2
3
0
Roots are real and not equal
Root1=0.00
Root2=-1.50

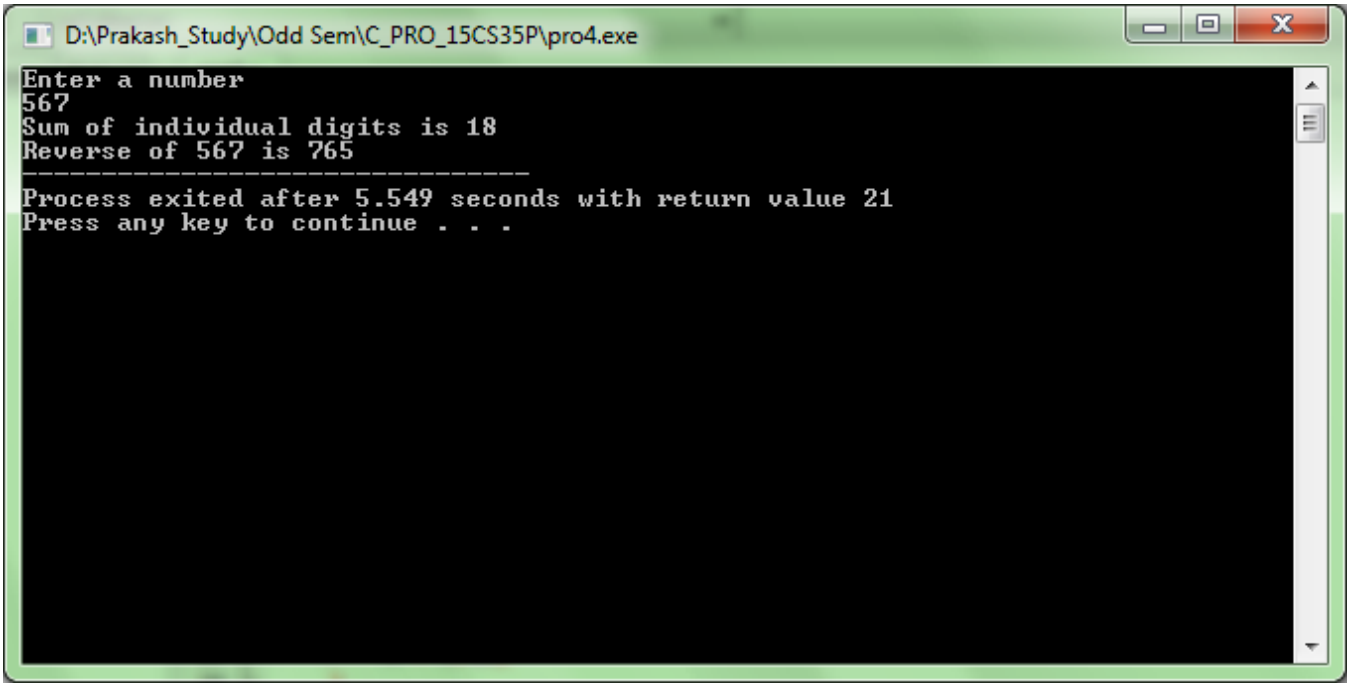
Process exited after 15.03 seconds with return value 12
Press any key to continue . . .


D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro3.exe
Enter the co-efficients a,b and c
4
1
2
Roots are imaginary
Root1 = -0.13 +i 0.70
Root2 = -0.13 -i 0.70

Process exited after 17.98 seconds with return value 22
Press any key to continue . . .

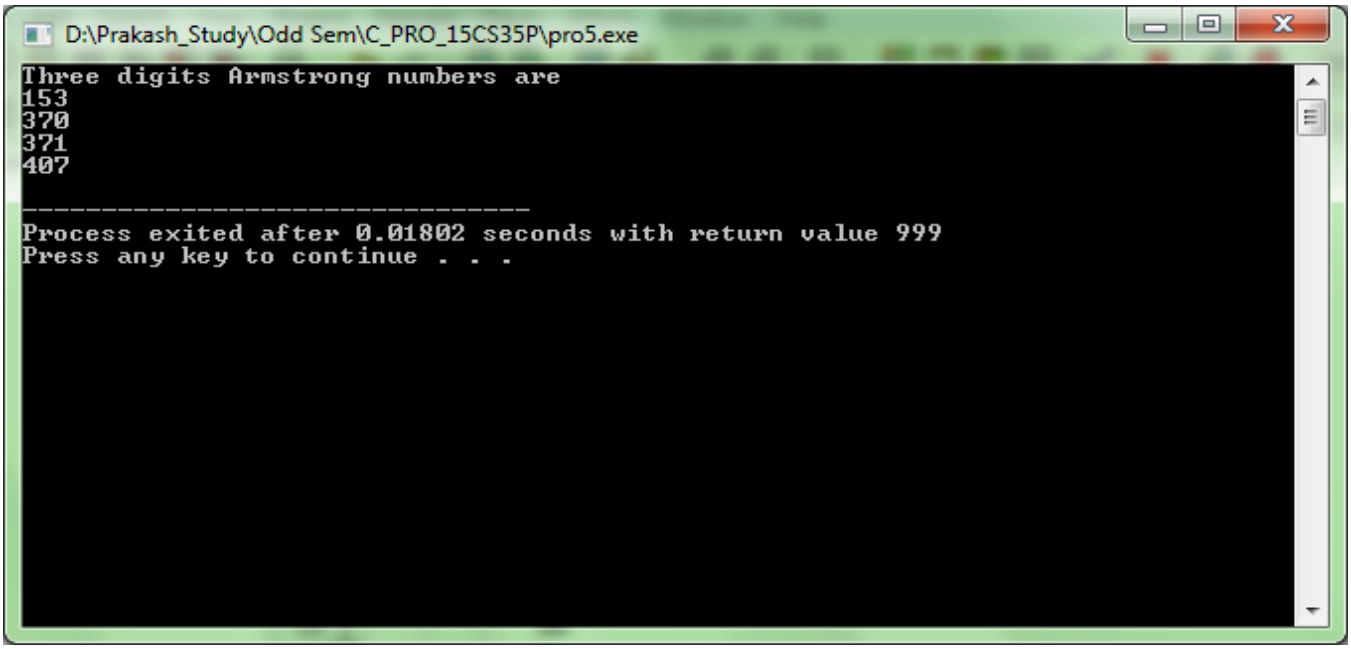
4. **WAP to sum & reverse a given integer (while loop).**

```
#include<stdio.h>
#include<conio.h>
void main()
{
      int n, temp;
      int sum=0, rev=0, rem;

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

      /* Logic to find sum of individual digits and reverse */
      temp = n;
      while(n!=0)
      {
            rem = n % 10;
            sum = sum+rem;
            rev = (rev*10) + rem;
            n = n/10;
      }
      printf("Sum of individual digits is %d \n",sum);
      printf("Reverse of %d is %d",temp,rev);
      getch();
}
```



