LAB FILE

INTRODUCTION TO C PROGRAMMING

BATCH: 2023-24

BCA HONS. (AI & DS)

Submitted to:

Mr. Rishi Kumar
Assistant Professor (CS/IT)
Graphic Era University

Submitted by:

Sujal Tyagi
(23021377)

INDEX

13	Area and volume of cube	12-13		
14	Area and volume of cuboid	13-14		

15.	WAP to find the largest number using the logical AND operator.	15		
16.	WAP to validate the username and password entered by the user is correct or not using the predefined username and password.	16		
17.	WAP to input the positive number from the user to perform the left shift operator.	17		
18.	WAP to input the positive number from the user to perform the right shift operator.	18		
19.	WAP to perform the pre increment operator and pre decrement operator on two integers and print both original value and updated value.	19		
20.	WAP to perform the post increment operator and post decrement operator on two integers and print both original value and updated value.	20		
21.	WAP for an integer and to check whether the number is divisible by 9 or 7 using OR logical operator.	21		
22.	WAP to identify gender in simple character and print full gender.	22		

23.	Write a C program to print all natural numbers in reverse (from n to 1).	23		
24.	Write a C program to print all alphabets from a to z.	24		
25.	Write a C program to print all natural numbers from 1 to n.	25		
26.	Write a C program to print all even numbers between 1 to 100.	26		
27.	Write a C program to print all odd number between 1 to 100.	27		
28.	Write a C program to find sum of all natural numbers between 1 to n.	28		
29.	Write a C program to find sum of all even numbers between 1 to n.	29		
30.	Write a C program to find sum of all odd numbers between 1 to n.	30		
31.	Write a C program to print multiplication table of any number.	31		
32.	Write a C program to count number of digits in a number.	32		
33.	Write a C program to find first and last digit of a number.	33		
34.	Write a C program to find sum of first and last digit of a number.	34		
35.	Write a C program to swap first and last digits of a number.	35		
36.	Write a C program to calculate sum of digits of a number.	36		

37.	Write a C program to calculate product of digits of a number.	37		
38.	Write a C program to enter a number and print its reverse.	38		
39.	Write a C program to check whether a number is palindrome or not.	39		
40.	Write a C program to find frequency of each digit in a given integer.	40		
41.	Write a C program to enter a number and print it in words.	41		
42.	Write a C program to print all ASCII character with their values	42		

43.	Write a C program to find power of a number using for loop.	43		
44.	Write a C program to find all factors of a number.	44		
45.	Write a C program to calculate factorial of a number.	45		
46.	Write a C program to find HCF (GCD) of two numbers.	46		
47.	Write a C program to find LCM of two numbers.	47		
48.	Write a C Program to check whether a number is prime or not.	48		
49.	Write a C Program to print all prime numbers between 1 to n.	49		
50.	Write a C Program to find sum of all prime numbers between 1 to n.	50		
51.	Write a C Program to find all prime factors of a number.	51		

52.	Write a C Program to check whether a given number is Armstrong number or not.	52		
53.	Write a C Program to print all Armstrong numbers between 1 to n.	53		
54.	Write a C Program to check whether a given number is Perfect number or not.	54		
55.	Write a C Program to print all Perfect numbers between 1 to n.	55		
56.	Write a C Program to check whether a given number is Strong number or not.	56		
57.	Write a C Program to print all Strong numbers between 1 to n.	57		
58.	Write a C Program to print the Fibonacci series upto n terms.	58		
59.	Write a C Program to print 1's complement of a binary number.	59		
60.	Write a C Program to print 2's complement of a binary number.	60		
61.	Write a C Program to convert binary to octal number system.	61		
62.	Write a C Program to convert binary to decimal.	62		
63.	WAP to convert binary to binary number system.	63		
64.	WAP to convert octal to hexadecimal number system.	64		
65.	WAP to convert octal to decimal number system.	65		

66.	WAP to convert octal to hexadecimal number system.	66		
67.	WAP to convert decimal to binary number system.	67		
68.	WAP to convert decimal to octal number system.	68		
69.	WAP to convert decimal to hexadecimal number system.	69		
70.	WAP to convert hexadecimal to binary number system.	70		
71.	WAP to convert hexadecimal to octal number system.	71		
72.	WAP to convert hexadecimal to decimal number system.	72		
73.	Star Pattern 1	73		
74.	Star Pattern 2	74		
75.	Star Pattern 3	75		
76.	Star Pattern 4	76		
77.	Star Pattern 5	77		
78.	Star Pattern 6	78		
79.	Star Pattern 7	79		

80.	Square Pattern 1	80		
81.	Square Pattern 2	81		
82.	Square Pattern 3	82		
83.	Square Pattern 4	83		
84	Square Pattern 5	84		
85.	Square Pattern 6	85		
86.	WAP to find maximum between two numbers.	86		
87.	WAP to find maximum between three numbers.	87		
88.	WAP to check whether a number is positive, negative or zero.	88		
89.	WAP to check whether a number is divisible by 5 and 11 or not.	89		
90.	WAP to check whether a number is even or odd.	90		
91.	WAP to check whether a year is leap year or not.	91		
92.	WAP to check whether it is an alphabet or not.	92		
93.	WAP to check whether an alphabet is vowel or consonant.	93		

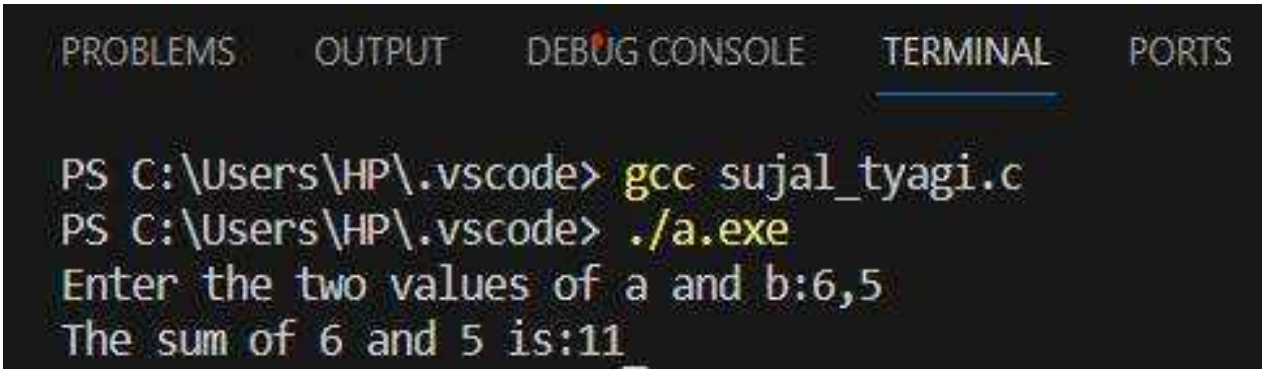
94.	WAP to input any character and check whether it is an alphabet, digit or special character.	94		
95.	WAP to check whether an alphabet is uppercase or lowercase.	95		
96.	WAP to input week and print day.	96		
97.	WAP to input month number and print number of days in that month.	97		
98.	WAP to count notes in given amount.	98		
99.	WAP to input three triangles and check whether it is valid or not.	99		
100.	WAP to input three sides of triangle and check whether it is valid or not.	100		
101.	WAP to check whether triangle is isosceles, equilateral or scalene.	101		
102.	WAP to find the roots of a quadratic equation.	102		
103.	WAP to find the profit or loss.	103		
104.	WAP to give the grade.	104		
105.	WAP to read n number of values in an array and display them in reverse order.	105		
106.	WAP to find the sum of all elements in an array.	106		

107.	WAP to copy the elements of one array to another array.	107		
108.	WAP to count the total number of duplicates in an array.	108		
109.	WAP to find the maximum and minimum elements in an array.	109		
110.	WAP to sort the elements of an array in descending order.	110		
111.	WAP to separate odd and even integers from an array.	111		
112.	WAP to merge two arrays of same size sorted in descending order.	112		
113.	WAP to merge two arrays of same size sorted in ascending order.	113		

PROGRAMS

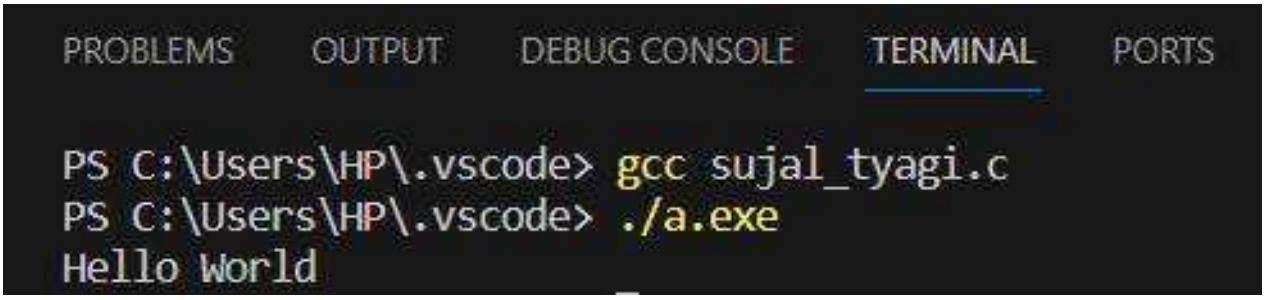
Program 1: WAP to find sum of 2 No.

```
#include<stdio.h>
int main(){ int
a,b; int sum;
printf(“Enter the two numbers separated by comma:”);
scanf(“%d,%d”,&a,&b);
sum=a+b;
printf(“The sum of %d and %d is:%d”,a,b,sum); return
0;
}
```



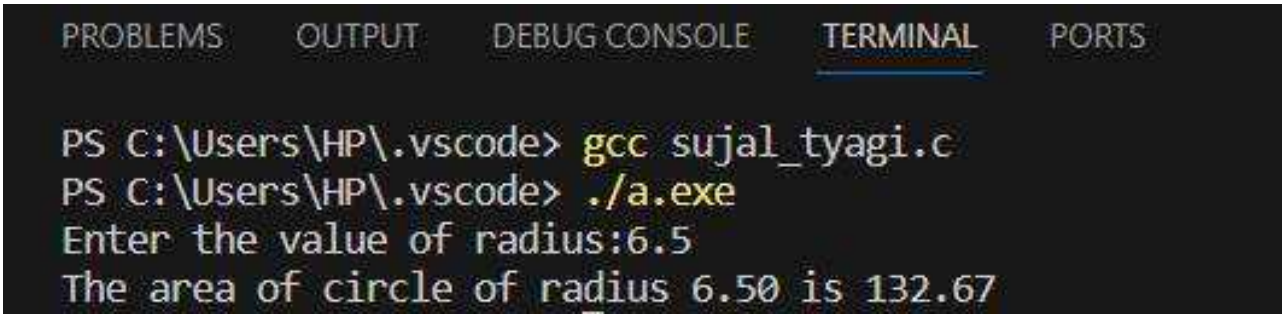
Program 2: WAP to print Hello World

```
#include<stdio.h>

int main(){
printf(“Hello World”);

return 0;
```



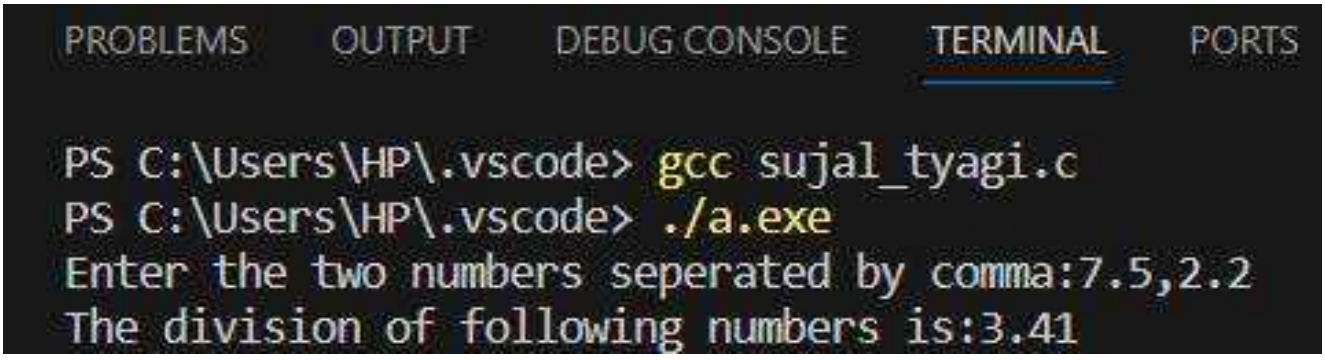
Program 3: WAP to find the Area of Circle.

```
#include<stdio.h>
int main(){ float
r,area; float
pi=3.14;
printf(“Enter the value of radius:”);
scanf(“%f”,&r); area= pi*r*r;
printf(“The area of circle of radius %0.2f is: %0.2f”,r,area);
return 0;
}
```



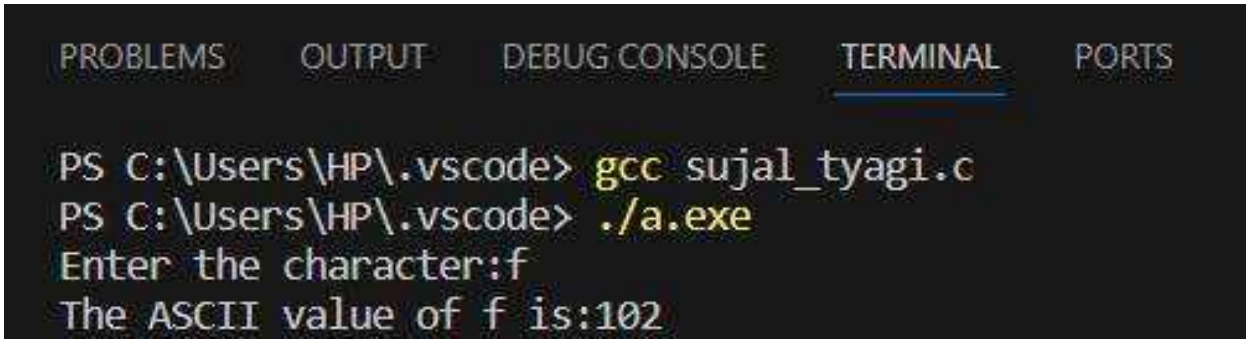
Program 4: WAP to divide two numbers.

```
#include<stdio.h>
int main(){ float
n1,n2; float div;
printf(“Enter the two numbers separated by comma:”);
scanf(“%f,%f”,&n1,&n2);

div=n1/n2;

printf(“The division of following numbers is:%0.2f”,div);

return 0;
}
```



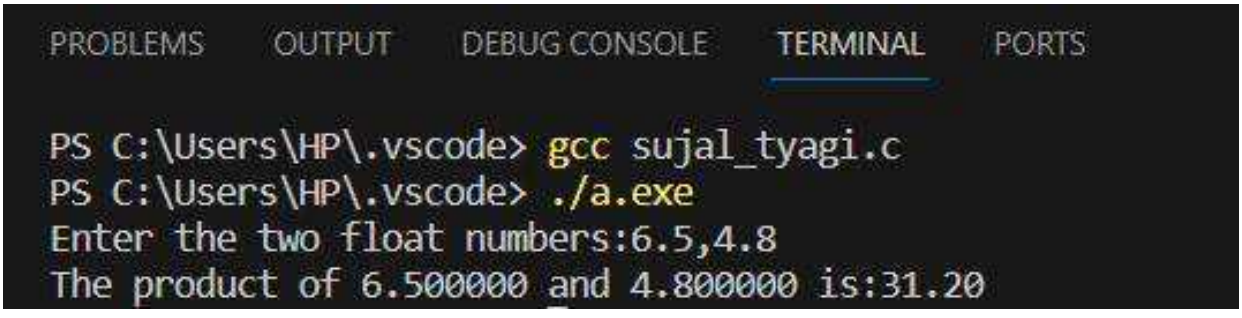
Program 5: WAP to print the ASCII value of a character

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

char c;
printf(“Enter the character:”); scanf(“%c”,c);

printf(“The ASCII value of %c is: %d”,c,c);

return 0;
}
```



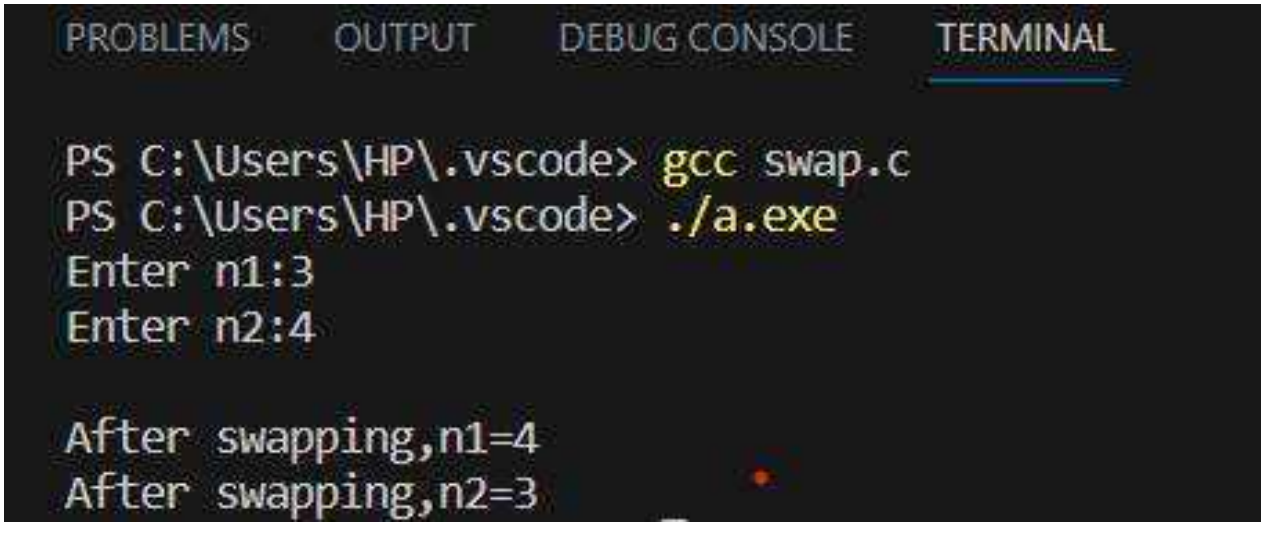
Program 6: WAP to show the product of two floating numbers.

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

float n1,n2;
float prod;

printf(“Enter the two float numbers:”);
scanf(“%f,%f”,&n1,&n2);

prod=n1*n2;

printf(“The product of %f and %f is :%0.2f\n”,n1,n2,prod);

return 0;
}
```



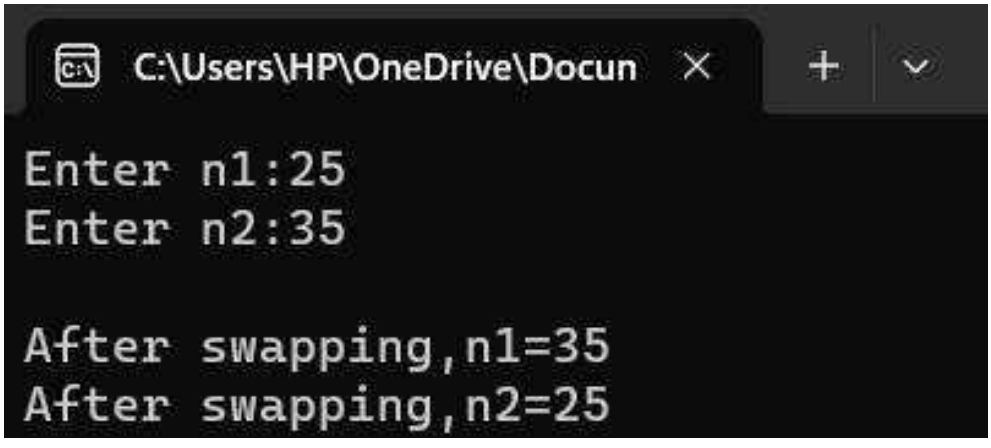
Program 7: WAP to swap two variable numbers using third variable.