5. WAP to detect the Armstrong numbers in three digits from 100 to 999. (do-while).

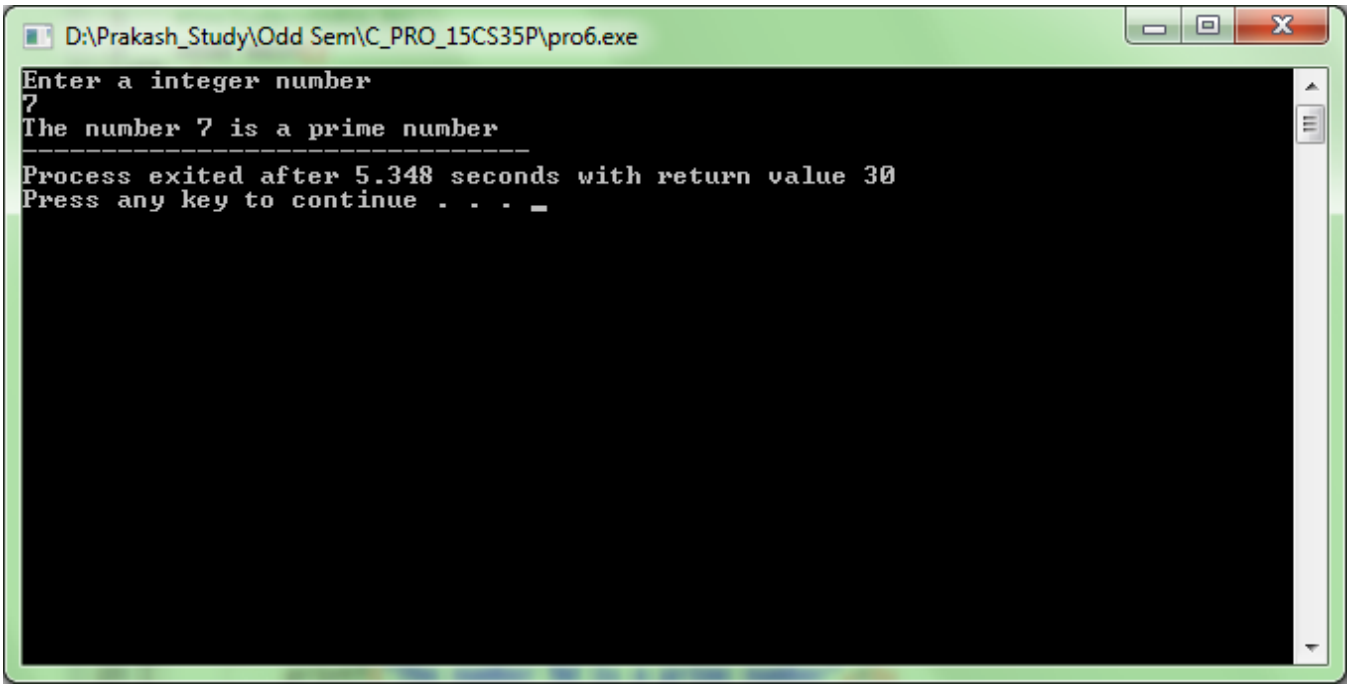
```
#include<stdio.h>
#include<math.h>
#include<conio.h>
void main()
{
   int i, j, k, number, sumcube;
   i=1;

   printf("Three digits Armstrong numbers are\n");
   do
   {
        j = 0;
        do
        {
             k=0;
             do
             {
                  number = (i*100)+(j*10)+(k*1);
                  sumcube = (i*i*i)+(j*j*j)+(k*k*k);
                  if(number == sumcube)
                       printf("%d\n",number);
             k++;
             }while(k<=9);
        j++;
        }while(j<=9);
   i++;
    }while(i<=9);
   getch();
}
```

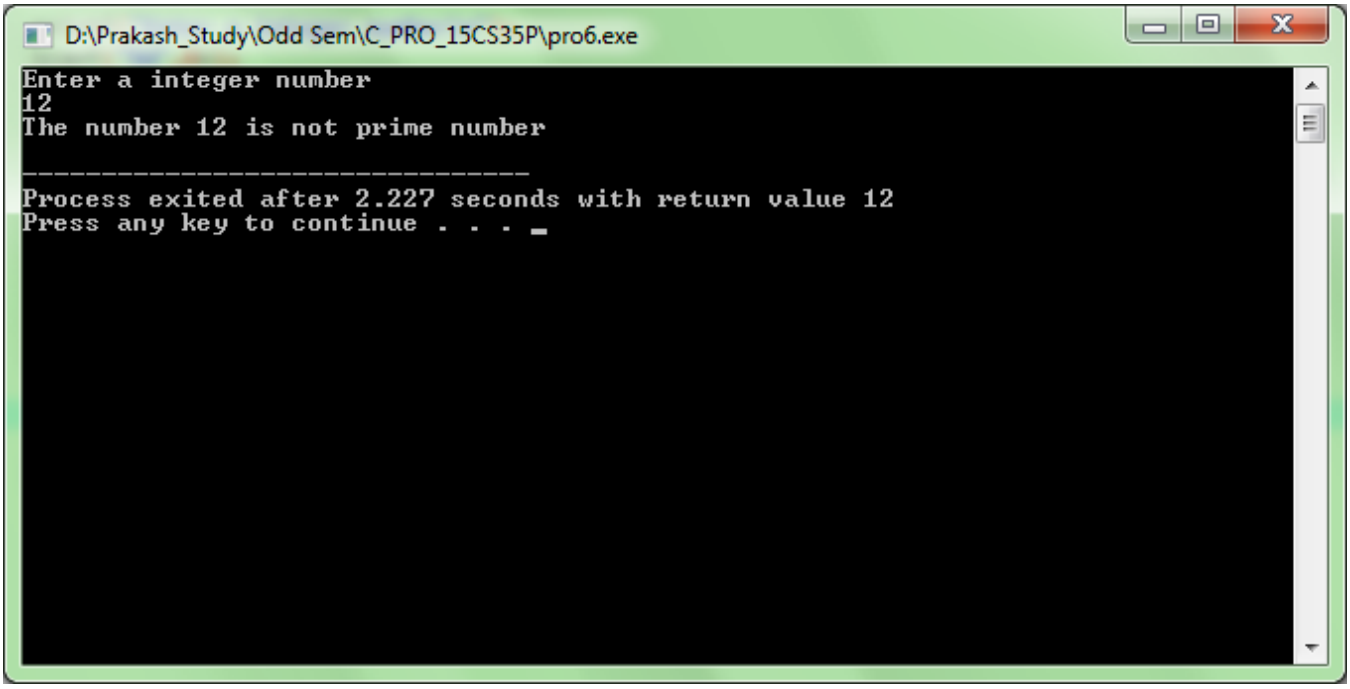

D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro5.exe
Three digits Armstrong numbers are
153
370
371
407

Process exited after 0.01802 seconds with return value 999
Press any key to continue . . .

6. WAP to check whether the given number is prime or not (for loop).

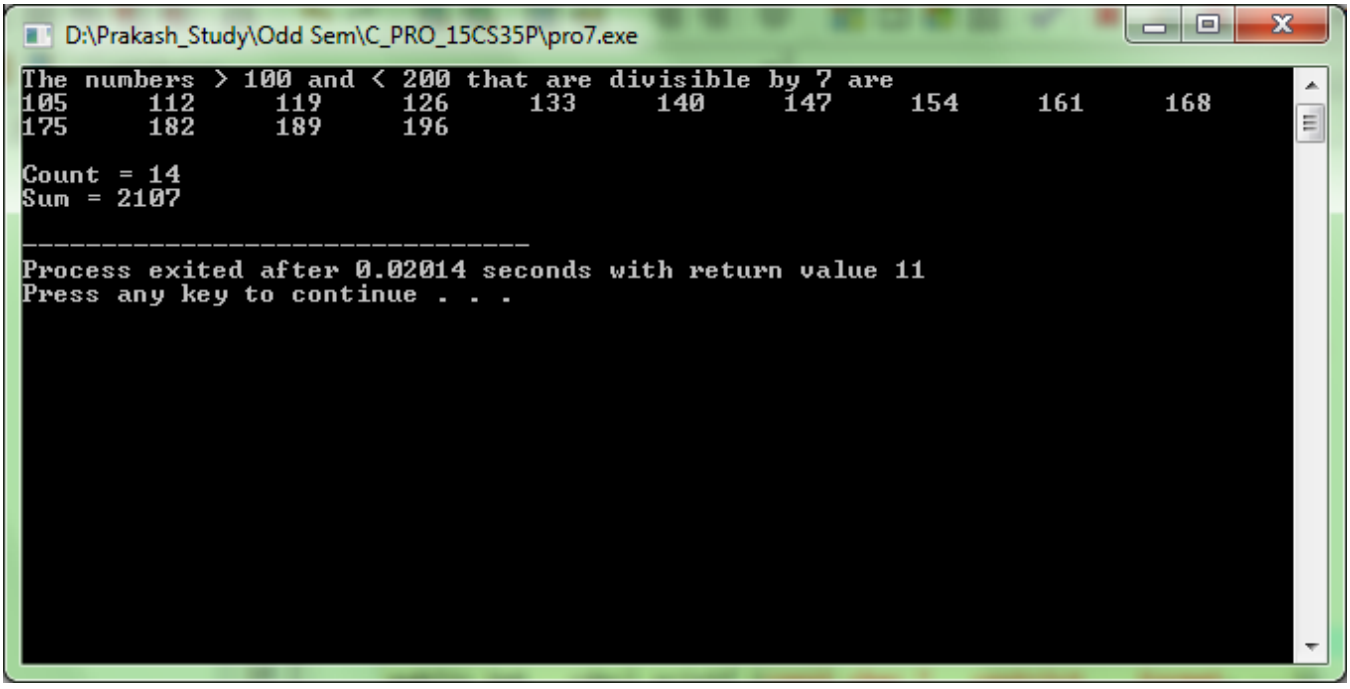
```
#include<stdio.h>
#include<conio.h>
void main()
 {

   int n;
   int i;

   printf("Enter a integer number\n");
   scanf("%d",&n);

   /* Logic to check whether the number is prime? */

   for(i=2;i<n;i++)
   {
        if(n%i == 0)
        {
             printf("The number %d is not prime number\n",n);
             break;
        }
   }
   if(i==n)
   {
        printf("The number %d is a prime number",n);
   }
   getch();
}
```


D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro6.exe
Enter a integer number
7
The number 7 is a prime number
Process exited after 5.348 seconds with return value 30
Press any key to continue . . .


D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro6.exe
Enter a integer number
12
The number 12 is not prime number
Process exited after 2.227 seconds with return value 12
Press any key to continue . . .

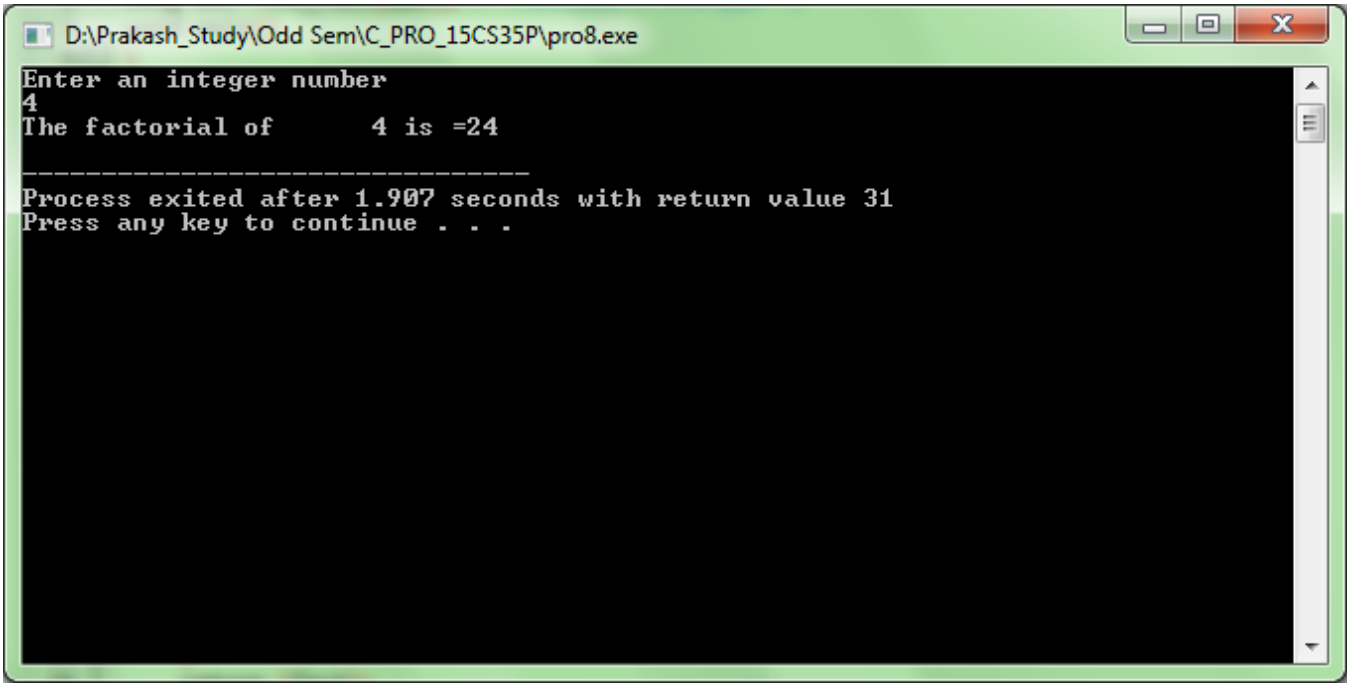
7. **WAP to find the number of and sum of all integers greater than 100 and less than 200 (that are divisible by 7 for loop).**

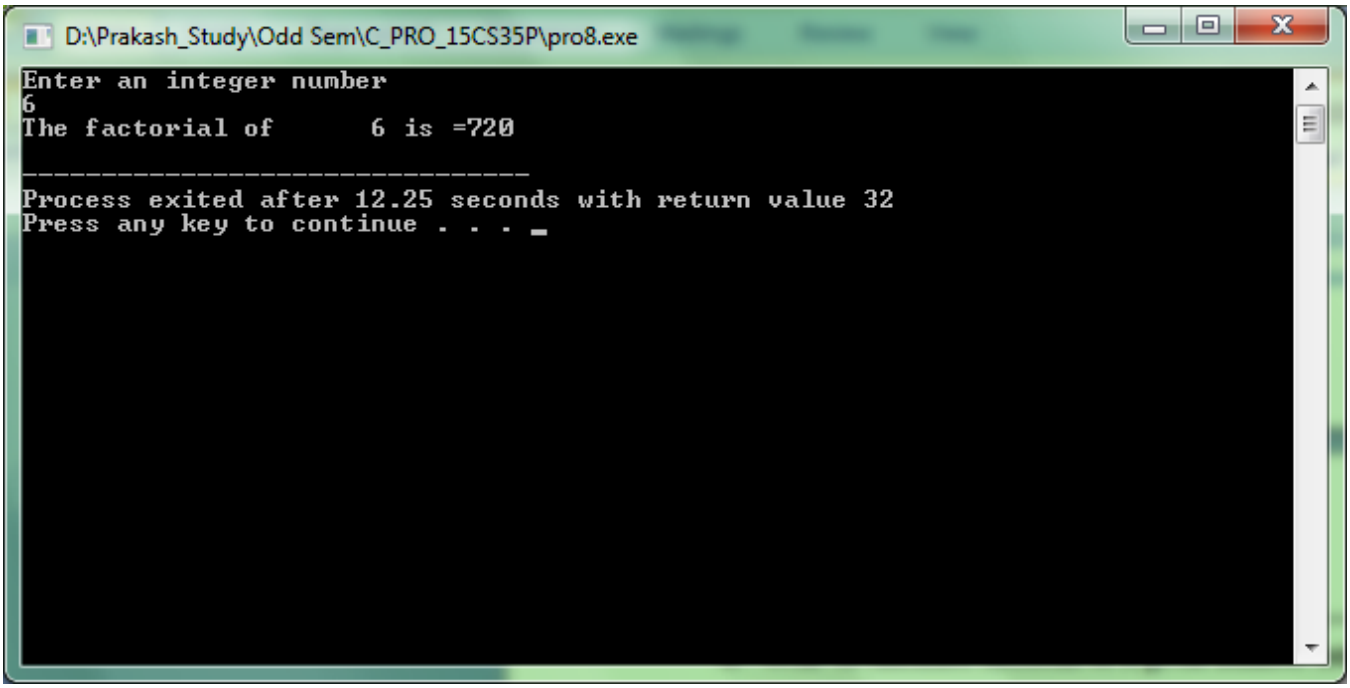
```
#include<stdio.h>
#include<conio.h>
void main()
{
      int sum=0,count=0,i;

      printf("The numbers > 100 and < 200 that are divisible by 7
      are\n");
      for(i=101;i<200;i++)
      {
            if(i%7 == 0)
            {
                  printf("%d\t",i);
                  count=count+1;
                  sum=sum+i;
            }
      }
      printf("\n\nCount = %d\n",count);
      printf("Sum = %d\n",sum);
      getch();
}
```