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

int n1,n2,temp;
printf("Enter n1:");
scanf("%d",&n1);

printf("Enter n2:);
scanf("%d",&n2);

temp=n1; n1=n2;
n2=temp;

printf("\nAfter swapping,n1=%d\n",n1); printf("\nAfter
swapping,n2=%d\n,n2);

return 0;
}
```



Program 8: WAP to swap the two variable numbers without using third variable

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

int n1,n2;

printf("Enter n1:"); scanf("%d",&n1);

printf("Enter n2:"); scanf("%d",&n2);

n2=n1-n2;
n1=n1+n2;

printf("\nAfter swapping, n1=%d\n",n1);
printf("After swapping ,n2=%d\n",n2); return
0;
}
```

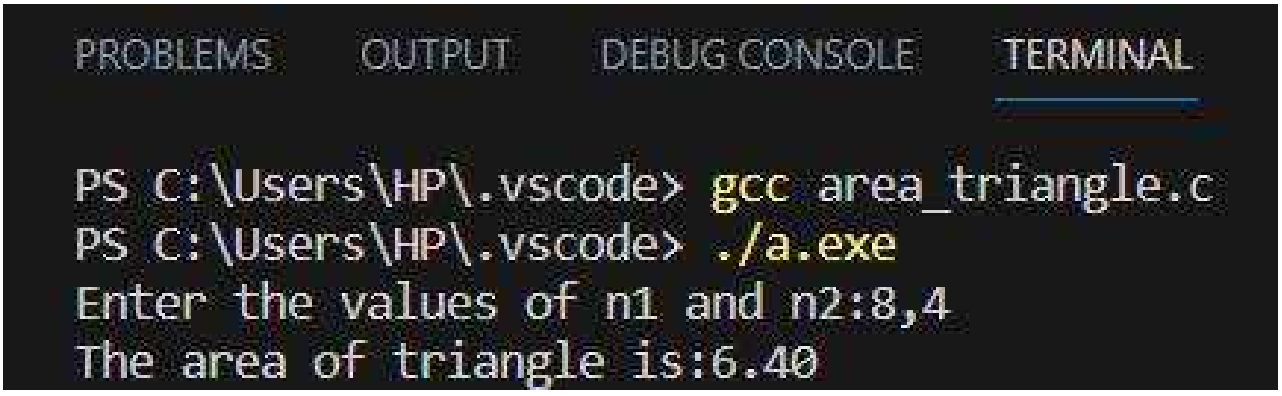


Program 9: WAP to swap three numbers without using any variable.

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

int n1,n2,n3; printf(“Enter n1:”);
scanf(“%d”,&n1); printf(“Enter n2:”);
scanf(“%d”,&n2); printf(“Enter n3:”);
scanf(“%d”,&n3); n2=n1-(n2+n3);
n3=n1-(n2+n3); n1=n1-(n2+n3);
printf(“\nAfter swapping,n1=%d\n”,n1);
printf(“After swapping,n2=%d\n”n2);
printf(“After swapping,n3=%d”,n3);}
```

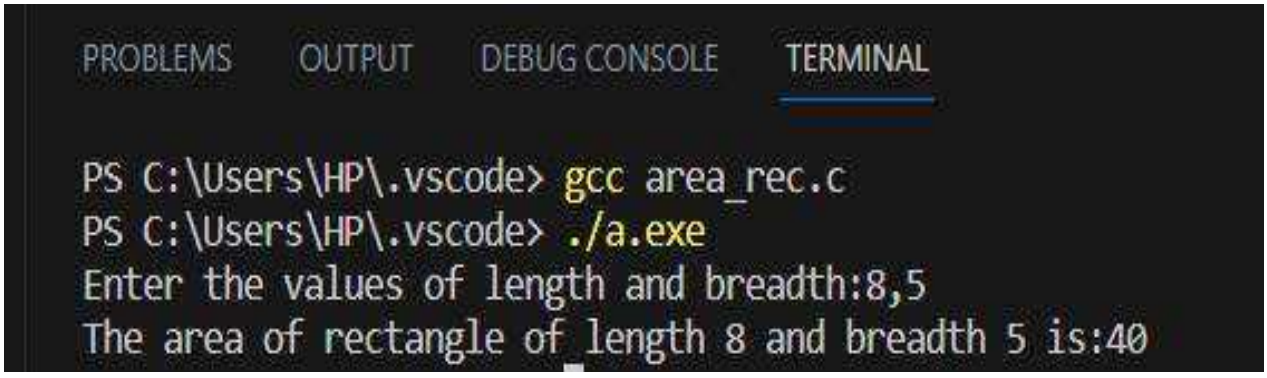
Program 10: WAP to find the Area of Triangle.

```
#include<stdio.h>
int   main(){   float
b,h; double area;

printf(“Enter the base and height:”); scanf(“%f,%f”,&b,&h);

area=0.2*b*h;

printf(“The area of right angled triangle is: %f\n”,b,h,area);
return 0;
}
```



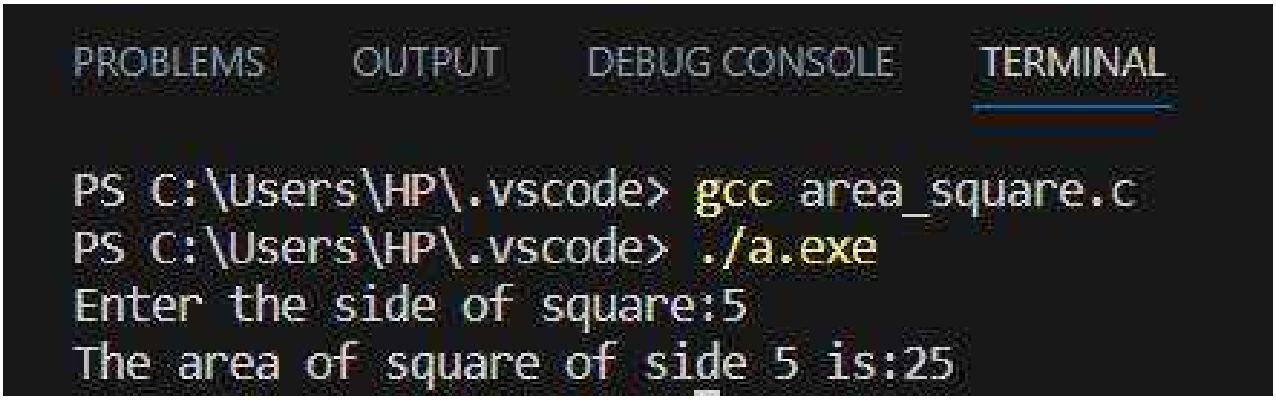
Program 11: WAP to find the area of rectangle.

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

int l,b; int
area;

printf(“Enter the values of length and breadth:”);
scanf(“%d,%d”,&l,&b);

area=l*b;

printf(“The area of rectangle of length %d and breadth %d
is:%d\n”l,b,area);

return 0;
}
```



Program 12: WAP to find the area of square.

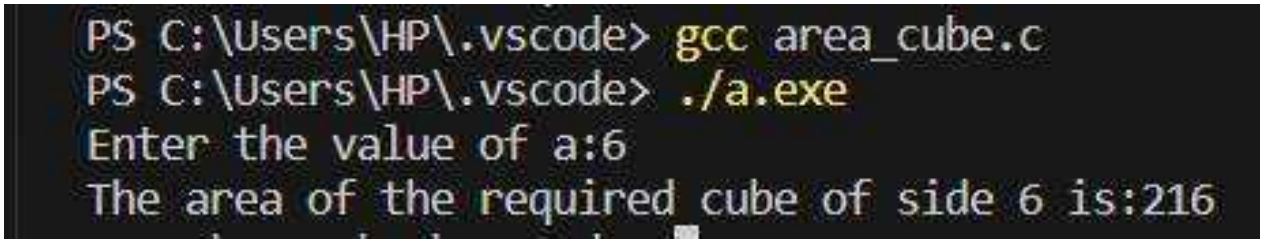
```
#include<stdio.h> int
main(){

int a; int
area;

printf("Enter the side of square:"); scanf("%d",
&a);

area=a*a;

printf("The area of square of side %d is:%d", a, area);
return 0;
}
```



Program 13: WAP to find the area and volume of cube.

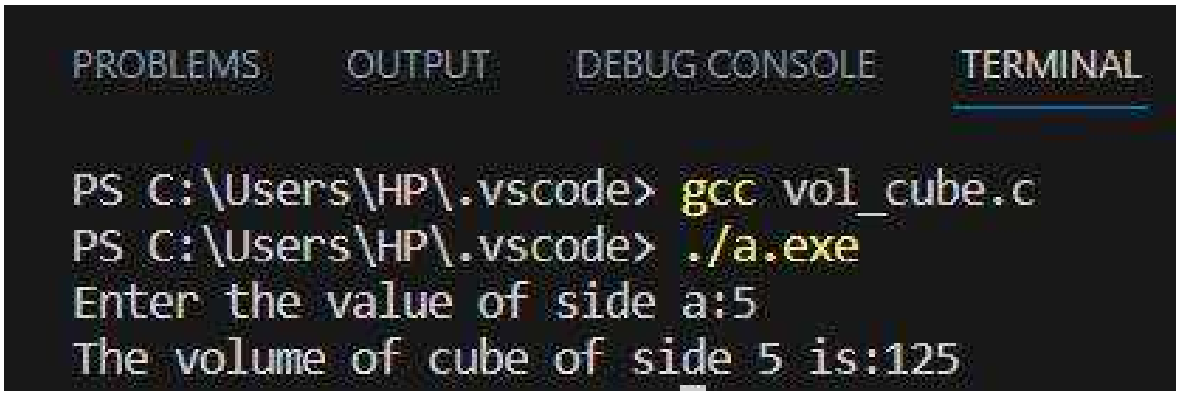
Area of cube:

```
#include<stdio.h>
//area of cube
int main(){
int a; int area; printf("Enter the value of a:"); scanf("%d",
&a); area=6*a*a; printf("The area of the required cube of
side %d is:%d", a, area); return 0;
}
```



Volume of cube:

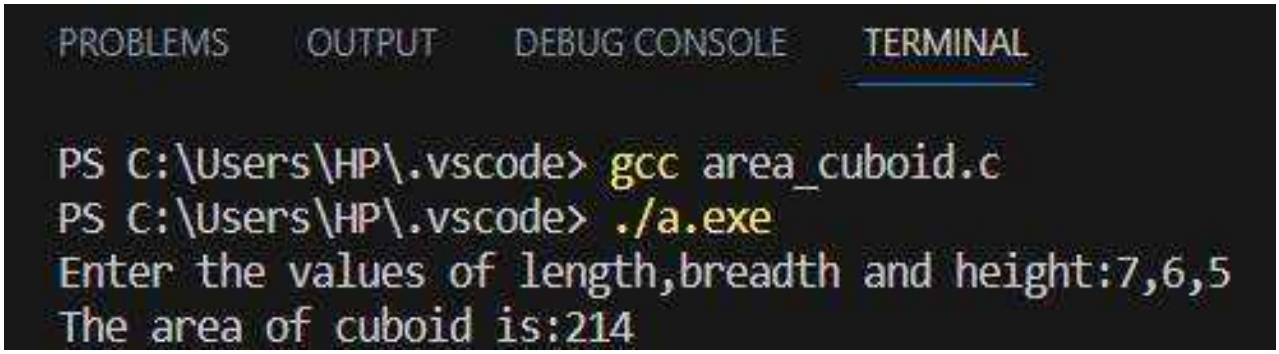
```
#include<stdio.h> int main(){ int a; int vol; printf("Enter
the value of side a:"); scanf("%d" , &a); vol=a*a*a;
printf("The volume of cube of side %d is:%d\n" , a ,vol);
return 0;
}
```



Program 14: WAP to find the Area and Volume of cuboid.

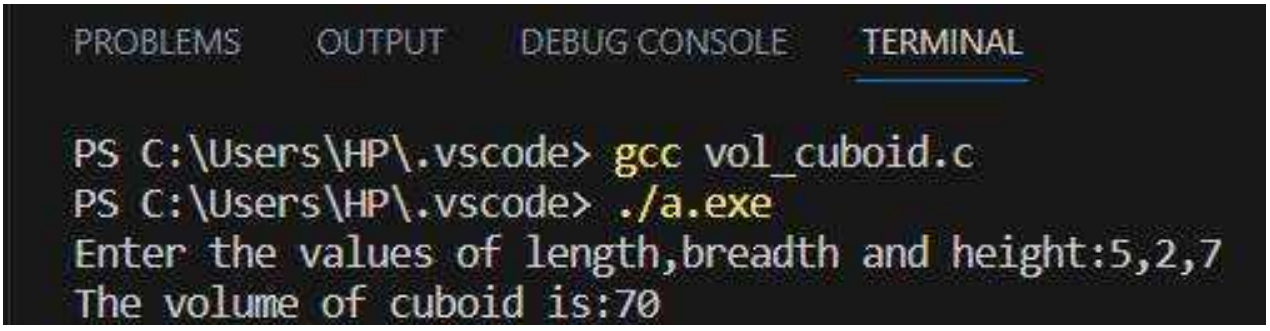
Area of Cuboid:

```
#include<stdio.h> int
main(){
   int l , b , h;     int area;     printf("Enter the values of
length , breadth and height:");     scanf("%d ,%d ,%d ",&l,
&b , &h);     area=2*(l*b +b* h + h*l);     printf("The area of
cuboid is:%d\n" , area);     return 0;
}
```



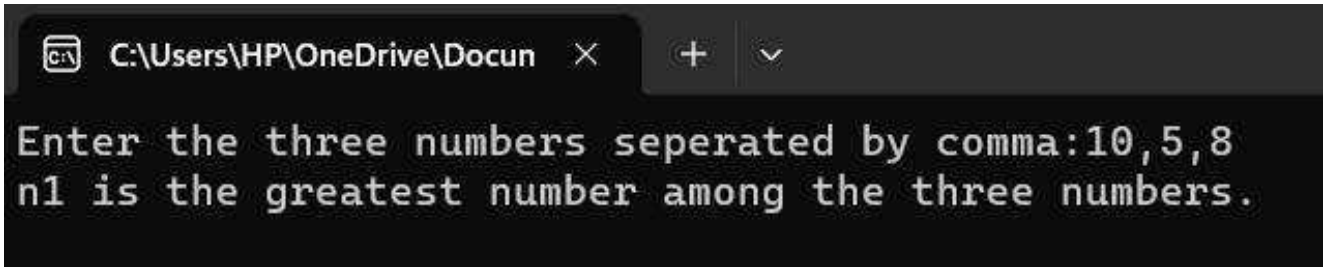
Volume of Cuboid:

```
#include<stdio.h>
int main(){ int l,b,h;
int vol;
printf("Enter the values of length,breadth and height:");
scanf("%d,%d,%d",&l,&b,&h);
vol=l*b*h;
printf("The volume of cuboid is:%d\n",vol);
return 0;
}
```



Program 15: WAP to find the greatest no. using && operator.

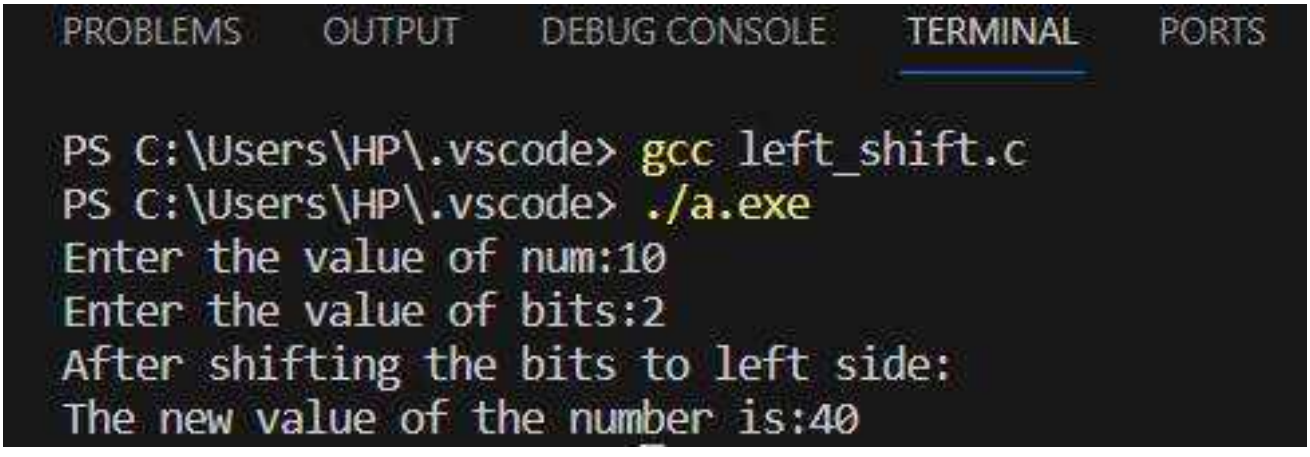
```
#include<stdio.h>
 int main(){
 int n1,n2,n3;
printf("Enter the three numbers seperated by comma:");
scanf("%d,%d,%d",&n1,&n2,&n3);
if(n1>n2 && n1>n3){
printf("n1 is the greatest number among the three
numbers.\n");
}
else if(n2>n1 && n2>n3){
printf("n2 is the greatest number among the three
numbers.\n);
}
else if(n3>n1 && n3>n2){
printf("n3 is the greatest number among the three
numbers.\n");
}
else{ printf("Not valid
task.\n");
}
}
```



Program 16: WAP to validate the username and password entered by the user is correct or not.