8. WAP to calculate factorial of a given number using function.

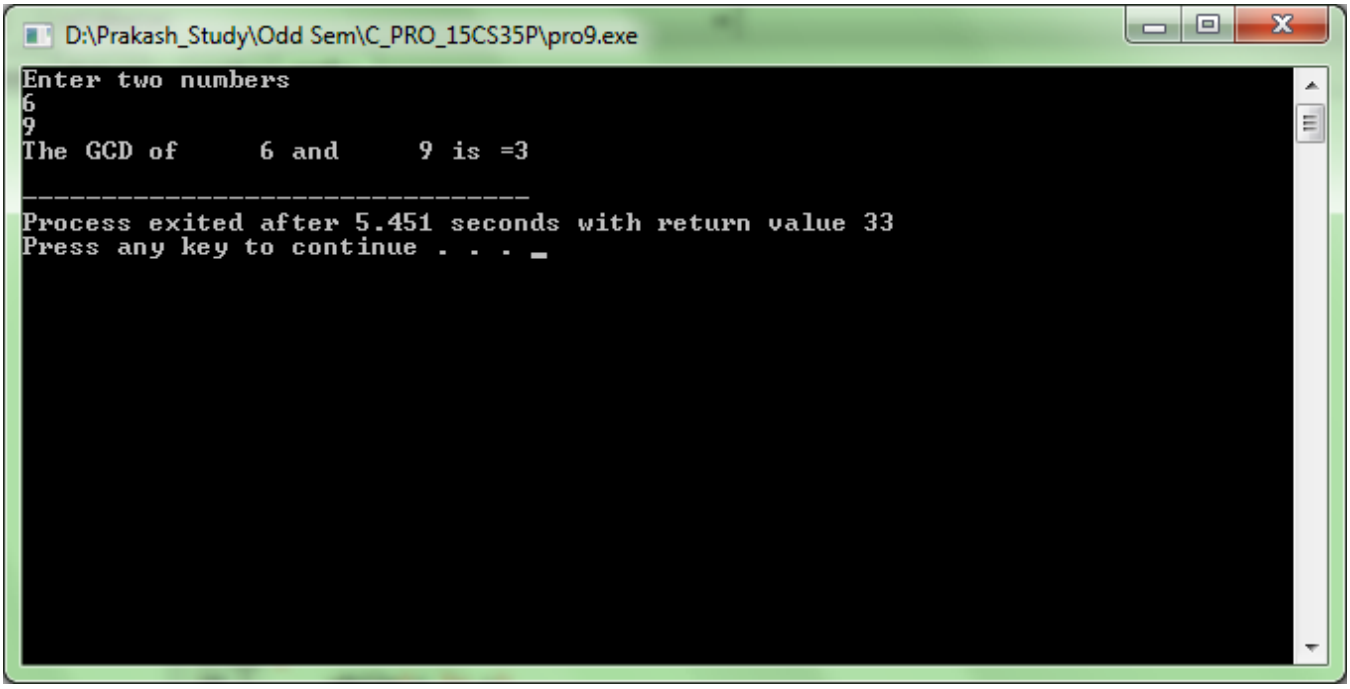
```
#include<stdio.h>
#include<conio.h>
long int find_fact(int); /* Function Prototype */
void main()
{
   int i,n;
   long int result;

   printf("Enter an integer number\n");
   scanf("%d",&n);
   result = find_fact(n); /* Function Call */
   printf("The factorial of %6d is =%ld\n",n,result);
   getch();
}

long int find_fact(int m) /* Function Definition */
{
   int i;
   long int fact=1;

   for(i=1; i<=m; i++)
   {
        fact = fact * i;
   }
   return (fact);
}
```

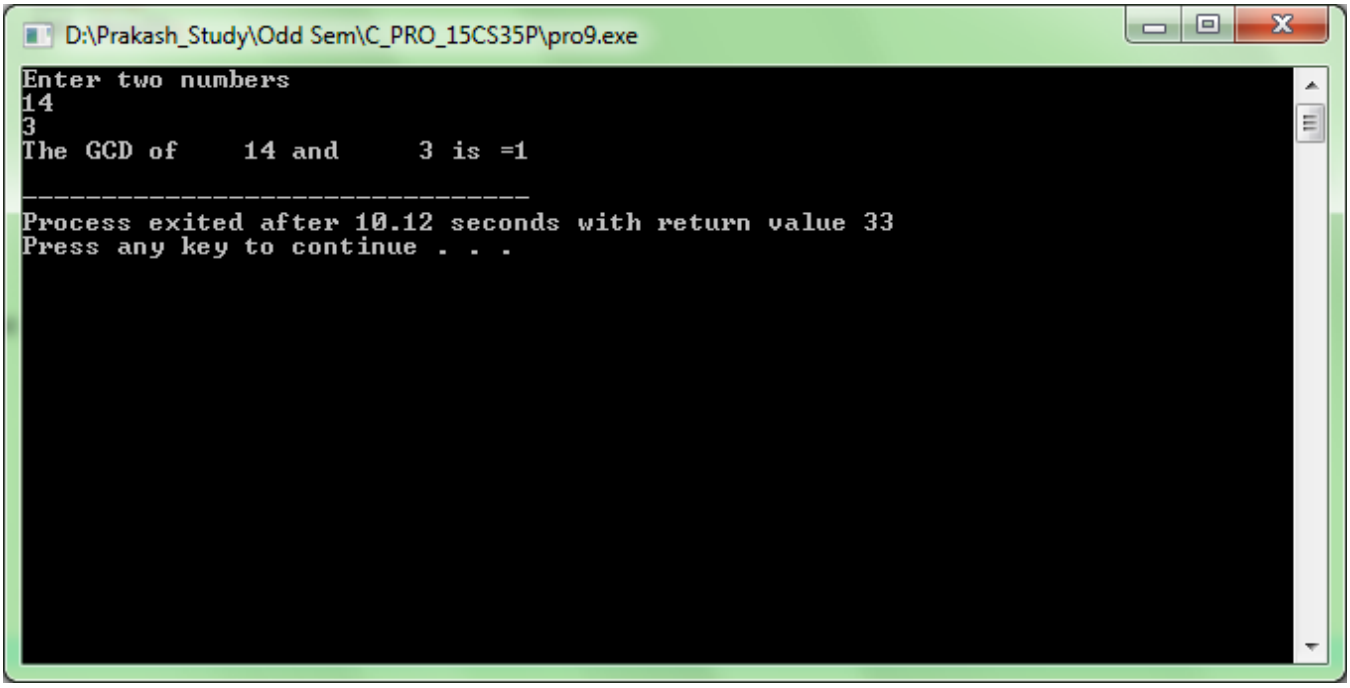

D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro8.exe
Enter an integer number
4
The factorial of 4 is =24
Process exited after 1.907 seconds with return value 31
Press any key to continue . . .


D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro8.exe
Enter an integer number
6
The factorial of 6 is =720
Process exited after 12.25 seconds with return value 32
Press any key to continue . . .

9. WAP to find GCD of two numbers using function

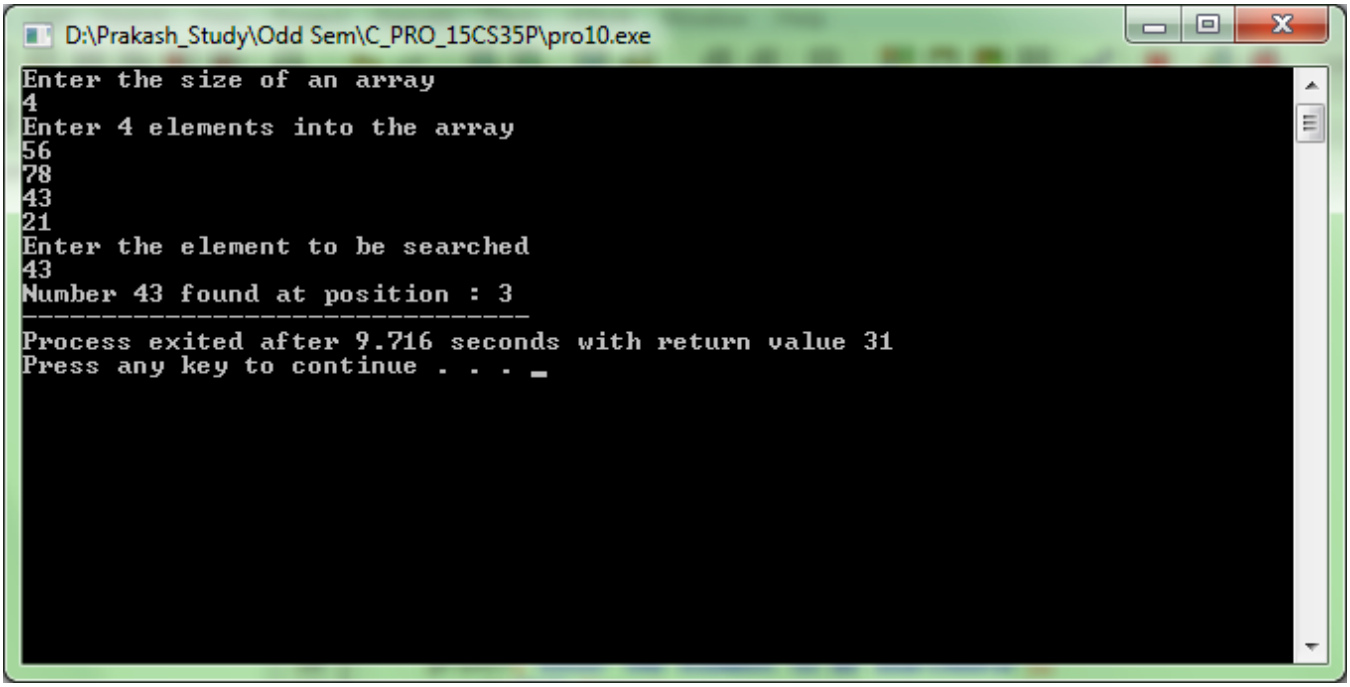
```
#include<stdio.h>
#include<conio.h>
int find_gcd(int ,int ); /* Function Prototype */
void main()
{

   int m,n;
   int result;

   printf("Enter two numbers \n");
   scanf("%d%d",&m,&n);

   result = find_gcd(m,n);  /* Function Call */
   printf("The GCD of %5d and %5d is =%d\n",m,n,result);\
   getch();
}

int find_gcd(int x,int y) /* Function Definition */
{
   while(x != y)
   {
        if(x > y)
             x = x - y;
        else if(y>x)
             y = y - x;
   }
   return (x);
}
```

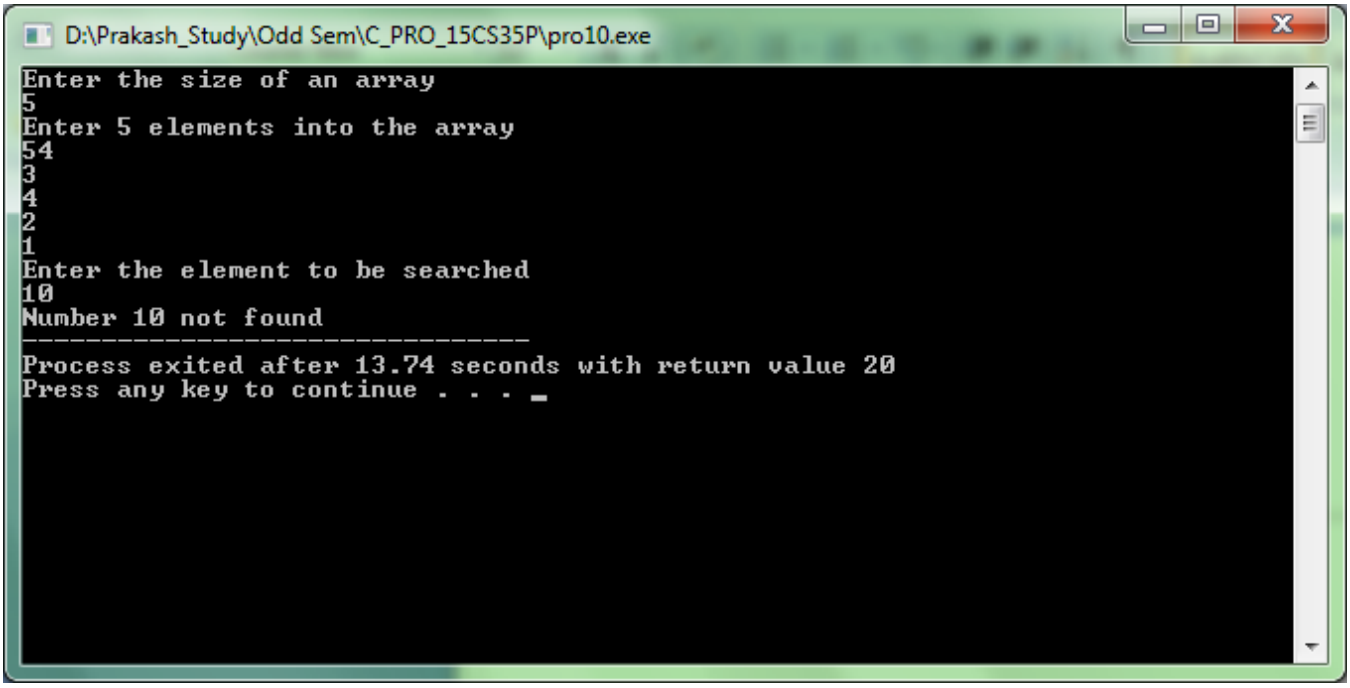

D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro9.exe
Enter two numbers
6
9
The GCD of 6 and 9 is =3
Process exited after 5.451 seconds with return value 33
Press any key to continue . . .


D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro9.exe
Enter two numbers
14
3
The GCD of 14 and 3 is =1
Process exited after 10.12 seconds with return value 33
Press any key to continue . . .

10. WAP to search for a given number in an array

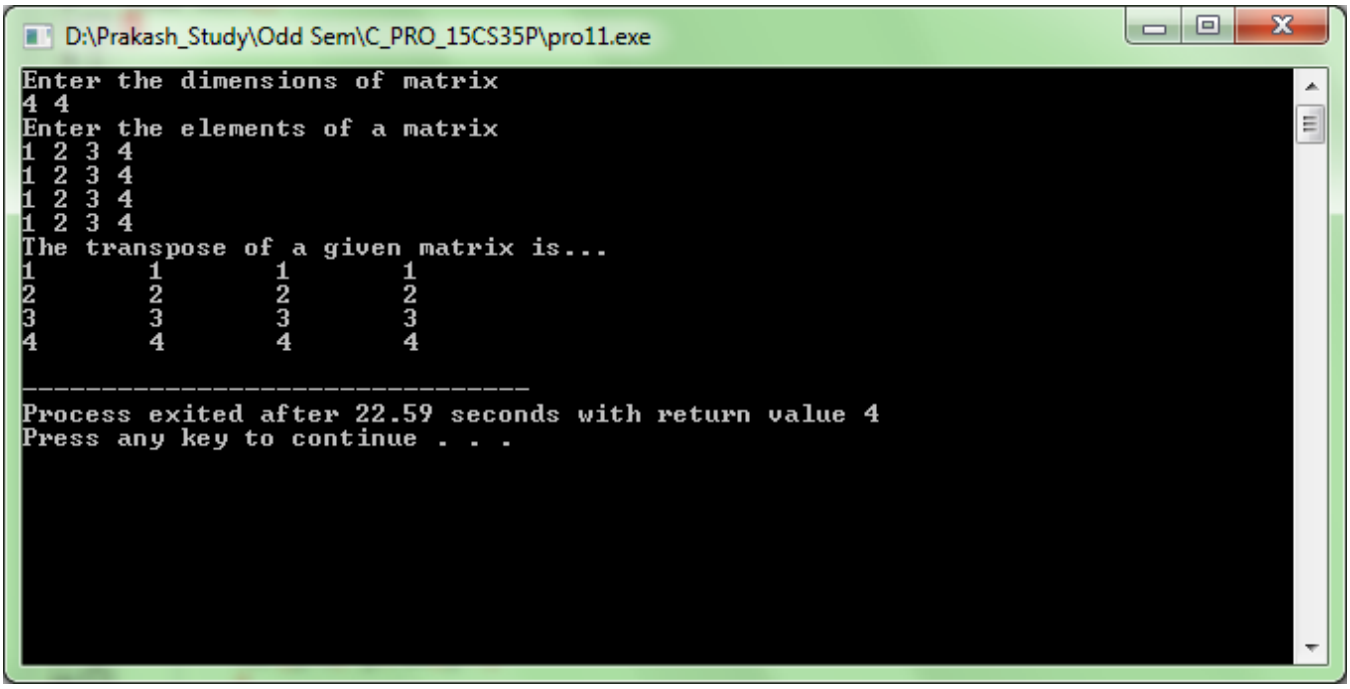
```
#include<stdio.h>
#include<conio.h>
void main()
{
   int arr[20],i,n,item,flag=0;

   printf("Enter the size of an array\n");
   scanf("%d",&n);

   printf("Enter %d elements into the array\n",n);
   for(i=0; i<n; i++)
   {
        scanf("%d",&arr[i]);
   }

   printf("Enter the element to be searched\n");
   scanf("%d",&item);

   for(i=0;i<n;i++)
   {
        if(arr[i]==item)
        {
             flag=1;
             break;
        }
   }
   if(flag==1)
             printf("Number %d found at position : %d",item,(i+1));
   else
             printf("Number %d not found ",item);
   getch();
}
```