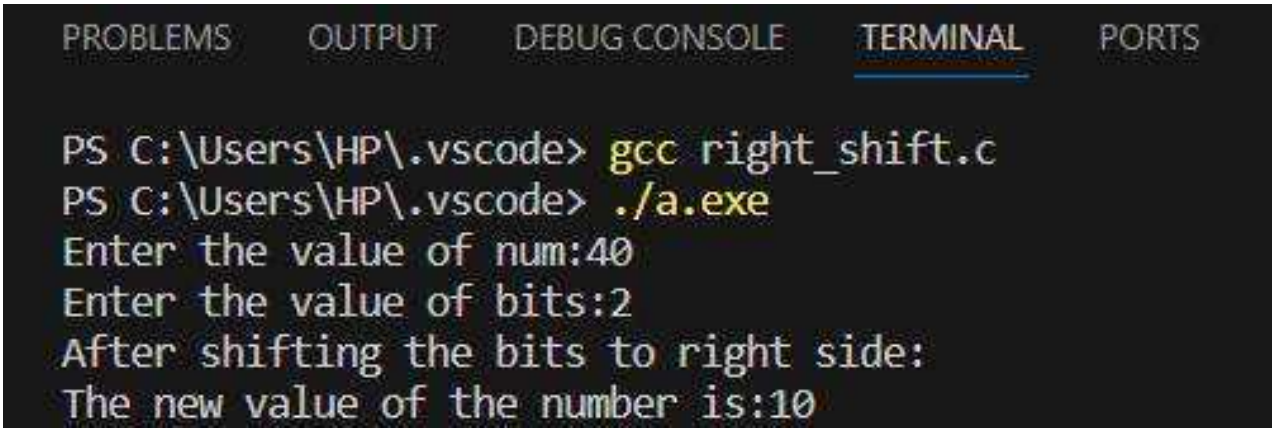
Program 17: WAP to input the positive number from user to perform left shift operator.

```
#include<stdio.h>
 int main(){
int num,bit;
 printf("Enter the value of no.:);
scanf("%d",&num);
printf("Enter the value of bits:");
scanf("%d",&bit);
num=(num<<bit);
printf("After shifting the bits to left side:\n");
printf("The new value of the number is:%d",num);
}
```



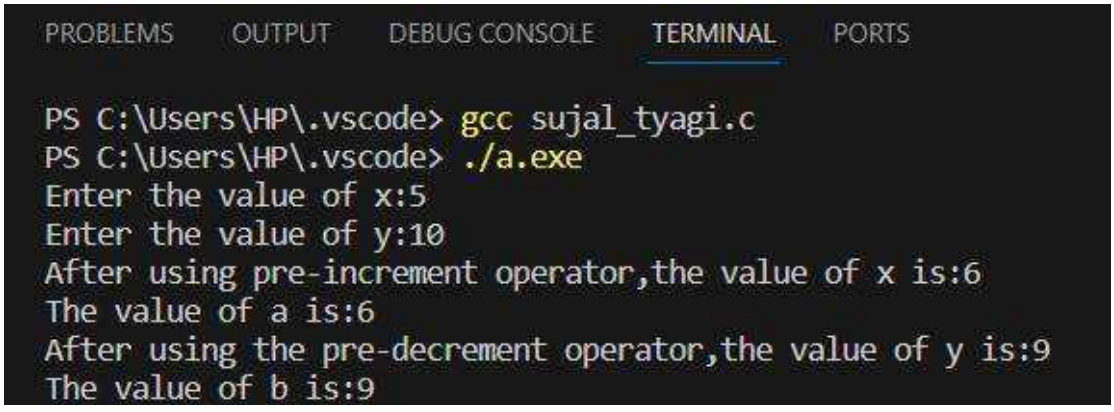
Program 18: WAP to input the positive number from user to perform right shift operator.

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

int num,bit;
printf("Enter the value of num:");
scanf("%d",&num);
printf("Enter the value of bits:");
scanf("%d",&bit);

num=(num>>bit);
printf("After shifting the bits to right side:\n");
printf("The new value of the number is:%d",num);
}
```



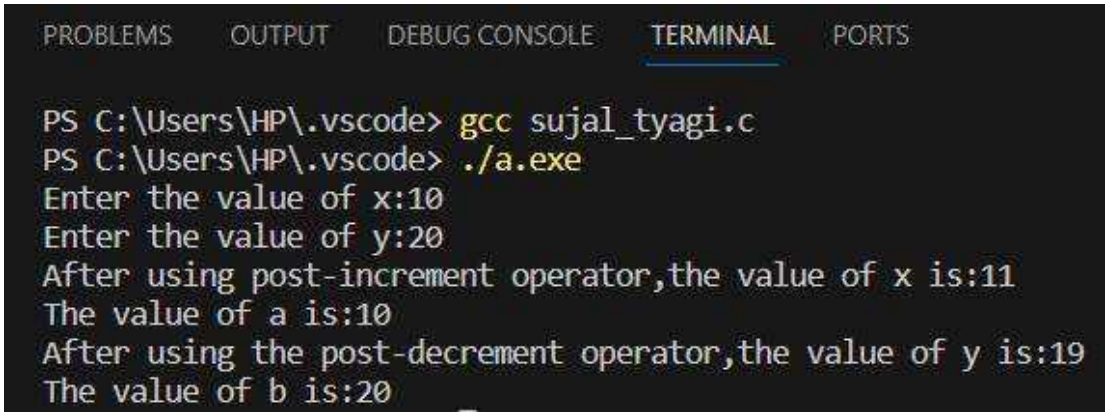
Program 19: WAP to perform pre increment and pre decrement operators on two values.

```
#include <stdio.h>
int main (){
int x,y,a,b;
printf ("Enter the value of x:");
scanf ("%d", &x);
printf ("Enter the value of y:");
scanf ("%d", &y);
a= ++x;
printf ("After using pre-increment operator,the value of x
is:%d\n",x);
printf ("The value of a is:%d\n",a);
b= --y;
printf ("After using the pre-decrement operator,the value of y
is:%d\n",y);
printf ("The value of b is:%d\n",b);
return 0;
}
```



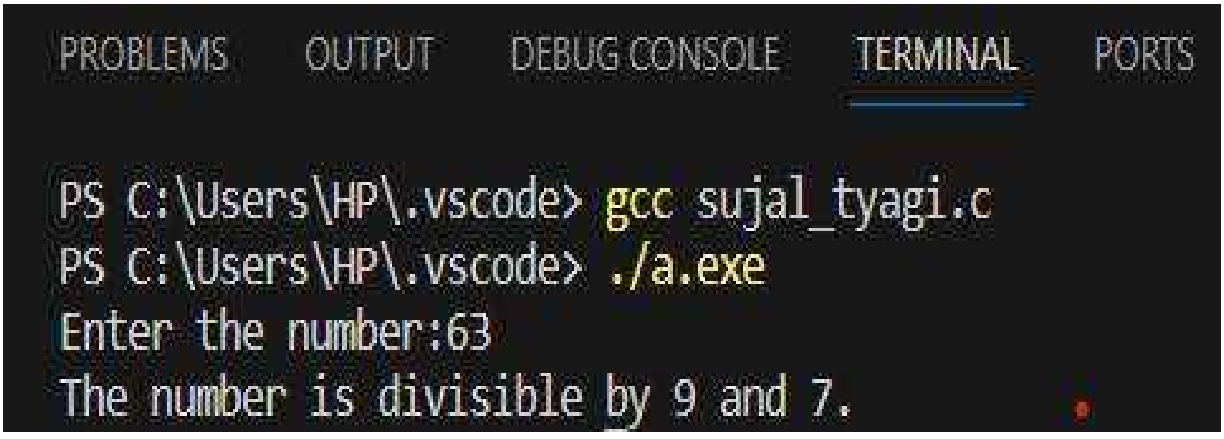
Program 20: WAP to perform post increment and post decrement operators on two values.

```
#include <stdio.h>
int main (){
int x,y,a,b;
printf ("Enter the value of x:");
scanf ("%d", &x);
printf ("Enter the value of y:");
scanf ("%d", &y);
a= x++;
printf ("After using post-increment operator,the value of x is:%d\n",x);
printf ("The value of a is:%d\n",a);
b= y--;
printf ("After using the post-decrement operator,the value of y is:%d\n",y);
printf ("The value of b is:%d\n",b);
return 0;
}
```



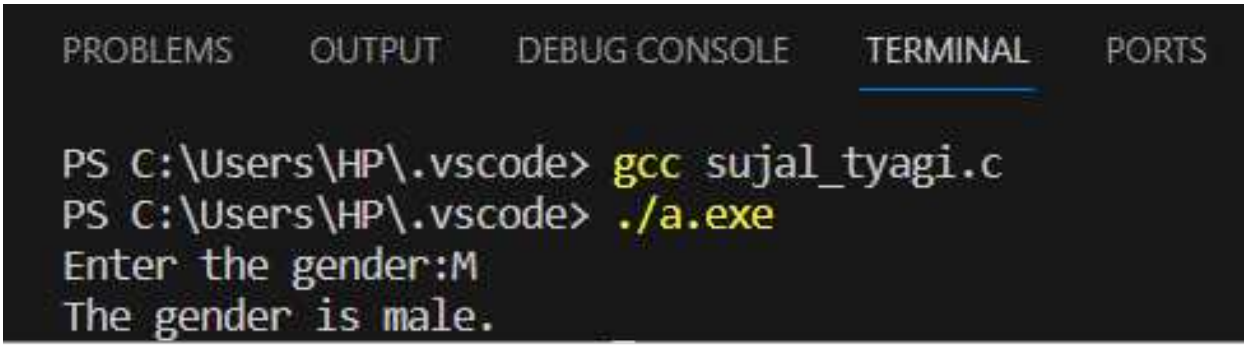
Program 21: WAP for an integer to check whether it is divisible by 9 or 7 using || operator.

```
#include <stdio.h>
int main (){
int num;
printf("Enter the number:");
scanf("%d",&num);
if(num%9==0 || num%7==0){
printf("The number is divisible by 9 and 7.\n");
}
return 0;
}
```



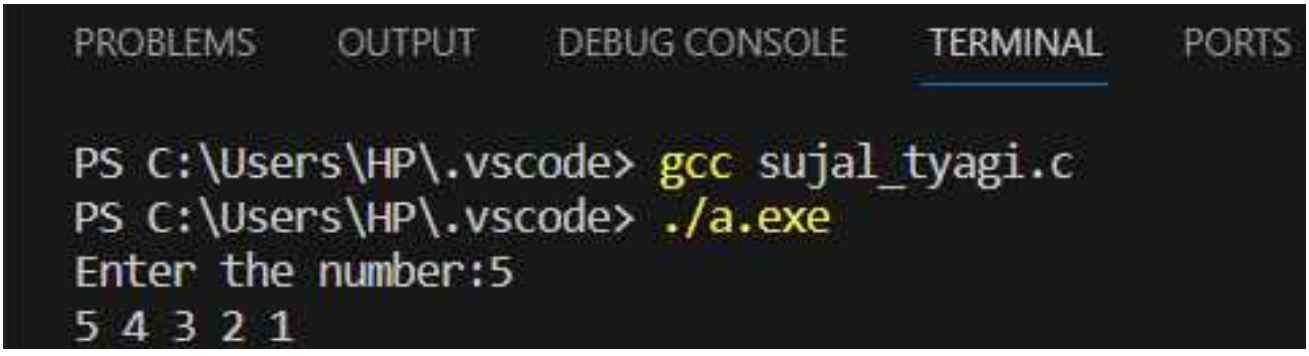
Program 22: WAP to identify gender in single character and print full gender.

```
#include<stdio.h>
int main(){
char gen;
printf("Enter the gender:");
scanf("%c",&gen);
if(gen='M' || gen=='m'){
      printf("The gender is male.\n");
   }
else{
printf("The gender is female.\n");
   }
}
```



Program 23: WAP to print all the numbers from n to 1.

```
#include <stdio.h>
void main()
{
   int n, i;
   printf("Enter the number:");
   scanf("%d", &n);
   for (i = n; i >= 1; i--)
   {
      printf("%d ", i);
   }
}
```

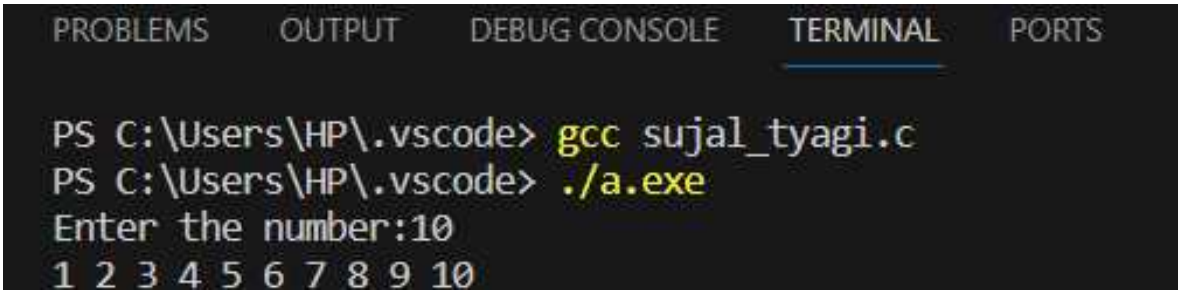


Program 24: WAP to print all alphabets from ‘a’ to ‘z’.

```
#include <stdio.h>
int main()
{
   char ch;
   for (ch = 'a'; ch <= 'z'; ch++)
   {
      printf(" %c ",ch);
   }
}
```

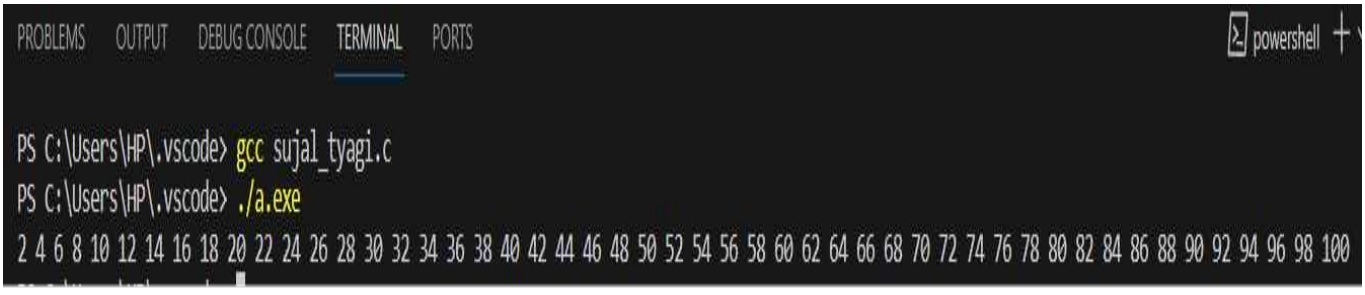
Program 25: WAP to print all the numbers from 1 to n.

```
#include <stdio.h>
void main()
{
   int n, i;
   printf("Enter the number:");
   scanf("%d", &n);
   for (i = 1; i <=n; i++)
   {
      printf("%d ", i);
   }
}
```



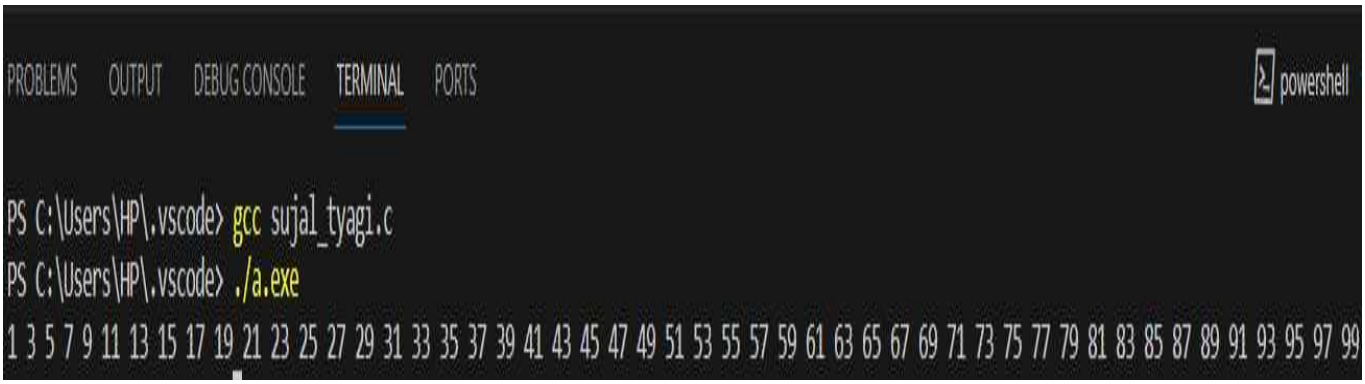
Program 26: WAP to print all the even numbers from 1 to 100.

```
#include <stdio.h>
void main()
{
   int i;
   for(i=1;i<=100;i++){
      if(i%2==0){
         printf("%d ",i);
      }
   }
}
```



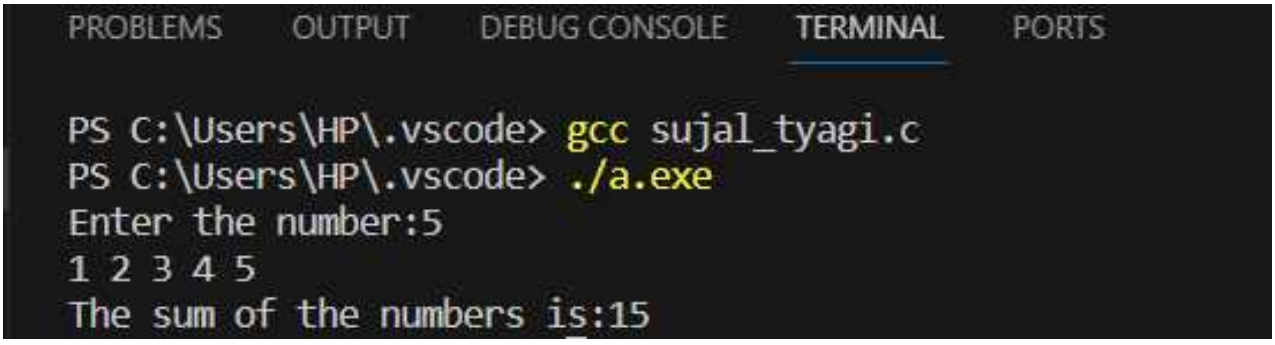
Program 27: WAP to print all the odd numbers from 1 to 100.

```
#include <stdio.h>
void main()
{
   int i;
   for(i=1;i<=100;i++){
      if(i%2!=0){
         printf("%d ",i);
      }
   }
}
```



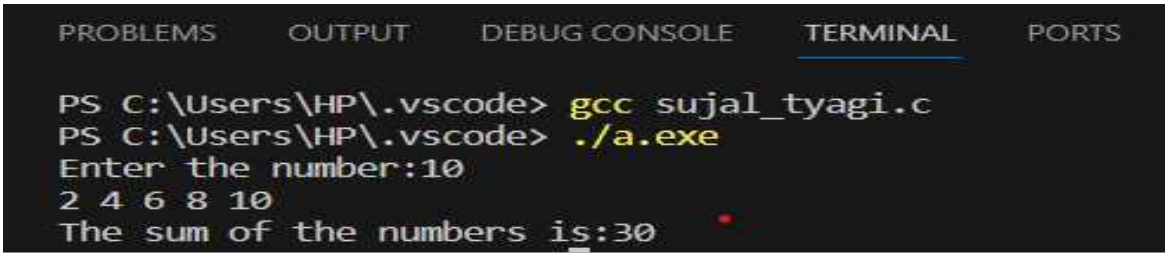
Program 27: WAP to print the sum of numbers between 1 to n.

```
#include <stdio.h>
void main()
{
   int i, n, sum = 0;
   printf("Enter the number:");
   scanf("%d", &n);
   for (i = 1; i <= n; i++)
   {
      printf("%d ", i);
      sum += i;
   }
   printf("\nThe sum of the numbers is:%d\n", sum);
}
```



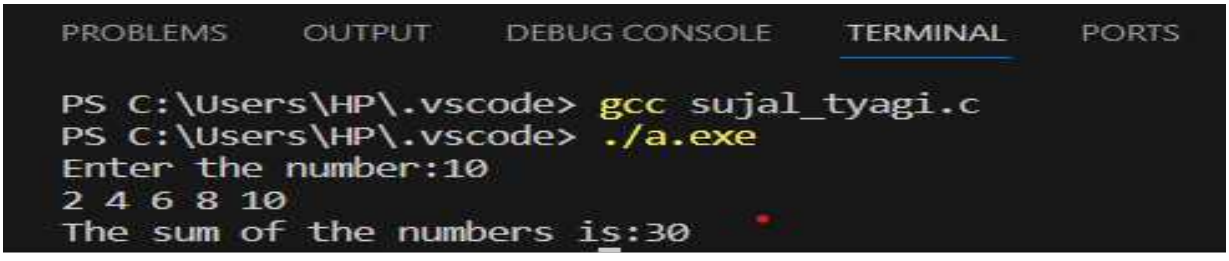
Program 29: WAP to print the sum of all even numbers between 1 to n.

```
#include <stdio.h>
void main()
{
   int i, n, sum = 0;
   printf("Enter the number:");
   scanf("%d", &n);
   for (i = 1; i <= n; i++)
   {
      if (i % 2 == 0)
      {
         printf("%d ", i);
         sum += i;
      }
   }
   printf("\nThe sum of the numbers is:%d\n", sum);
}
```



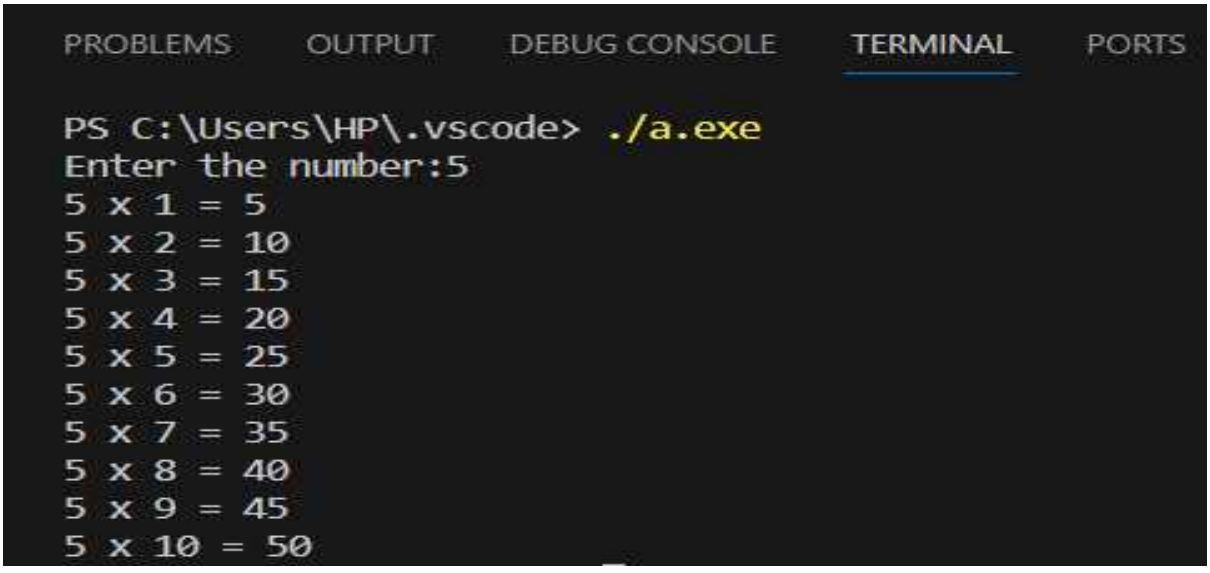
Program 30: WAP to print the sum of all odd numbers between 1 to n.

```
#include <stdio.h>
void main()
{
   int i, n, sum = 0;
   printf("Enter the number:");
   scanf("%d", &n);
   for (i = 1; i <= n; i++)
   {
      if (i % 2 != 0)
      {
         printf("%d ", i);
         sum += i;
      }
   }
   printf("\nThe sum of the numbers is:%d\n", sum);
}
```



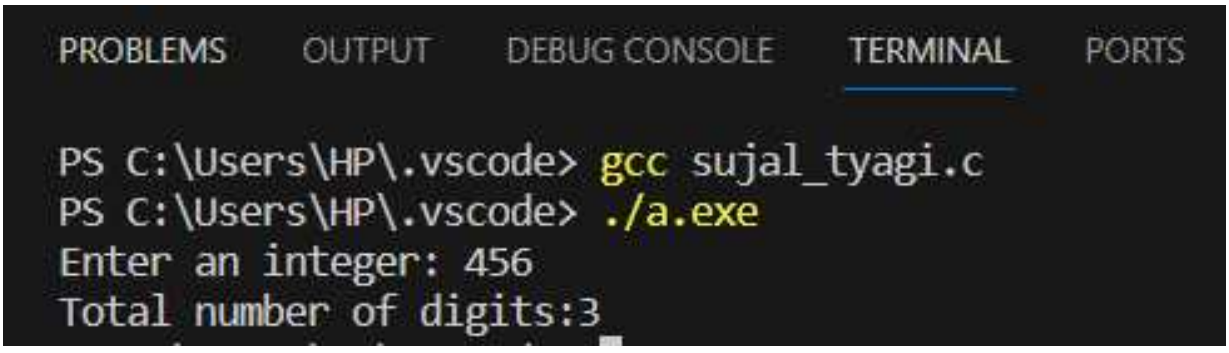
Program 31: WAP to print the multiplication table of any number.

```
#include <stdio.h>
void main()
{
   int i, n, prod = 1;
   printf("Enter the number:");
   scanf("%d", &n);
   for (i = 1; i <= 10; i++)
   {
      prod = n * i;
      printf("%d x %d = %d\n", n, i, prod);
   }
}
```



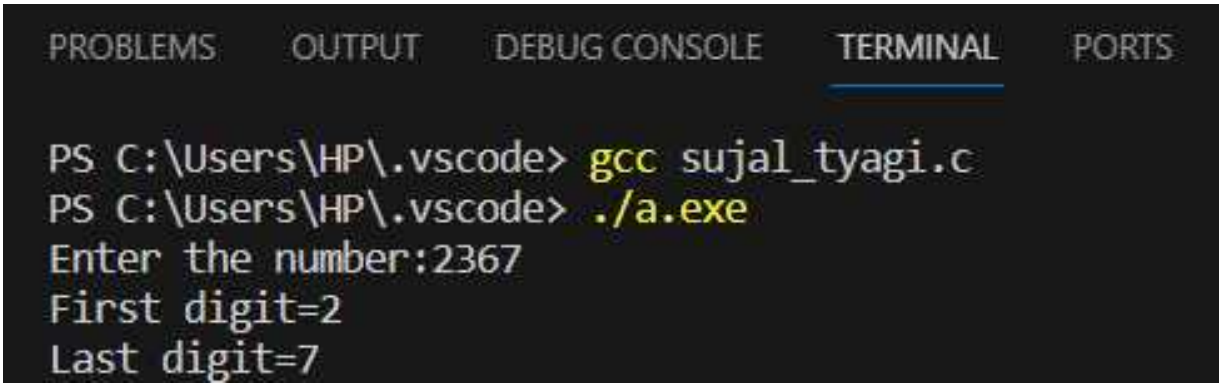
Program 32: WAP to count the digits of a number.

```
#include <stdio.h>
void main()
{
   int i,n;
   int count = 0;
  printf("Enter an integer: ");
  scanf("%d", &n);

  while(n>0){
    n=n/10;
    count++;
  }
  printf("Total number of digits:%d\n", count);
}
```



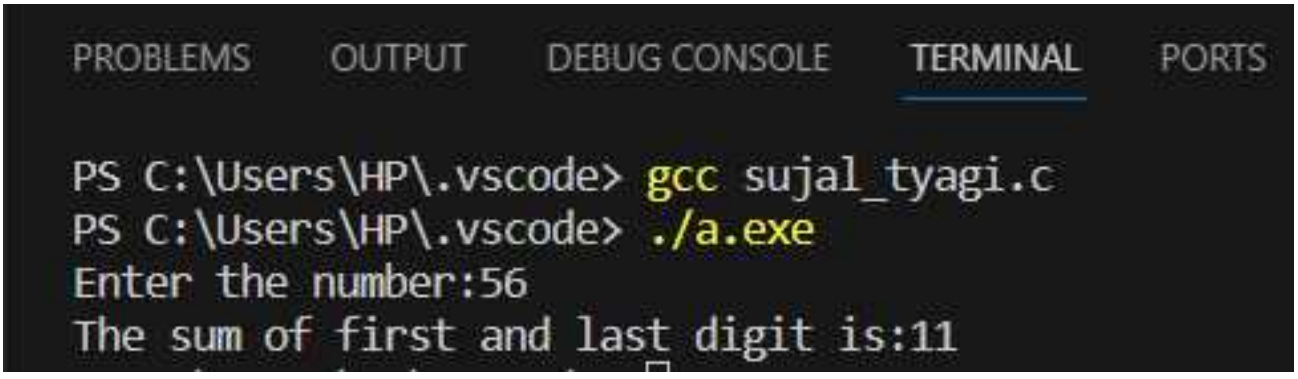
Program 33: WAP to print the first and last digits of a number.

```
#include <stdio.h>
void main()
{
   int n, sum = 0, frstdig, lstdig;
   printf("Enter the number:");
   scanf("%d", &n);

   lstdig = n % 10;

   while (n >= 10)
   {
      n = n / 10;
   }
   frstdig = n;
   printf("First digit=%d\nLast digit=%d\n", frstdig, lstdig);
}
```



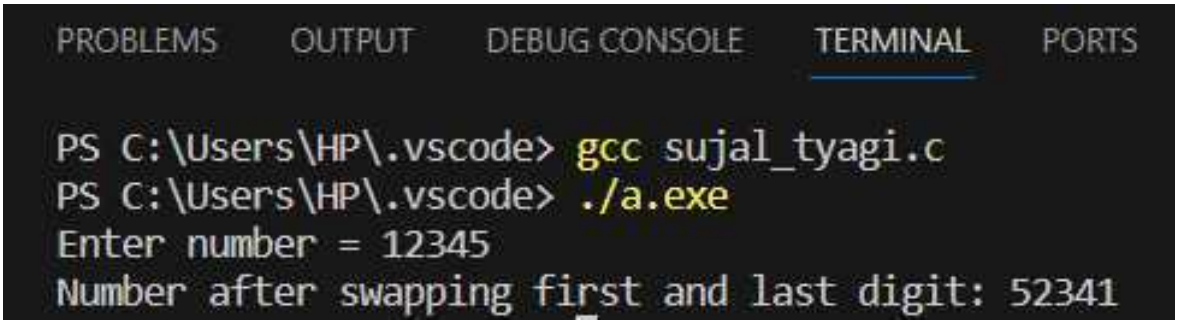
Program 34: WAP to print the sum of first and last digit of a number.

```
#include <stdio.h>
void main()
{
   int n, sum = 0, frstdig, lstdig;
   printf("Enter the number:");
   scanf("%d", &n);

   lstdig = n % 10;

   while (n >= 10)
   {
      n = n / 10;
   }
   frstdig = n;
   sum = frstdig + lstdig;
   printf("The sum of first and last digit is:%d\n", sum);
}
```



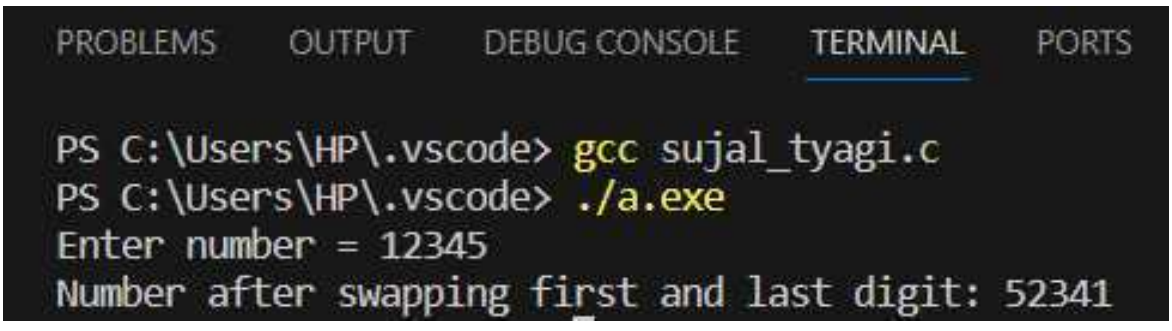
Program 35: WAP to swap first and last digit of a number.

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

int main()
{
   int n, firstDigit, lastDigit, digits, swappedNum;
   printf("Enter number = ");
   scanf("%d", &n);

   lastDigit = n % 10;

   digits = (int)log10(n);

   firstDigit = (int)(n / pow(10, digits));
   swappedNum = lastDigit;
   swappedNum *= (int)round(pow(10, digits));
   swappedNum += n % ((int)round(pow(10, digits)));
   swappedNum -= lastDigit;
   swappedNum += firstDigit;
   printf("Number after swapping first and last digit: %d", swappedNum);

   return 0;
}
```



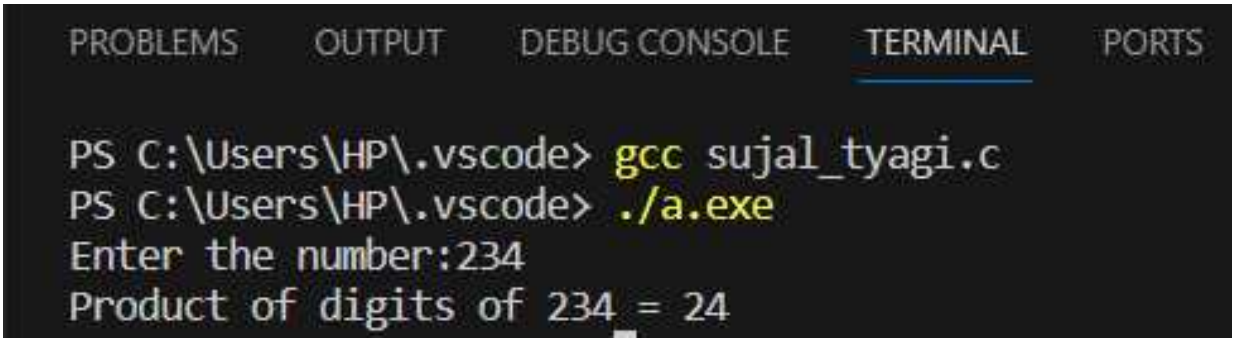
Program 36: WAP to calculate sum of digits of a number.

```
#include <stdio.h>
int main()
{
   int n, t, sum = 0, remainder;

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

   t = n;

   while (t != 0)
   {
      remainder = t % 10;
      sum = sum + remainder;
      t = t / 10;
   }

   printf("Sum of digits of %d = %d\n", n, sum);

   return 0;
}
```



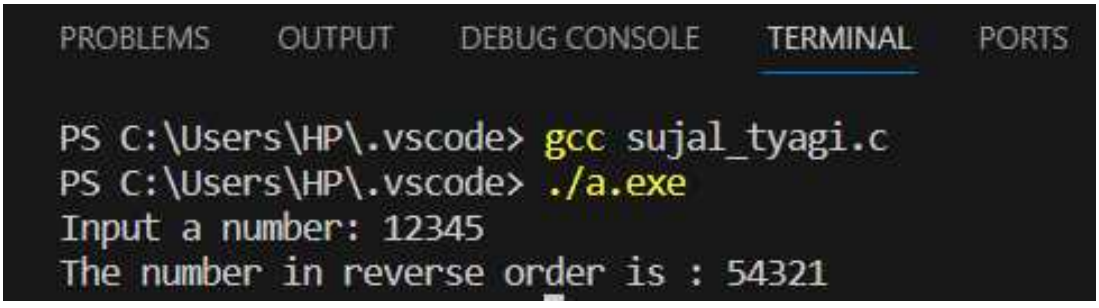
Program 37: WAP to calculate product of digits of a number.

```
#include <stdio.h>
int main()
{
   int n, t, product = 1, remainder;

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

   t = n;

   while (t != 0)
   {
      remainder = t % 10;
      product = product *  remainder;
      t = t / 10;
   }

   printf("Product of digits of %d = %d\n", n, product);

   return 0;
}
```



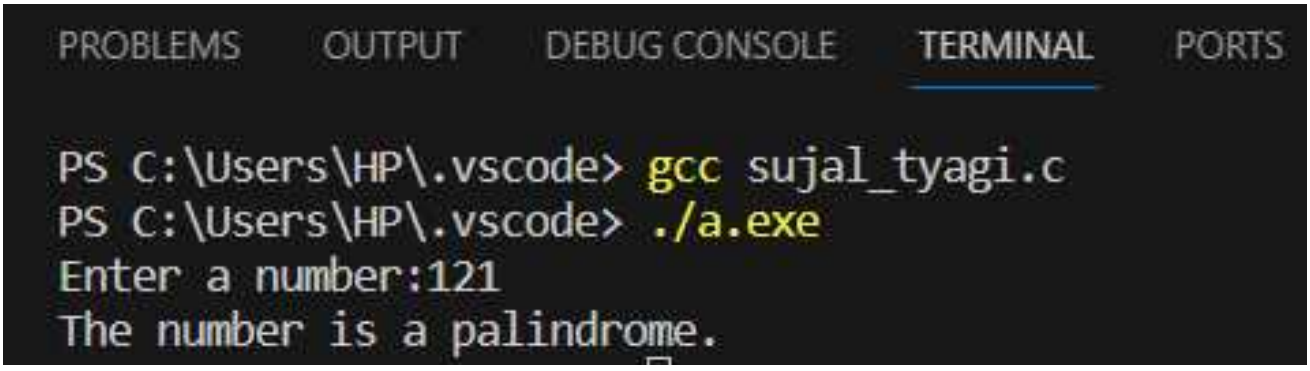
Program 38: WAP to enter a number and print its reverse.

```
#include <stdio.h>

void main(){
   int num,r,sum=0,t;

   printf("Input a number: ");
   scanf("%d",&num);

   for(t=num;num!=0;num=num/10){
       r=num % 10;
       sum=sum*10+r;
   }
printf("The number in reverse order is : %d \n",sum);
}
```



Program 39: WAP to check whether a number is palindrome or not.

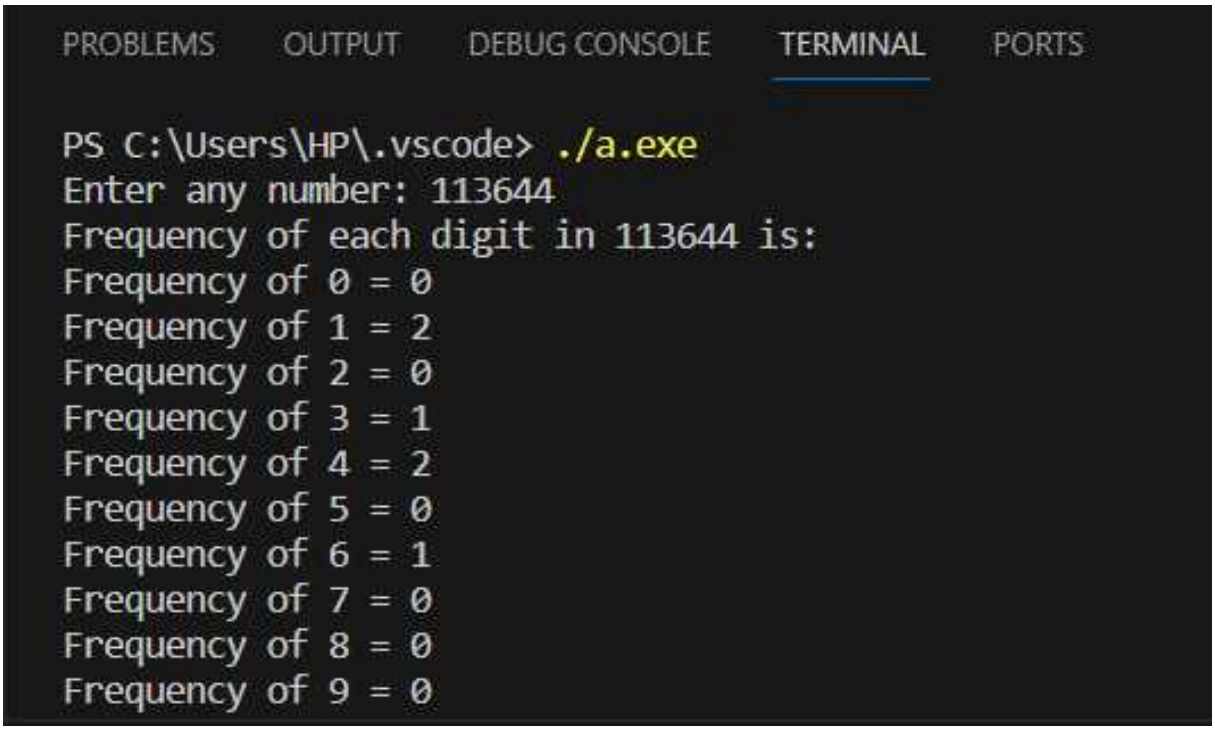
```
#include <stdio.h>

void main(){
   int c,n,s=0,r;
   printf("Enter a number:");
   scanf("%d",&n);
   c=n;
   while(n>0){
    r=n%10;
    s=r+(s*10);
    n=n/10;
   }if(c==s){
    printf("The number is a palindrome.\n");
   }else{
    printf("The number is not a palindrome.\n");
   }
}
```



Program 40: WAP to find the frequency of digits in a number.

```
#include <stdio.h>
#define BASE 10

int main()
{
   long num, n;

   int i, lastDigit;

   int freq[BASE];

   printf("Enter any number: ");
   scanf("%lld", &num);

   for(i=0; i<BASE; i++)
   {
      freq[i] = 0;
   }
   n = num;
   while(n != 0)
   {
      lastDigit = n % 10;

      n /= 10;

      freq[lastDigit]++;
   }
```

```
    printf("Frequency of each digit in %lld is: \n", num);

    for(i=0; i<BASE; i++)
    {
        printf("Frequency of %d = %d\n", i, freq[i]);
    }

    return 0;
}
```



Program 41: WAP to find the frequency of digits in a number.