D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro10.exe
Enter the size of an array
4
Enter 4 elements into the array
56
78
43
21
Enter the element to be searched
43
Number 43 found at position : 3
Process exited after 9.716 seconds with return value 31
Press any key to continue . . .


D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro10.exe
Enter the size of an array
5
Enter 5 elements into the array
54
3
4
2
1
Enter the element to be searched
10
Number 10 not found
Process exited after 13.74 seconds with return value 20
Press any key to continue . . .

11. WAP to find the transpose of a given matrix

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

      int matrix[10][10];
      int transpose[10][10];
      int row,column,i,j;

      printf("Enter the dimensions of matrix\n");
      scanf("%d%d",&row,&column);

      printf("Enter the elements of a matrix\n");
      for(i=0; i<row; i++)
      {
            for(j=0; j<column;j++)
            {
                  scanf("%d",&matrix[i][j]);
            }
      }

      /* Logic to transpose a matrix*/
      for(i=0; i<column; i++)
      {
            for(j=0; j<row;j++)
            {
                  transpose[i][j] = matrix[j][i];
            }
      }
      printf("The transpose of a given matrix is...\n");
      for(i=0; i<column; i++)
      {
            for(j=0; j<row; j++)
            {
                  printf("%d\t",transpose[i][j]);
            }
            printf("\n");
      }
      getch();
}
```

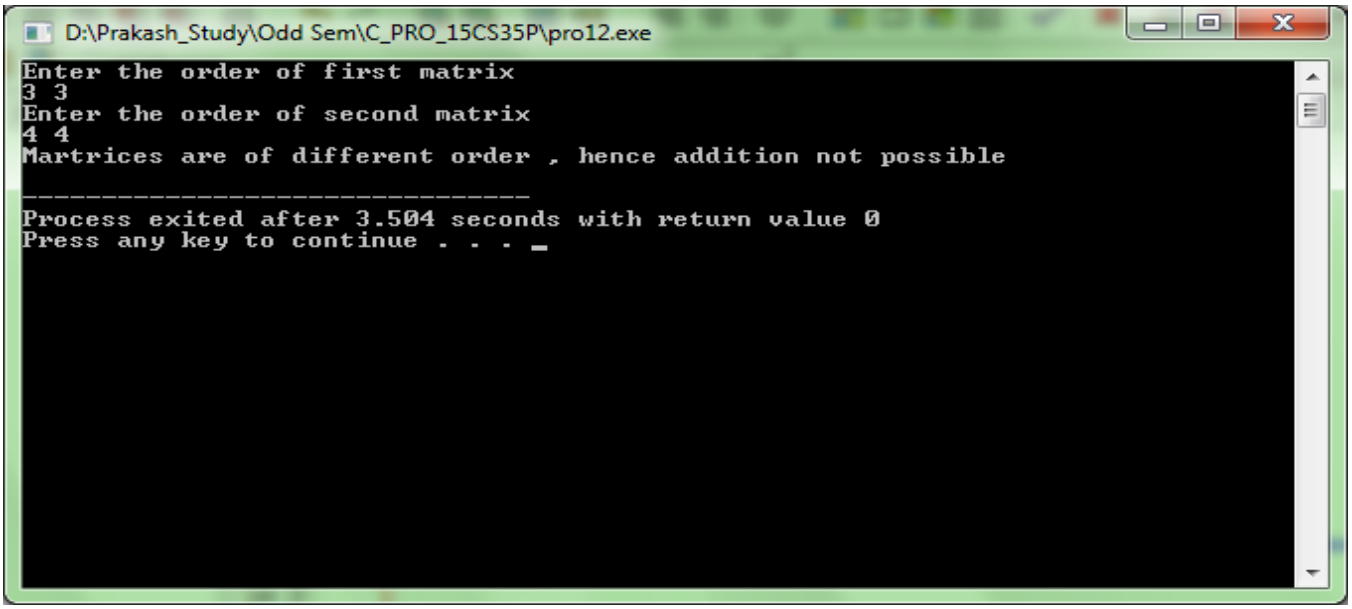

D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro11.exe
Enter the dimensions of matrix
4 4
Enter the elements of a matrix
1 2 3 4
1 2 3 4
1 2 3 4
1 2 3 4
The transpose of a given matrix is...
1 1 1 1
2 2 2 2
3 3 3 3
4 4 4 4

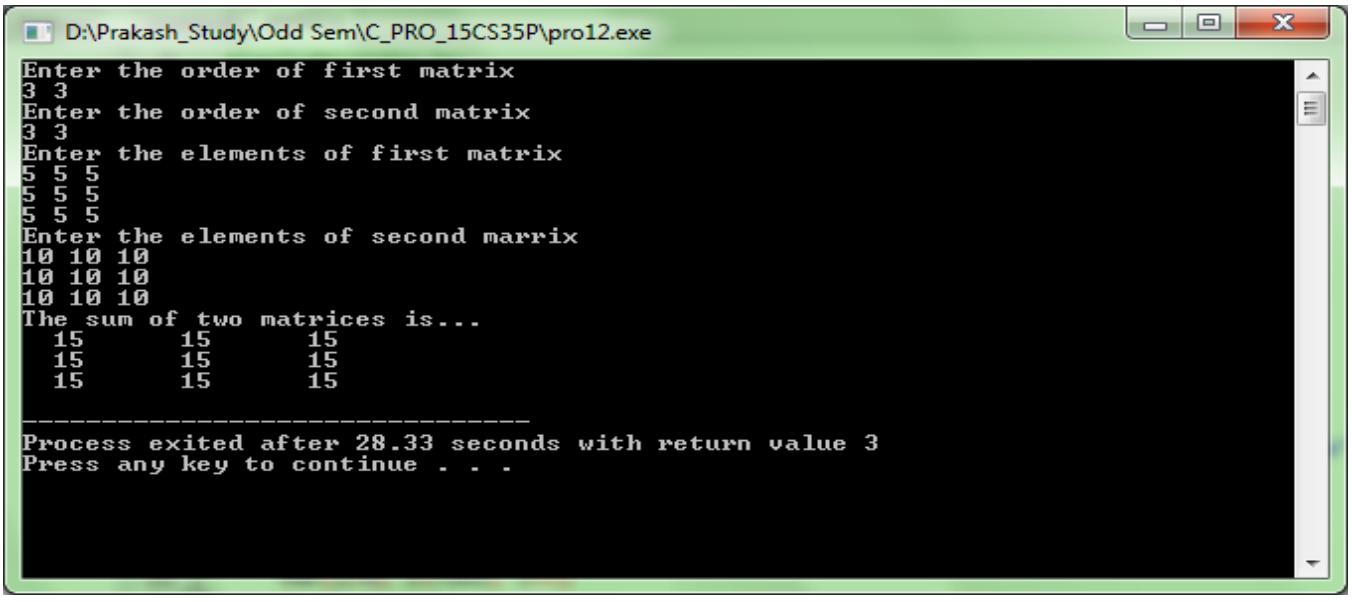
Process exited after 22.59 seconds with return value 4
Press any key to continue . . .

12. WAP to addition two matrices