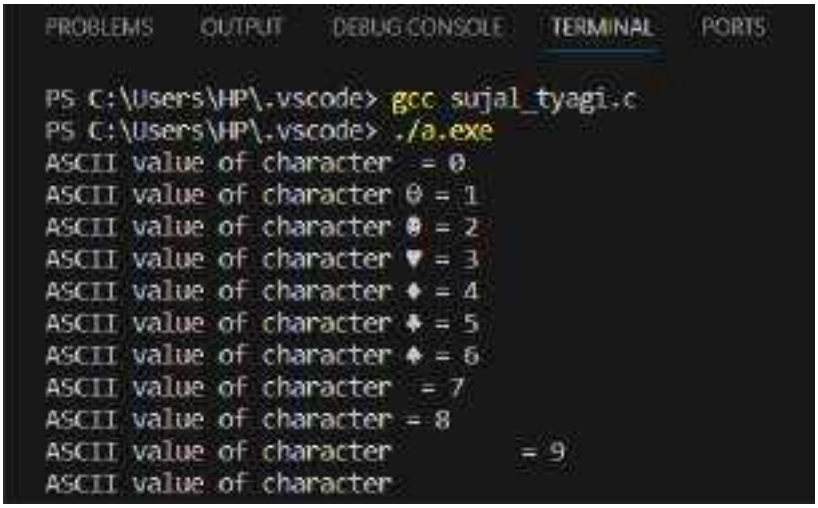
```
#include <stdio.h>
#include <math.h>
int main()
{
   int data, num = 0, digits;
   printf("Enter any number to print in words: ");
   scanf("%d", &data);
   digits = (int) log10(data);
   while(data != 0)
   {
      num = (num * 10) + (data % 10);
      data /= 10;
   }
   digits =  digits - ((int) log10(num));

   while(num != 0)
   {
      switch(num % 10)
      {
      case 0:
         printf("Zero ");
         break;
      case 1:
         printf("One ");
         break;
      case 2:
         printf("Two ");
         break;
```

```
        case 3:
            printf("Three ");
            break;
        case 4:
            printf("Four ");
            break;
        case 5:
            printf("Five ");
            break;
        case 6:
            printf("Six ");
            break;
        case 7:
            printf("Seven ");
            break;
        case 8:
            printf("Eight ");
            break;
        case 9:
            printf("Nine ");
            break;
        }
        num /= 10;
    }
    // Print all trailing 0
    while(digits)
    {
        printf("Zero ");
        digits--;
    }
    return 0;
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\HP\.vscode> gcc sujal_tyagi.c
PS C:\Users\HP\.vscode> ./a.exe
Enter any number to print in words: 6737
Six Seven Three Seven
```

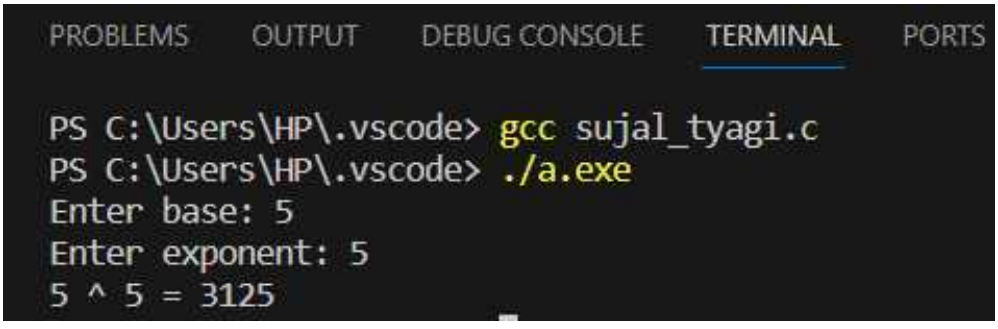
Program 42: WAP to print all the ASCII characters with their values.

```
#include<stdio.h>

int main(){
     int i;

   for(i=0; i<=255; i++)
   {
      printf("ASCII value of character %c = %d\n", i, i);
   }
   return 0;
}
```



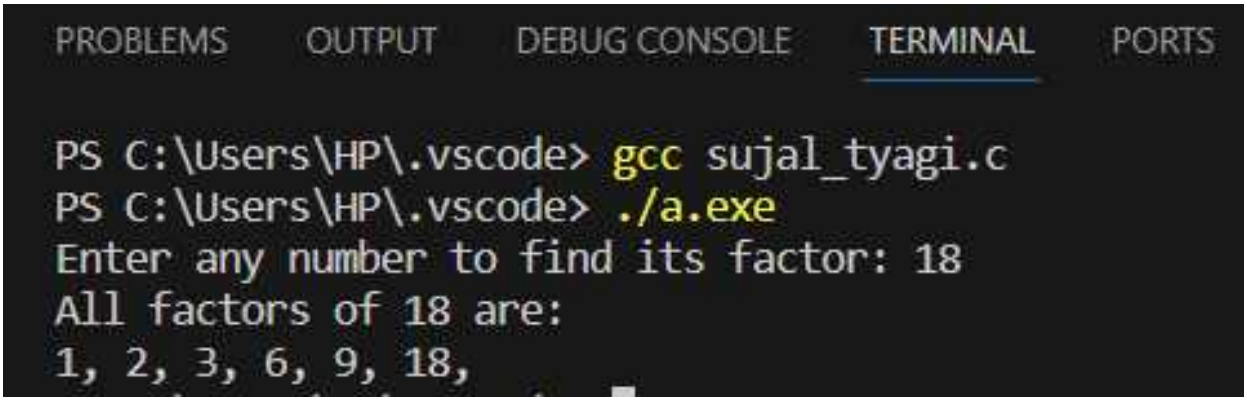
Program 43: WAP to find the power of a number using for loop.

```
#include <stdio.h>

int main()
{
   int base, exponent;
   long power = 1;
   int i;

   printf("Enter base: ");
   scanf("%d", &base);
   printf("Enter exponent: ");
   scanf("%d", &exponent);

   for(i=1; i<=exponent; i++)
   {
      power = power * base;
   }

   printf("%d ^ %d = %lld", base, exponent, power);

   return 0;
}
```



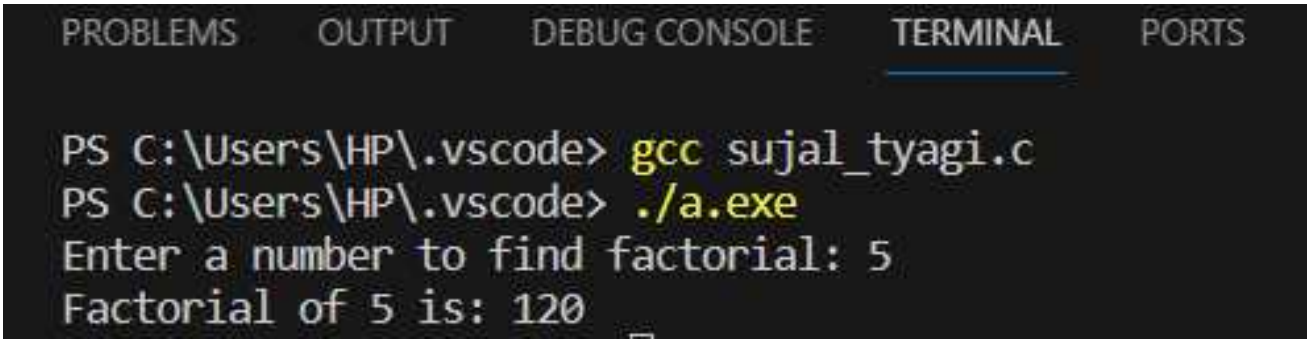
Program 44: WAP to find all factors of a number.

```
#include <stdio.h>

int main()
{
  int i, num;

   printf("Enter any number to find its factor: ");
   scanf("%d", &num);

   printf("All factors of %d are: \n", num);

   for(i=1; i<=num; i++)
   {
      if(num % i == 0)
      {
         printf("%d, ",i);
      }
   }

   return 0;
}
```



Program 45: WAP to find all factorial of a number.

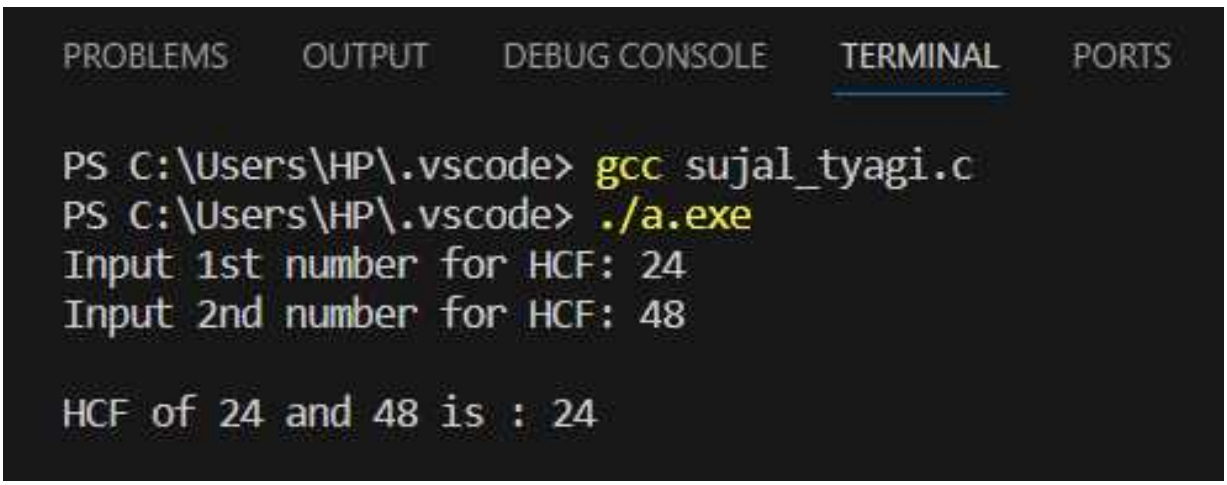
```
#include <stdio.h>

int main()
{
   int x,fact=1,n;

    printf("Enter a number to find factorial: ");

    scanf("%d",&n);

    for(x=1;x<=n;x++)

        fact=fact*x;

    printf("Factorial of %d is: %d",n,fact);

    return 0;
}
```



Program 46: WAP to find the HCF of a given number.

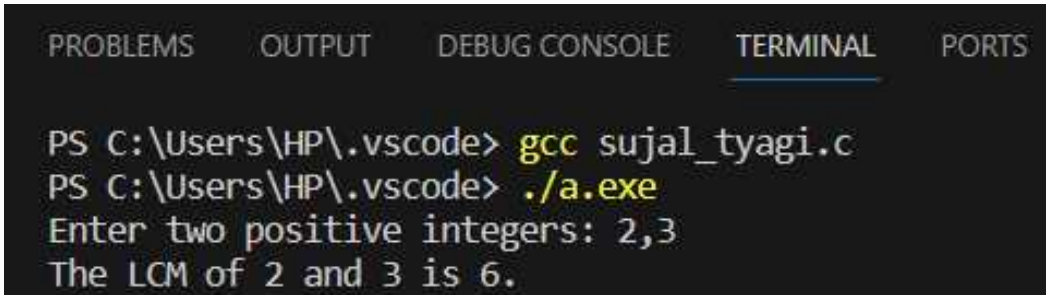
```
#include <stdio.h>


int main()
{
  int i, n1, n2, j, hcf=1;


   printf("Input 1st number for HCF: ");
   scanf("%d", &n1);
   printf("Input 2nd number for HCF: ");
   scanf("%d", &n2);


   j = (n1<n2) ? n1 : n2;


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


      if(n1%i==0 && n2%i==0)
      {
         hcf = i;
```

```
    }
  }

  printf("\nHCF of %d and %d is : %d\n\n", n1, n2, hcf);

  return 0;
}
```



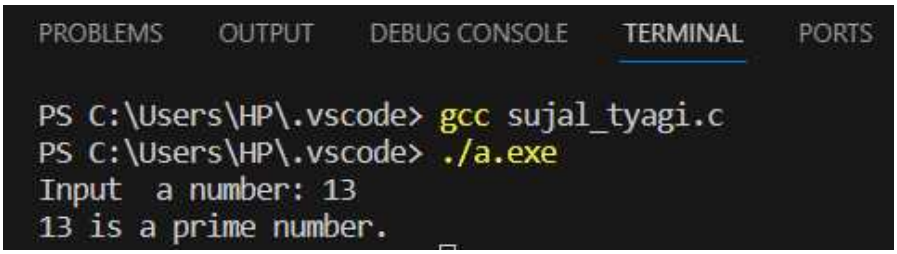
Program 47: WAP to find the LCM of a number.

```
#include <stdio.h>
int main()
{
  int n1, n2, max;
   printf("Enter two positive integers: ");
   scanf("%d,%d", &n1, &n2);
   max = (n1 > n2) ? n1 : n2;
   while (1) {
      if ((max % n1 == 0) && (max % n2 == 0)) {
         printf("The LCM of %d and %d is %d.", n1, n2, max);
         break;
      }
      ++max;
   }
   return 0
}
```



Program 48: WAP to check whether a number is prime or not.

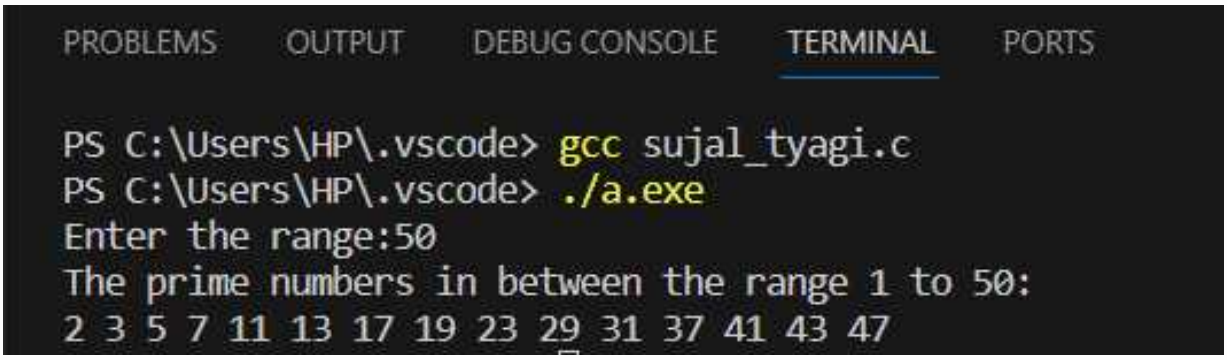
```
#include <stdio.h>


int main()
{
   int num,i,ctr=0;
    printf("Input  a number: ");
    scanf("%d",&num);
    for(i=2;i<=num/2;i++){
        if(num % i==0){
         ctr++;
            break;
        }
    }
   if(ctr==0 && num!= 1)
        printf("%d is a prime number.\n",num);
   else
     printf("%d is not a prime number",num);
    return 0;
}
```



Program 49: WAP to print all prime numbers from 1 to n.

```
#include<stdio.h>
void main(){
  int i, num, n, count;
  printf("Enter the range:");
  scanf("%d", &n);
  printf("The prime numbers in between the range 1 to %d:\n", n);
  for(num = 1;num<=n;num++){
    count = 0;
    for(i=2;i<=num/2;i++){
      if(num%i==0){
        count++;
      break;
    }
  }
  if(count==0 && num!= 1)
```

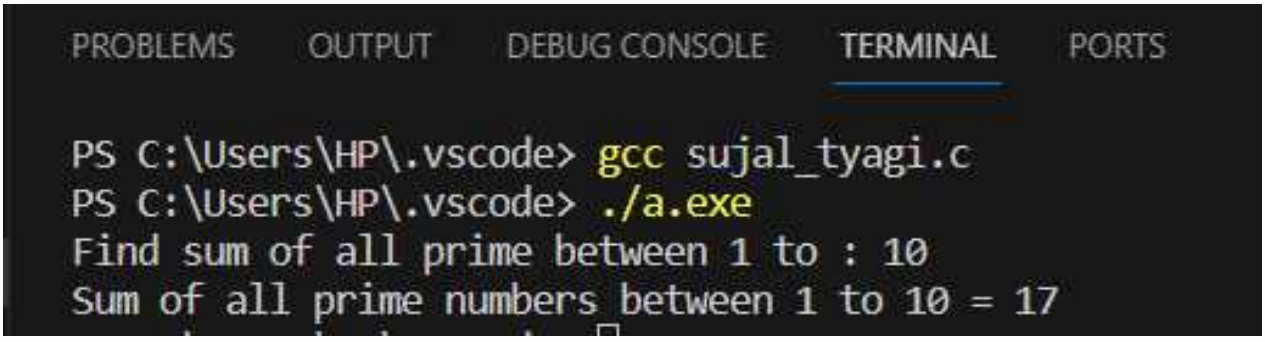
```
    printf("%d ",num);
  }
}
```


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS C:\Users\HP\.vscode> gcc sujal_tyagi.c
PS C:\Users\HP\.vscode> ./a.exe
Enter the range:50
The prime numbers in between the range 1 to 50:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47

Program 50: WAP to find sum of all prime numbers from 1 to n.

```
#include<stdio.h>
int main(){
   int i, j, end, isPrime, sum=0;

   printf("Find sum of all prime between 1 to : ");
   scanf("%d", &end);

   for(i=2; i<=end; i++)
   {

      isPrime = 1;
      for(j=2; j<=i/2 ;j++)
      {
         if(i%j==0)
         {
            isPrime = 0;
            break;
         }
      }
```

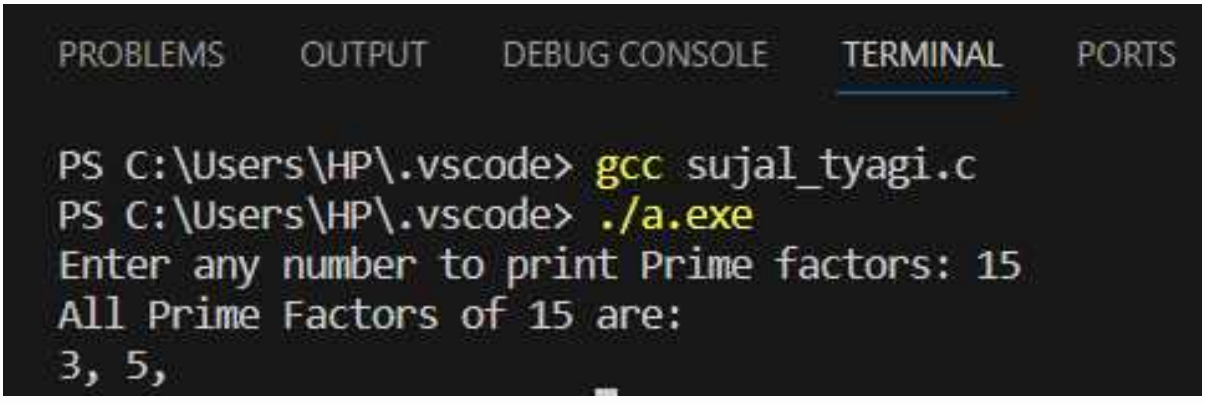
```
        if(isPrime==1)
        {
            sum += i;
        }
    }

    printf("Sum of all prime numbers between 1 to %d = %d", end, sum);

    return 0;
}
```



Program 51: WAP to find prime factors of a number.

```
#include <stdio.h>

int main()
{
   int i, j, num, isPrime;

   printf("Enter any number to print Prime factors: ");
   scanf("%d", &num);

   printf("All Prime Factors of %d are: \n", num);

   for(i=2; i<=num; i++)
   {
      if(num%i==0)
      {
         isPrime = 1;
         for(j=2; j<=i/2; j++)
         {
            if(i%j==0)
```

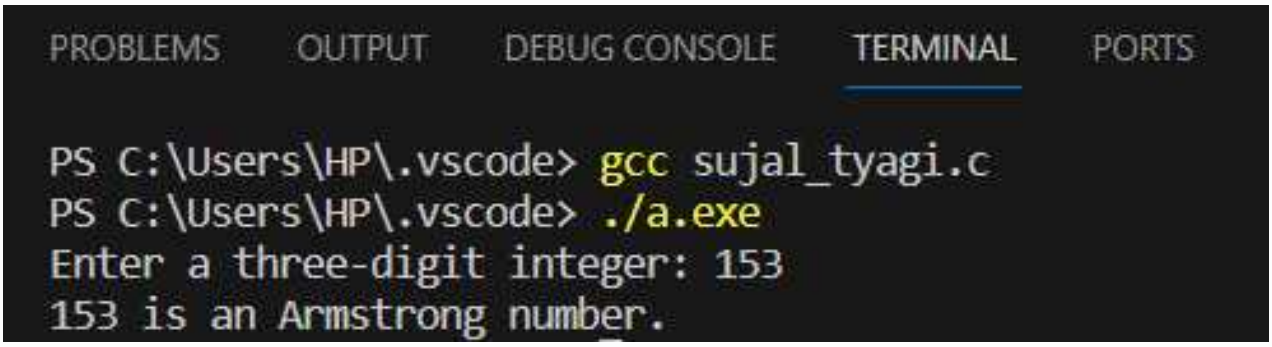
```
        {
            isPrime = 0;
            break;
          }
        }
        if(isPrime==1)
        {
          printf("%d, ", i);
        }
      }
   }
   return 0;
}
```



Program 52: WAP to check whether a number is an Armstrong number or not.

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

   int num, originalNum, remainder, result = 0;

   printf("Enter a three-digit integer: ");
   scanf("%d", &num);
   originalNum = num;

   while (originalNum != 0) {

      remainder = originalNum % 10;

     result += remainder * remainder * remainder;


     originalNum /= 10;
   }
```

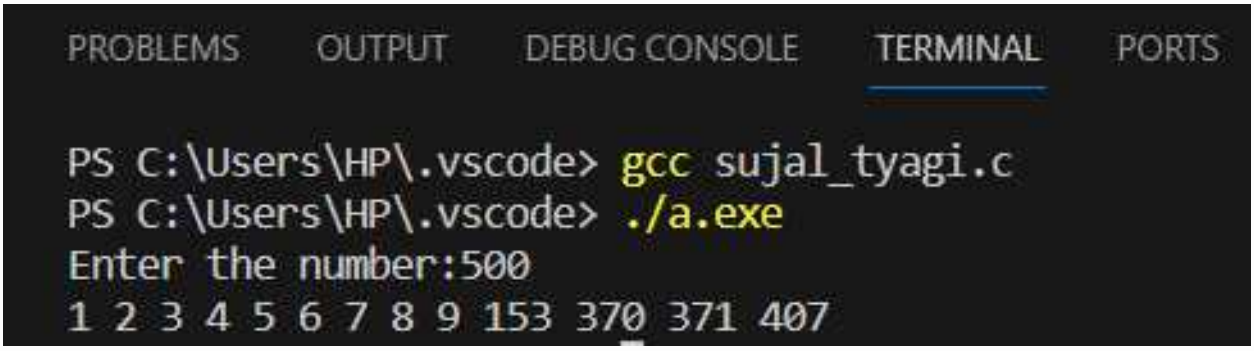
```
    if (result == num)
        printf("%d is an Armstrong number.", num);
    else
        printf("%d is not an Armstrong number.", num);

    return 0;
}
```



Program 53: WAP to print all the Armstrong numbers from 1 to n.

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

int main() {
  int i,n,sum,num,count = 0;
    printf("Enter the number:");
    scanf("%d",&n);

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

        while (num != 0) {
            num /= 10;
            count++;
        }
        num = i;
        sum = pow(num % 10, count)
              + pow((num % 100 - num % 10) / 10, count)
              + pow((num % 1000 - num % 100) / 100, count);
```

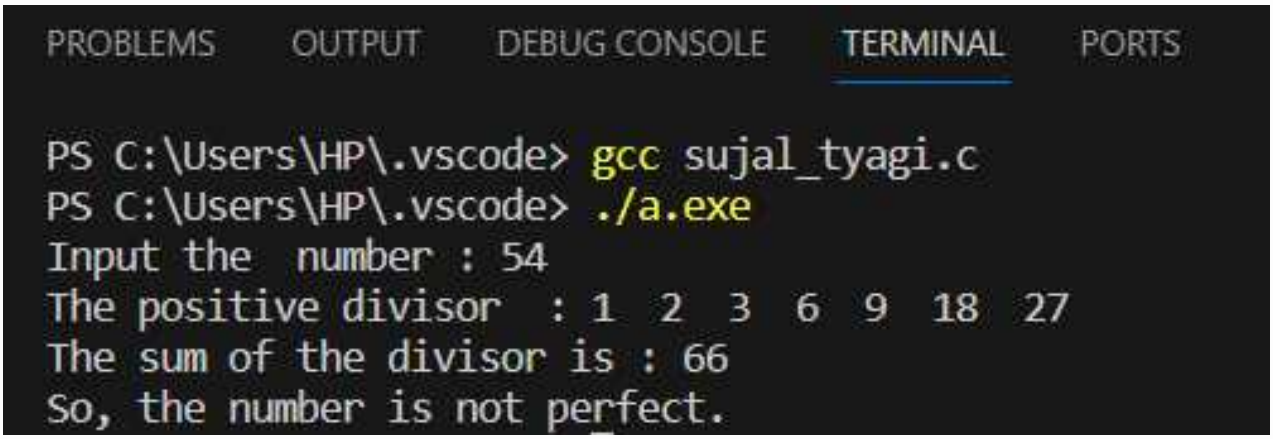
```
        if (sum == i) {
            printf("%d ", i);
        }
        count = 0;
    }
    return 0;
}
```



Program 54: WAP to check whether a number is a perfect number or not.

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

int main()
{
   int n, i, sum;
   int mn, mx;

   printf("Input the  number : ");
   scanf("%d", &n);
   sum = 0;
   printf("The positive divisor  : ");
   for (i = 1; i < n; i++)
   {
      if (n % i == 0)
      {
         sum = sum + i;
         printf("%d  ", i);
      }
```

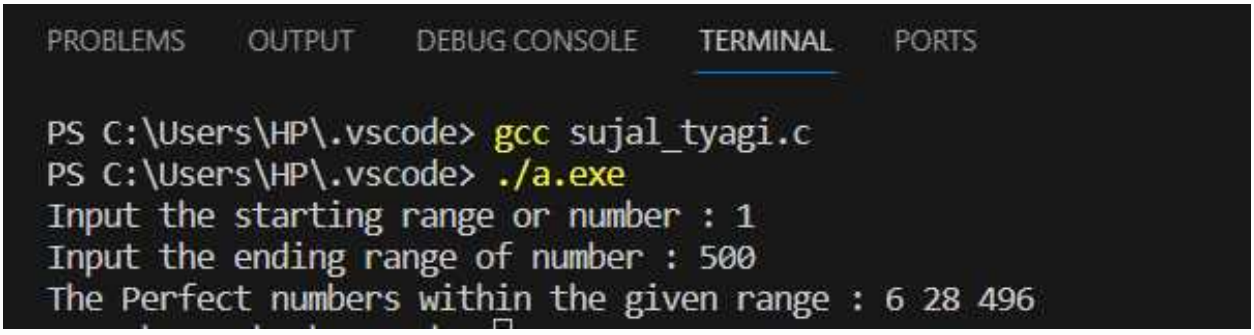
```
    }
    printf("\nThe sum of the divisor is : %d", sum);
    if (sum == n)
        printf("\nSo, the number is perfect.");
    else
        printf("\nSo, the number is not perfect.");
    printf("\n");
    return 0;
}
```



Program 55: WAP to print all the perfect numbers from 1 to n.

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

int main() {
int n,i,sum,mn,mx;
 printf("Input the starting range or number : ");
 scanf("%d",&mn);
 printf("Input the ending range of number : ");
 scanf("%d",&mx);
 printf("The Perfect numbers within the given range : ");
 for(n=mn;n<=mx;n++){
  i=1;
  sum = 0;
  while(i<n){
   if(n%i==0)
       sum=sum+i;
      i++;
  }
  if(sum==n)
```

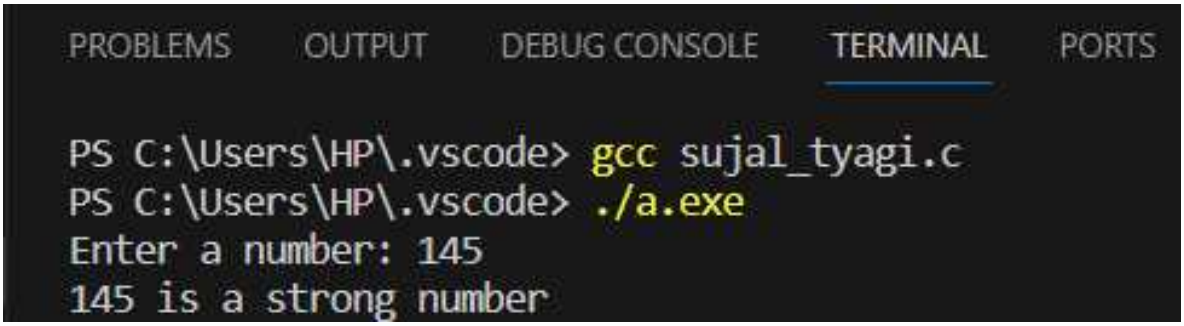
```
    printf("%d ",n);
  }
    printf("\n");
   return 0;
}
```



Program 56: WAP to check whether a number is a strong number or not.

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

int main()
{
   int num, i, f, r, sum = 0, temp;
   printf("Enter a number: ");
   scanf("%d", &num);
   temp = num;
   while (num)
   {
      i = 1, f = 1;
      r = num % 10;
      while (i <= r)
      {
         f = f * i;
         i++;
      }
      sum = sum + f;
```

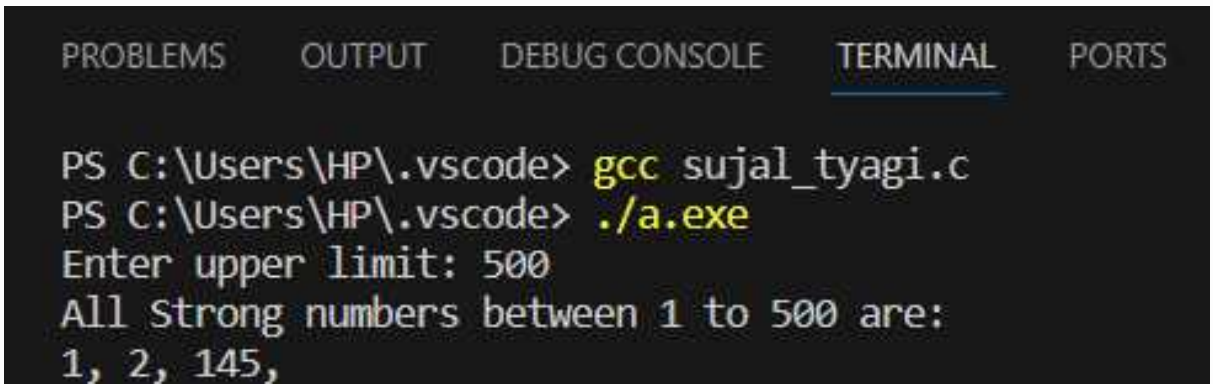
```
        num = num / 10;
    }
    if (sum == temp)
        printf("%d is a strong number", temp);
    else
        printf("%d is not a strong number", temp);

    return 0;
}
```



Program 57: WAP to print all the strong numbers from 1 to n.

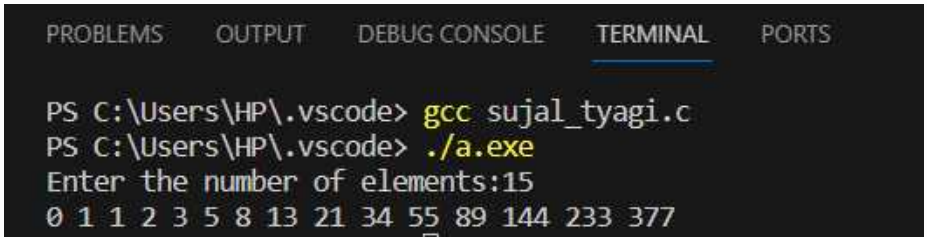
```
#include <stdio.h>
int main()
{
   int i, j, cur, lastDigit, end;
   long long fact, sum;

   printf("Enter upper limit: ");
   scanf("%d", &end);

   printf("All Strong numbers between 1 to %d are:\n", end);

   for (i = 1; i <= end; i++)
   {
      cur = i;
sum = 0;

      while (cur > 0)
      {
         fact = 1ll;
```

```
        lastDigit = cur % 10;


        for (j = 1; j <= lastDigit; j++)
        {
           fact = fact * j;
        }
        sum += fact;
        cur /= 10;
     }
     if (sum == i)
     {
        printf("%d, ", i);
     }
   }
  return 0;
}
```



Program 58: WAP to print Fibonacci terms upto n.

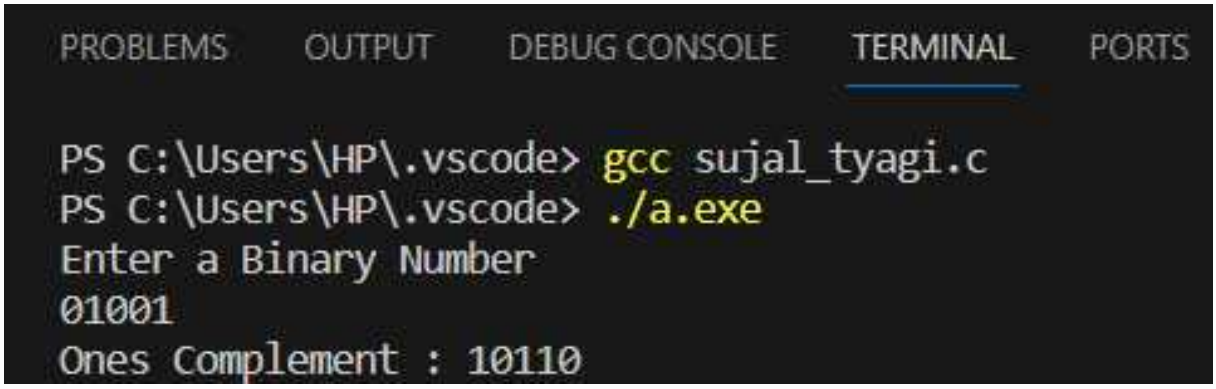
```
#include <stdio.h>

int main()

{

 int n1=0,n2=1,n3,i,number;

 printf("Enter the number of elements:");

 scanf("%d",&number);

 printf("\n%d %d",n1,n2);

 for(i=2;i<number;++i)

 {

  n3=n1+n2;

  printf(" %d",n3);

  n1=n2;

  n2=n3;

 }

    return 0;

}
```



Program 59: WAP to find the one's complement of a binary number.

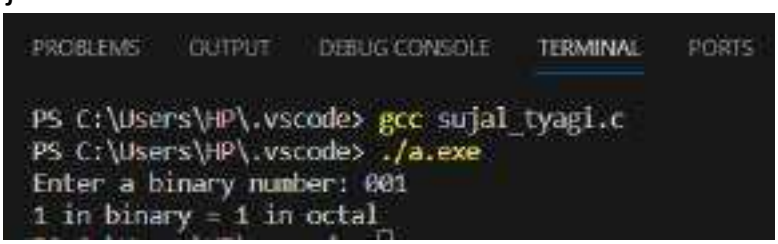
```
#include <stdio.h>
#include<string.h>

int main()
{
 char binaryNumber[100], onesComplement[100];
   int counter, error=0, digitCount;

   printf("Enter a Binary Number\n");
   scanf("%s", binaryNumber);

   digitCount = strlen(binaryNumber);

   for(counter=0; counter < digitCount; counter++) {
      if(binaryNumber[counter]=='1') {
         onesComplement[counter] = '0';
      } else if(binaryNumber[counter]=='0') {
         onesComplement[counter] = '1';
      } else {
```

```
            printf("Error :( ");
            return 1;
        }
    }
    onesComplement[digitCount] = '\0';

    printf("Ones Complement : %s", onesComplement);
    return 0;
}
```



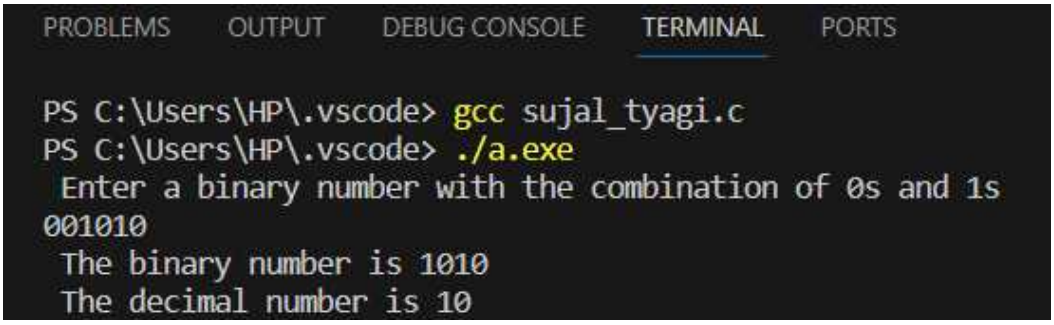
Program 61: WAP to convert binary to octal number system.