```
#include<stdio.h>
#include<conio.h>
void main()
{
     int matA[10][10],matB[10][10];
     int matSum[10][10];
     int row1,column1,row2,column2,i,j;

     printf("Enter the order of first matrix\n");
     scanf("%d%d",&row1,&column1);

     printf("Enter the order of second matrix\n");
     scanf("%d%d",&row2,&column2);

     if((row1 != row2 )||( column1 != column2))
     {
          printf("Martrices are of different order , hence
          addition not possible\n");
          exit(0);
     }
     /* if matrices order are same*/
     printf("Enter the elements of first matrix \n");
     for(i=0; i<row1; i++)
     {
          for(j=0; j<column2;j++)
          {
          scanf("%d",&matA[i][j]);
          }
     }

     printf("Enter the elements of second marrix \n");
     for(i=0; i<row2; i++)
     {
          for(j=0; j<column2; j++)
          {
               scanf("%d",&matB[i][j]);
          }
     }

     /* logic for addition of two matrices */
     for(i=0; i<row1; i++)
          for(j=0; j<column1; j++)
               matSum[i][j] = matA[i][j] + matB[i][j];
```

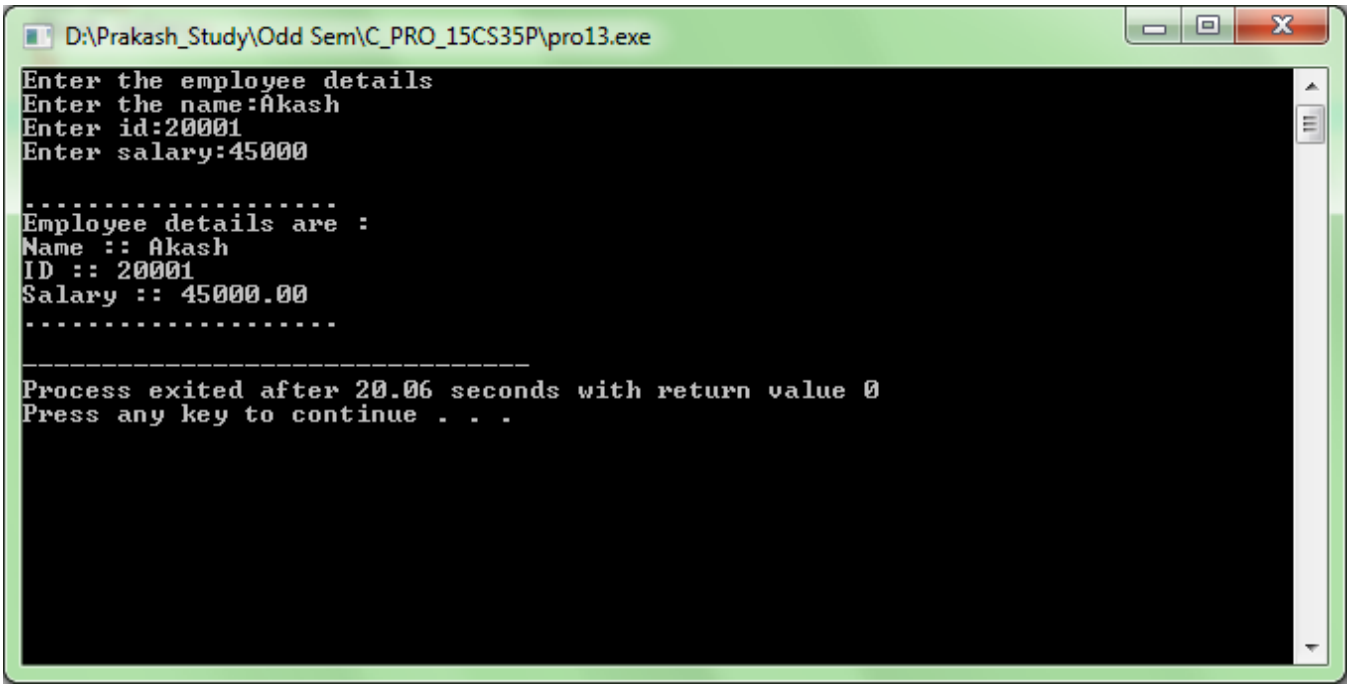
```
        printf("The sum of two matrices is...\n");
        for(i=0; i<row1; i++)
        {
              for(j=0; j<column1; j++)
              {
                    printf("%4d\t",matSum[i][j]);
              }
              printf("\n");
        }
        getch();
  }
```





13. WAP to create a structure with employee details and display the same

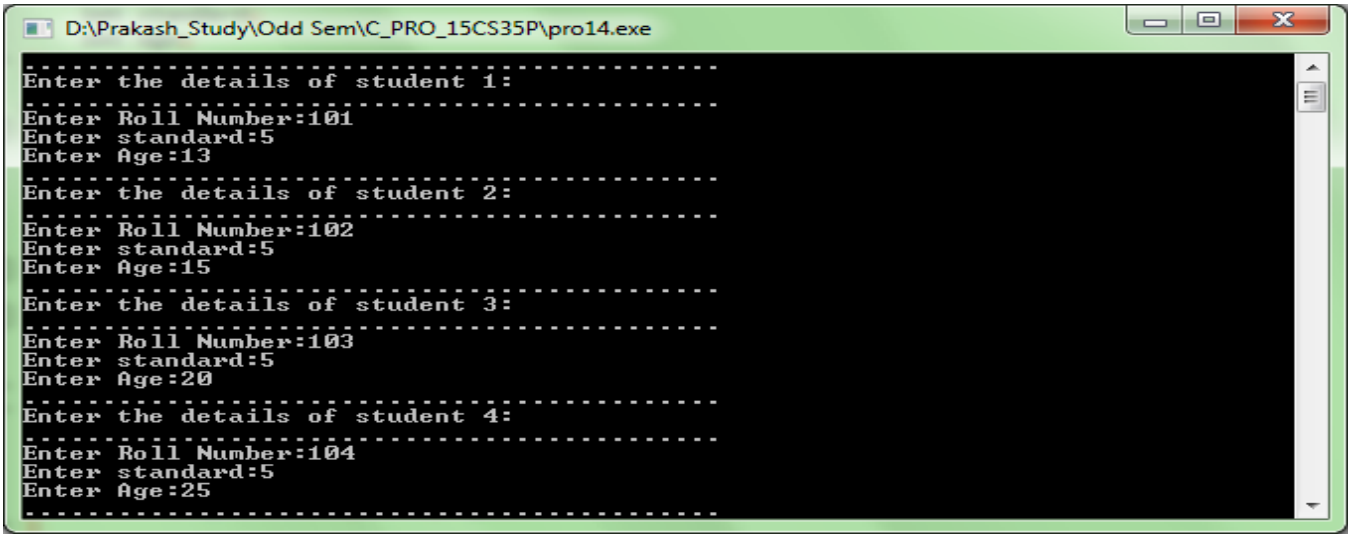
```
#include<stdio.h>
#include<conio.h>
void main()
{
      struct employee
      {
            char name[25];
            int id;
            float salary;
      };
      struct employee emp;

      printf("Enter the employee details\n");
      printf("Enter the name:");
      scanf("%s",emp.name);

      printf("Enter id:");
      scanf("%d",&emp.id);

      printf("Enter salary:");
      scanf("%f",&emp.salary);