```
#include <math.h>
#include <stdio.h>
int convert(long long bin);
int main() {
   long long bin;
   printf("Enter a binary number: ");
   scanf("%lld", &bin);
   printf("%lld in binary = %d in octal", bin, convert(bin));
   return 0;
}

int convert(long long bin) {
   int oct = 0, dec = 0, i = 0;
   while (bin != 0) { dec += (bin % 10) * pow(2, i);
      ++i;
      bin /= 10;
   }
   i = 1;
   while (dec != 0) { oct += (dec % 8) * i;
      dec /= 8;
      i *= 10;
   }
   return oct;
}
```



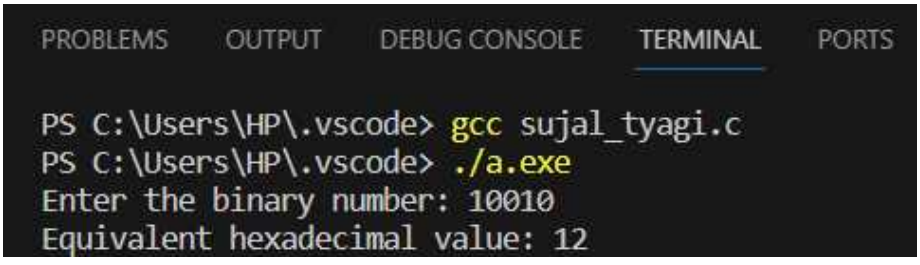
Program 62: WAP to convert binary to decimal number system.

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

int main() {
   int num, binary_num, decimal_num = 0, base = 1, rem;
   printf (" Enter a binary number with the combination of 0s and 1s \n");
   scanf (" %d", &num);

   binary_num = num;
   while ( num > 0)
   {
      rem = num % 10;
      decimal_num = decimal_num + rem * base;
      num = num / 10;
      base = base * 2;
   }

   printf ( " The binary number is %d \t", binary_num);
   printf (" \n The decimal number is %d \t", decimal_num);
   }
```



Program 63: WAP to convert binary to hexadecimal number system.

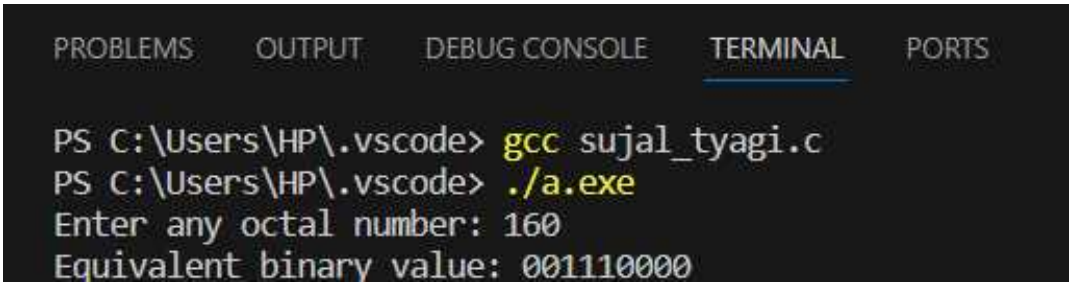
```
#include <stdio.h>
int main()
{
   long int binaryval, hexadecimalval = 0, i = 1, remainder;
   printf("Enter the binary number: ");
   scanf("%ld", &binaryval);
   while (binaryval != 0)
   {
      remainder = binaryval % 10;
      hexadecimalval = hexadecimalval + remainder * i;
      i = i * 2;
      binaryval = binaryval / 10;
   }
   printf("Equivalent hexadecimal value: %lX", hexadecimalval);
   return 0;
}
```



Program 64: WAP to convert octal to binary number system.

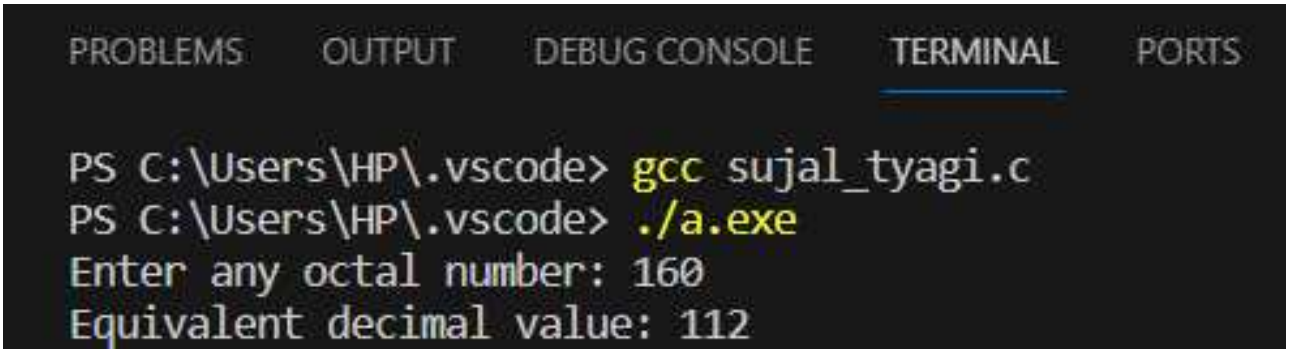
```
#include <stdio.h>
#define MAX 1000

int main()
{
   char octalnum[MAX];
   long i = 0;

   printf("Enter any octal number: ");
   scanf("%s", octalnum);

   printf("Equivalent binary value: ");
   while (octalnum[i])
   {
      switch (octalnum[i])
      {
      case '0':
         printf("000"); break;
      case '1':
         printf("001"); break;
      case '2':
         printf("010"); break;
      case '3':
         printf("011"); break;
      case '4':
         printf("100"); break;
      case '5':
         printf("101"); break;
```

```
        case '6':
            printf("110"); break;
        case '7':
            printf("111"); break;
        default:
            printf("\n Invalid octal digit %c ", octalnum[i]);
            return 0;
        }
        i++;
    }
    return 0;
}
```



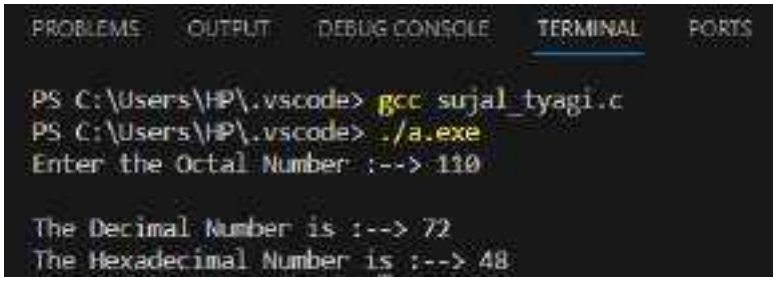
Program 65: WAP to convert octal to decimal number system.

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

int main()
{

   long int octal, decimal = 0;
   int i = 0;

   printf("Enter any octal number: ");
   scanf("%ld", &octal);
   while (octal != 0)
   {
      decimal =  decimal +(octal % 10)* pow(8, i++);
      octal = octal / 10;
   }
   printf("Equivalent decimal value: %ld",decimal);
   return 0;
}
```



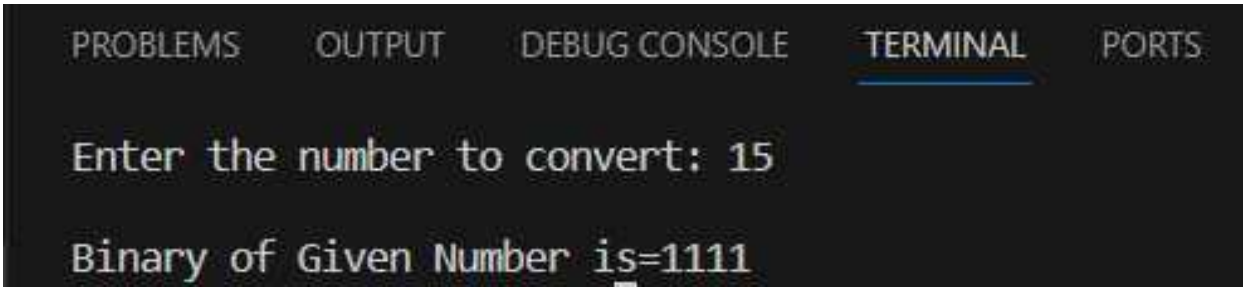
Program 66: WAP to convert octal to hexadecimal number system.

```
#include <stdio.h>
#include <math.h>
int main()
{
    int n, sum = 0;
    printf("Enter the Octal Number :--> ");
    scanf("%d", &n);
    int i = 0;
    while(n != 0)
    {
        int digit = n % 10;
        sum = sum + (digit * pow(8,i));
        n = n / 10;
        i++;
    }
    printf("\nThe Decimal Number is :--> %d",sum);
    int ans = 0,j = 0;
    while(sum != 0)
    {
        int digit = sum % 16;
        ans = ans + (digit * pow(10, j));
        sum = sum / 16;         j++;
    } printf("\nThe Hexadecimal Number is :--> %d",ans);
}
```



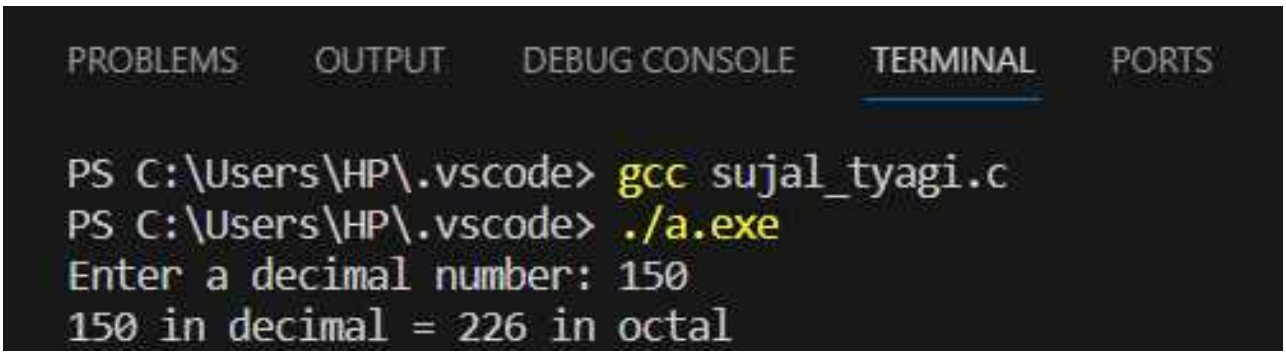
Program 67: WAP to convert decimal to binary number system.

```
#include <stdio.h>
#include <math.h>
int main()
{
   int a[10],n,i;
system ("cls");
printf("Enter the number to convert: ");
scanf("%d",&n);
for(i=0;n>0;i++)
{
a[i]=n%2;
n=n/2;
}
printf("\nBinary of Given Number is=");
for(i=i-1;i>=0;i--)
{
printf("%d",a[i]);
}
    return 0;
}
```



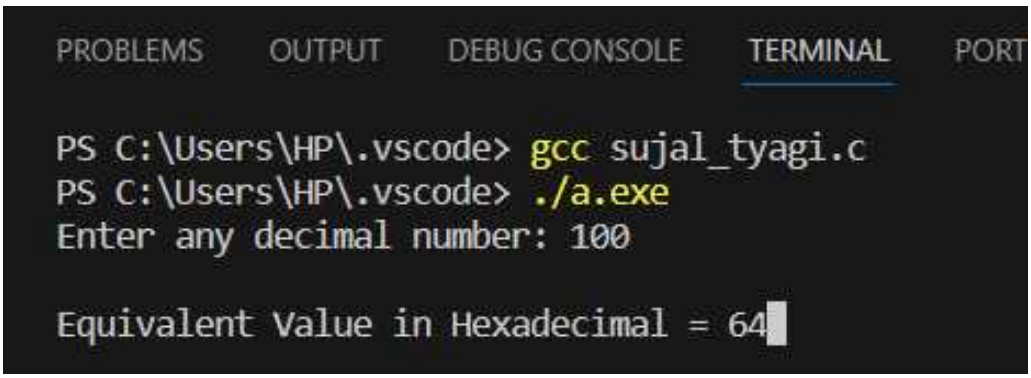
Program 68: WAP to convert decimal to octal number system.

```
#include <stdio.h>
#include <math.h>
int main()
{
   int a[10],n,i;
system ("cls");
printf("Enter the number to convert: ");
scanf("%d",&n);
for(i=0;n>0;i++)
{
a[i]=n%2;
n=n/2;
}
printf("\nBinary of Given Number is=");
for(i=i-1;i>=0;i--)
{
printf("%d",a[i]);
}
    return 0;
}
```



Program 69: WAP to convert decimal to hexadecimal number system.

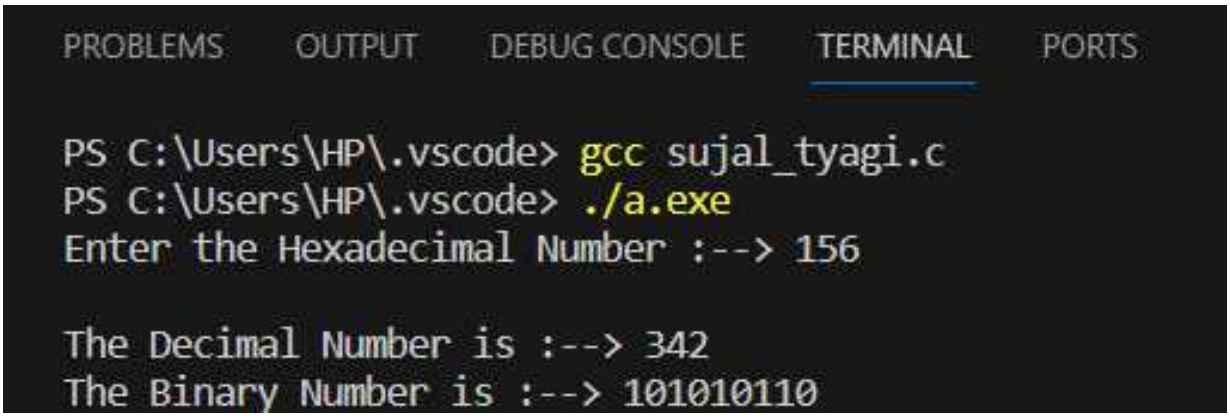
```
#include<stdio.h>
int main()
{
   int decnum, rem, i=0;
   char hexnum[50];
   printf("Enter any decimal number: ");
   scanf("%d", &decnum);
   while(decnum!=0)
   {
      rem = decnum%16;
      if(rem<10)
         rem = rem+48;
      else
         rem = rem+55;
      hexnum[i] = rem;
      i++; decnum = decnum/16;
   }
   printf("\nEquivalent Value in Hexadecimal = ");
   for(i=i-1; i>=0; i--)
      printf("%c", hexnum[i]);
}
```



Program 70: WAP to convert hexadecimal to binary number system.

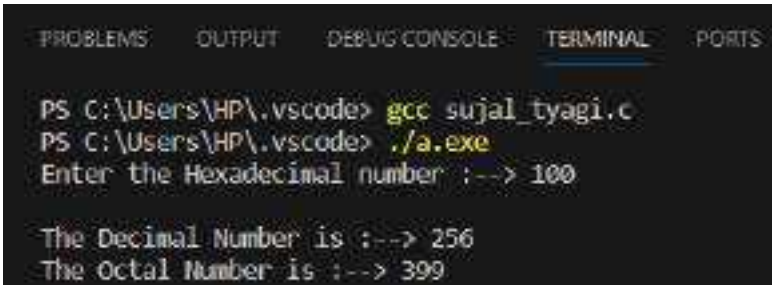
```
#include <stdio.h>
#include <math.h>
int main()
{
    int n, ans = 0, B[100];

    printf("Enter the Hexadecimal Number :--> ");
    scanf("%d", &n);

    int i = 0;
    while(n != 0)
    {
        int digit = n % 10;
        ans = ans + (digit * pow(16, i));
        n = n / 10;
        i++;
    }

    printf("\nThe Decimal Number is :--> %d", ans);

    int j = 0, k;
    while(ans > 0)
    {
        B[j] = ans % 2;             //to store the remainder in array
        ans = ans >> 1;
        j++;
    }
```

```
    printf("\nThe Binary Number is :--> ");

    for(k = j - 1; k >= 0; k--)
    {
        printf("%d",B[k]);
    }
    return 0;
}
```



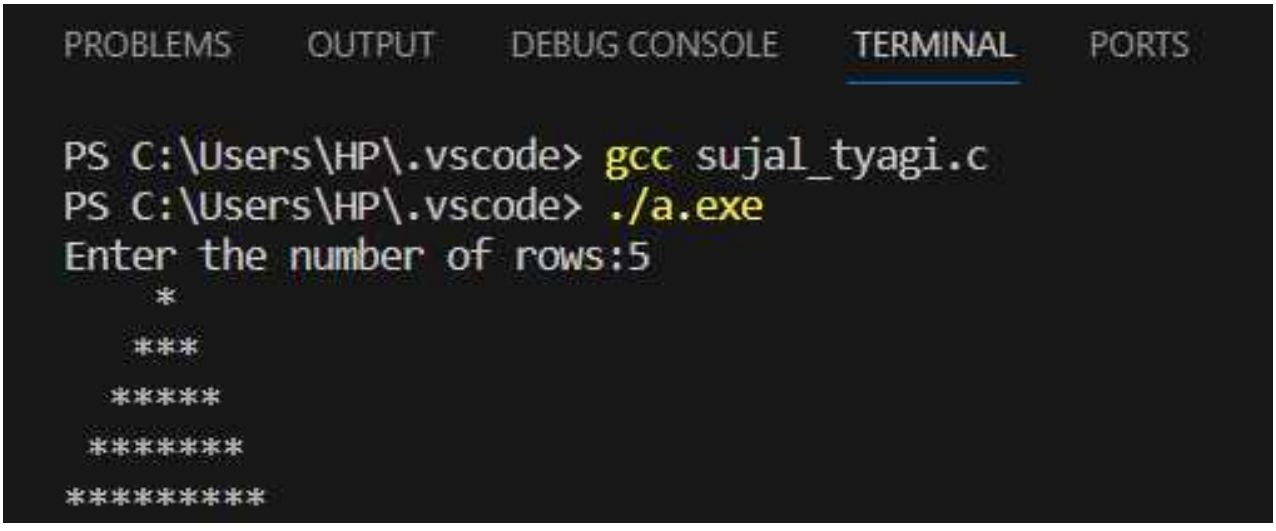
Program 71: WAP to convert hexadecimal to octal and decimal number system.

```
#include <stdio.h>
#include <math.h>
int main()
{
    int n, ans = 0;
    printf("Enter the Hexadecimal number :--> ");
    scanf("%d", &n);
    int i = 0;
    while(n != 0)
    {
        int digit = n % 10;
        ans = ans + (digit * pow(16, i));
        n = n / 10;         i++;
    }
    printf("\nThe Decimal Number is :--> %d", ans);
    int j = 0, ans2 = 0;
    while(ans != 0)
    {
        int digit = ans%8;
        ans2 = ans2 + (digit * pow(10, j));
        ans = ans / 8;         j++;

    } printf("\nThe Octal Number is :--> %d", ans2); }
```



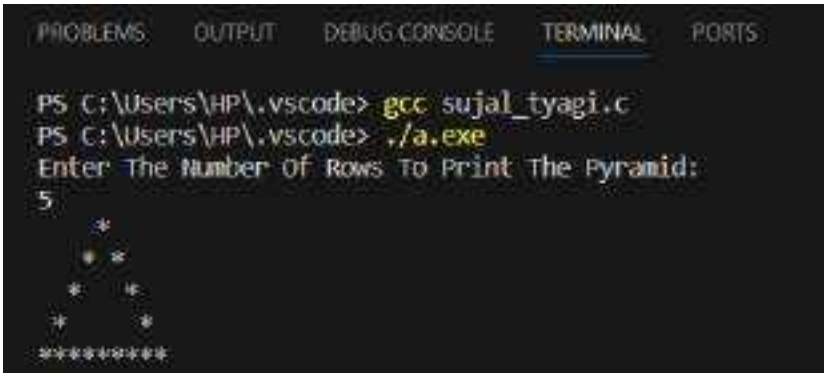
Program 73: Star Patter 1.