      printf("\n.....................\n");
      printf("Employee details are :\n");
      printf("Name :: %s\n",emp.name);
      printf("ID :: %d\n",emp.id);
      printf("Salary :: %8.2f\n",emp.salary);
      printf(".....................\n");
      getch();
}
```

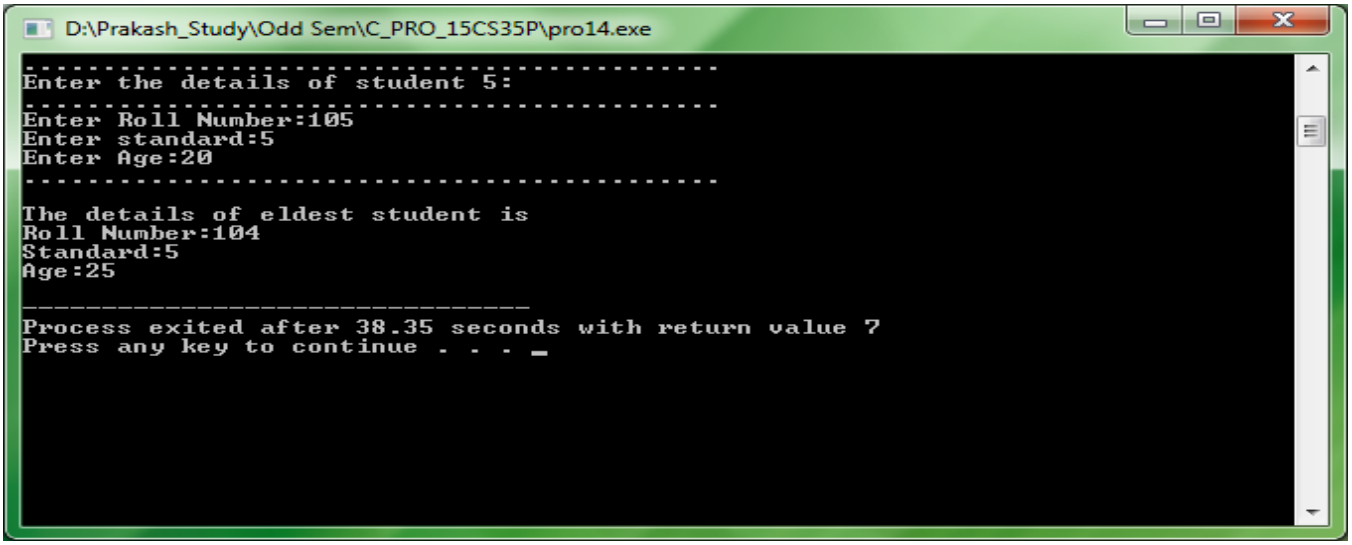

D:\Prakash_Study\Odd Sem\C_PRO_15CS35P\pro13.exe
Enter the employee details
Enter the name:Akash
Enter id:20001
Enter salary:45000
.....
Employee details are :
Name :: Akash
ID :: 20001
Salary :: 45000.00
.....
Process exited after 20.06 seconds with return value 0
Press any key to continue . . .

14. WAP to process student structure containing roll number, class and age as members. The program must read 5 student records in an array of structure and display the details of a student who is eldest. Use a function to find the eldest for which array of structure is an argument.

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

struct student
{
     int roll_num;
     int standard;
     int age;
};

void find_eldest(struct student s[],int size)
{
     int eldest,i,index;

     eldest = s[0].age;
     index = 0;
     for(i=1; i<size; i++)
     {
          if(s[i].age > eldest)
          {
               eldest = s[i].age;
               index = i;
          }
     }
     printf("\nThe details of eldest student is\n");
     printf("Roll Number:%d\n",s[index].roll_num);
     printf("Standard:%d\n",s[index].standard);
     printf("Age:%d\n",s[index].age);
}

void main()
{
     struct student s[5];
     int i;

     printf("..........................................
     \n");
     for(i=0; i<5; i++)
     {
          printf("Enter the details of student %d:\n",i+1);
```

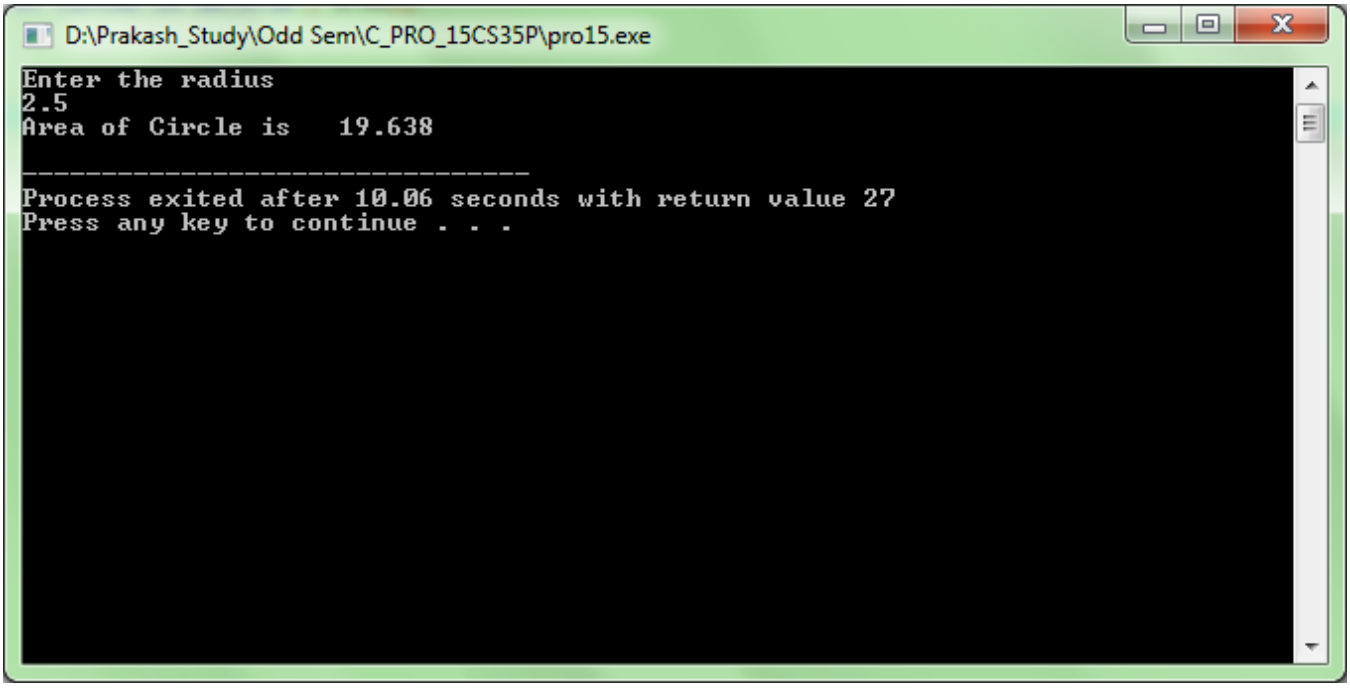
```
            printf("..........................................
        \n");
            printf("Enter Roll Number:");
            scanf("%d",&s[i].roll_num);
            printf("Enter standard:");
            scanf("%d",&s[i].standard);
            printf("Enter Age:");
            scanf("%d",&s[i].age);
            printf("..................................... \n");
        }
        find_eldest(s,5);
        getch();
    }
```





15. WAP to demonstrate # define function.

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

#define PI 3.142
#define SQUARE(X) (X*X)

void main()
{
      float area;
      float radius;
      printf("Enter the radius\n");
      scanf("%f",&radius);

      area = PI * SQUARE(radius);
      printf("Area of Circle is %8.3f\n", area);
      getch();
}
```



"Study like there's no tomorrow because if you keep putting off your studies for tomorrow, you'll probably be too late."

"Life is less complicated than it seems. Good habits and hard work are all it takes to succeed and win."

"Don't study to earn, study to learn. What you learn today is what you will become tomorrow."

"Failure is only temporary. The only thing that should be permanent is your will to overcome it."

"Everyone has a talent and so do you. Let it shine out, is all you have to do."

"Try not to become a man of success. Rather become a man of value."

"If you really want to do something, you will find a way. If you don't, you'll find an excuse."