```
#include<stdio.h>
int main(){
   int i,j,rows,space;

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

   for(i=1;i<=rows;i++){
      for(space=1;space<=rows-i;space++){
         printf(" ");
      }for(j=1;j<=2*i-1;j++){
         printf("*");
      }printf("\n");
   }
}
```



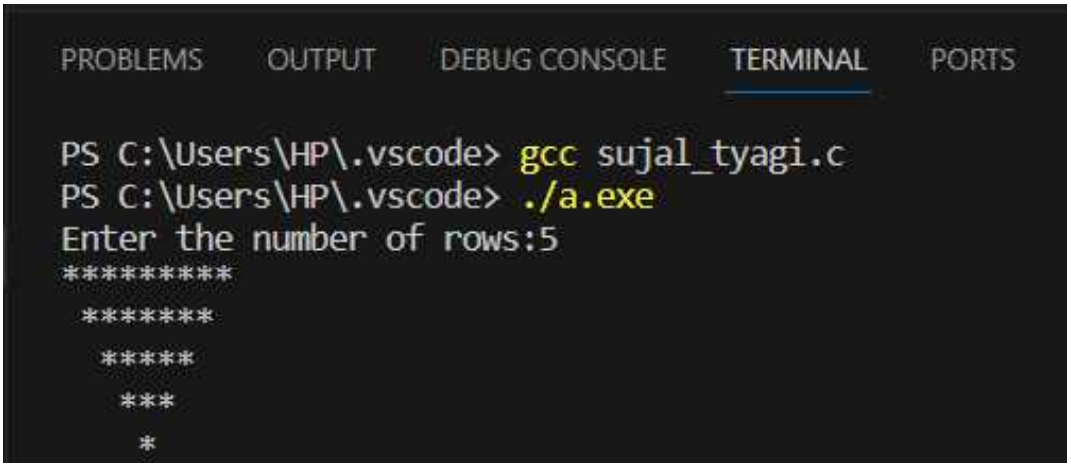
Program 74: Star Patter 2.

```
#include<stdio.h>

int main() {
int i, space, rows, star=0;
printf("Enter The Number Of Rows To Print The Pyramid: \n");
scanf("%d",&rows);
for(i = 0; i < rows-1; i++) {
for(space = 1; space < rows-i; space++) {
printf(" ");
}
for (star = 0; star <= 2*i; star++) {
if(star==0 || star==2*i)
printf("*");
else
printf(" ");
}
printf("\n");
}
for(i=0; i<2*rows-1; i++){
printf("*");
}
return 0;
}
```



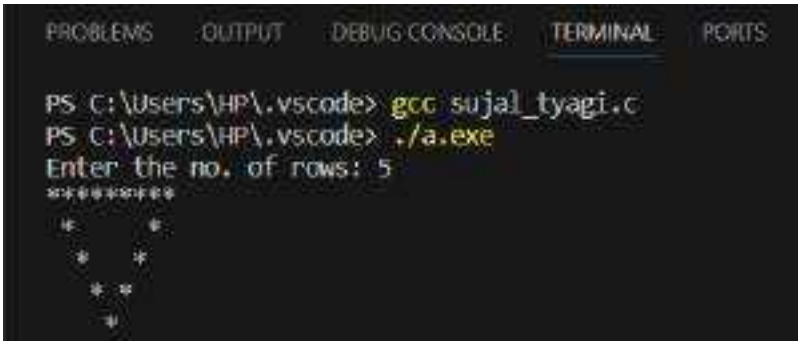
Program 75: Star Patter 3.

```
#include<stdio.h>

int main() {

int i, space, rows,j;

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

for(i=rows;i>=1;i--){
   for(space=1;space<=rows-i;space++){
      printf(" ");
   }for(j=1;j<=2*i-1;j++){
      printf("*");
   }printf("\n");
}

return 0;
}
```



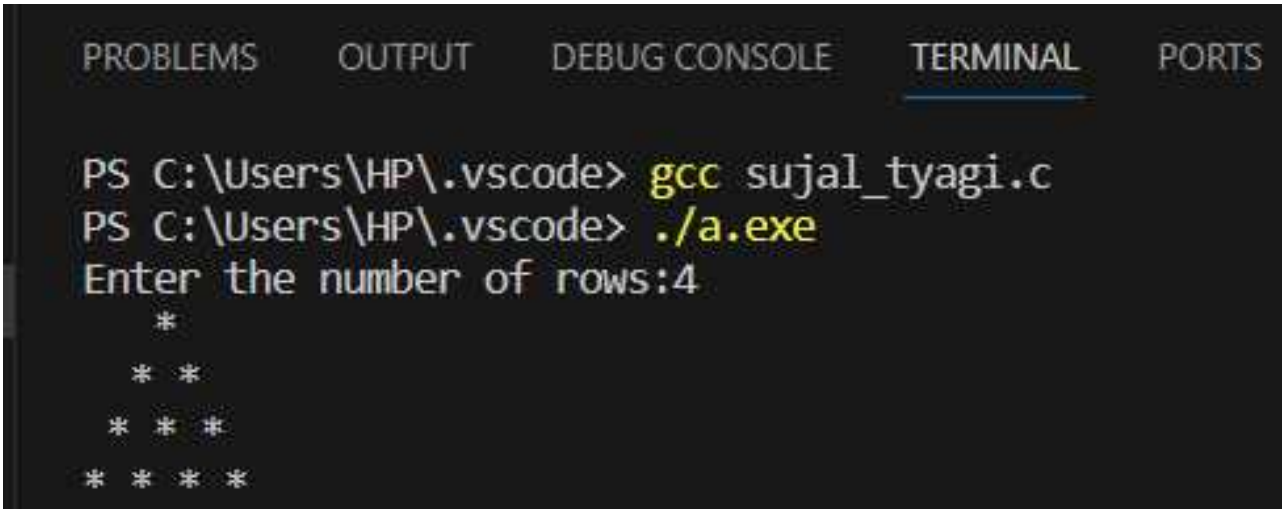
Program 76: Star Patter 4.

```
#include <stdio.h>
int main()
{
  int i, j, k, m = 1, rows;
  printf("Enter the no. of rows: ");
  scanf("%d", &rows);

  for (i = rows; i >= 1; i--)
  {
    for (j = 1; j < m; j++)
      printf(" ");

    for (k = 1; k <= 2 *i - 1; k++)
    {
      if (k == 1 || k == 2 *i - 1 || i == rows)
        printf("*");
      else
        printf(" ");
    }
    m++;
    printf("\n");
  }
}
```



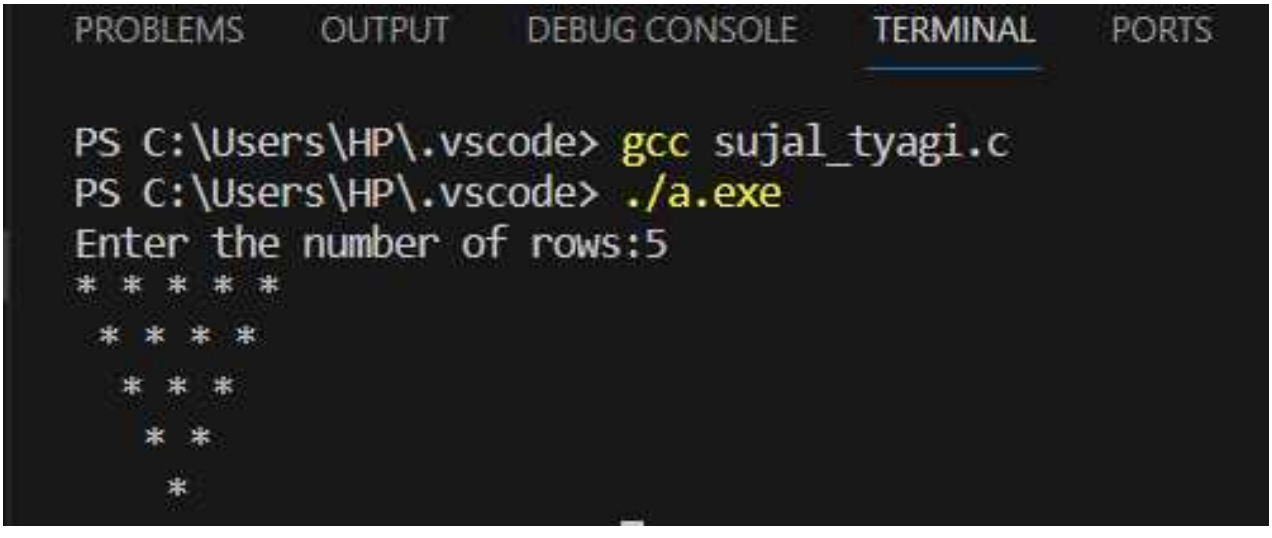
Program 77: Star Patter 5.

```
#include<stdio.h>

int main(){

   int i,j,rows,space;

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

   for(i=1;i<=rows;i++){
      for(space=1;space<=rows-i;space++){
         printf(" ");
      }for(j=1;j<=i;j++){
         printf("* ");
      }printf("\n");
   }
}
```



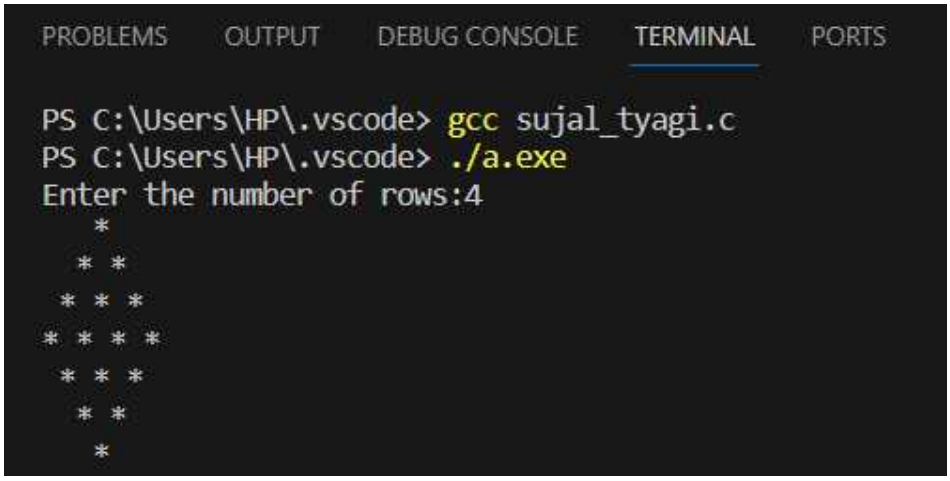
Program 78: Star Patter 6.

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

   int i,j,rows,space;

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

   for(i=rows;i>=1;i--){
      for(space=1;space<=rows-i;space++){
         printf(" ");
      }for(j=1;j<=i;j++){
         printf("* ");
      }printf("\n");
   }
}
```



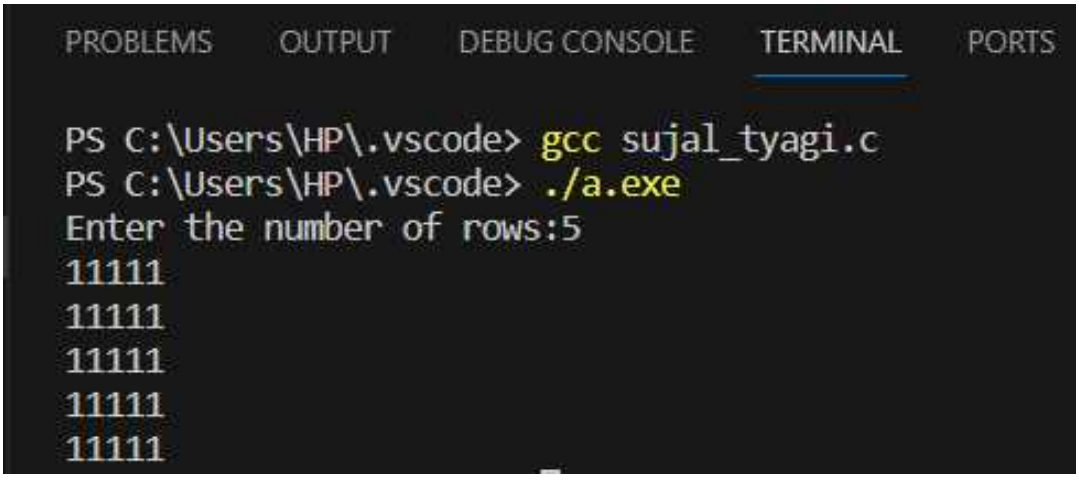
Program 79: Star Patter 7.

```
#include<stdio.h>
int main(){
   int i,j,rows,space;
   printf("Enter the number of rows:");
   scanf("%d",&rows);
    for(i=1;i<=rows;i++){
      for(space=1;space<=rows-i;space++){
         printf(" ");
      }for(j=1;j<=i;j++){
         printf("* ");
      }printf("\n");
   }
   for(i=rows-1;i>=1;i--){
      for(space=1;space<=rows-i;space++){
         printf(" ");
      }for(j=1;j<=i;j++){
         printf("* ");
      }printf("\n");
   }
}
```



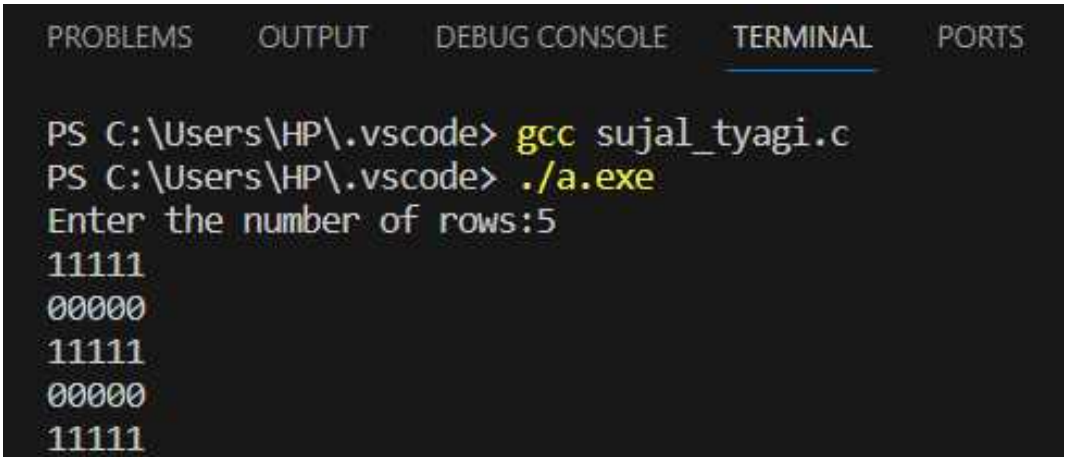
Program 80: Square Pattern 1.

```
#include<stdio.h>
int main(){
   int i,j,rows,space;
   printf("Enter the number of rows:");
   scanf("%d",&rows);
   for(i=1;i<=rows;i++){
      for(j=1;j<=rows;j++){
         printf("1");
      }printf("\n");
   }
}
```



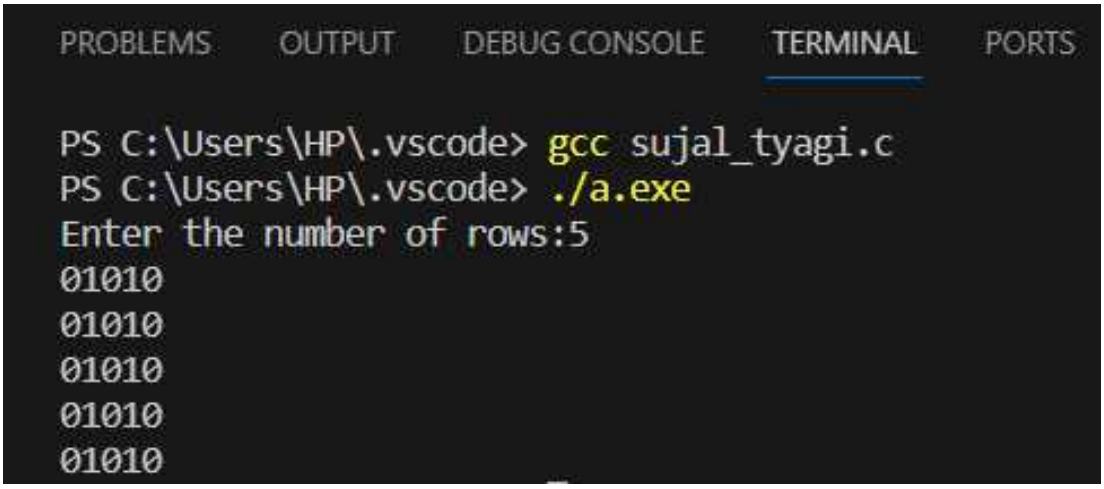
Program 81: Square Pattern 2.

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

   int i,j,rows,space;

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

   for(i=1;i<=rows;i++){
      for(j=1;j<=rows;j++){
         if(i%2==0){
            printf("0");
         }else{
            printf("1");
         }
      }printf("\n");
   }
}
```



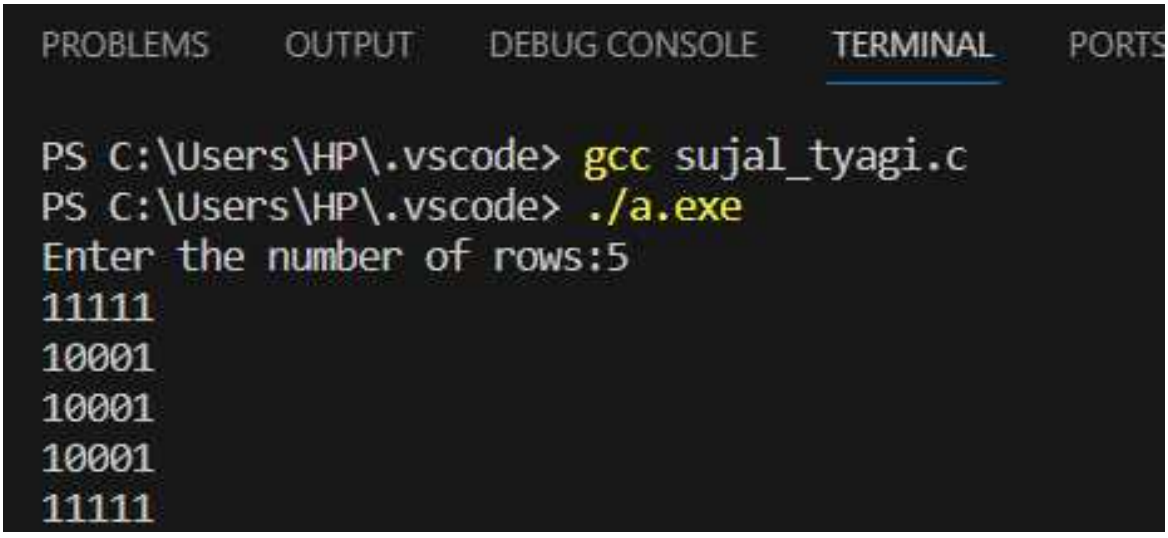
Program 82: Square Pattern 3.

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

   int i,j,rows,space;

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

   for(i=1;i<=rows;i++){
      for(j=1;j<=rows;j++){
         if(j%2==1){
            printf("0");
         }else{
            printf("1");
         }
      }printf("\n");
   }
}
```



Program 83: Square Pattern 4.

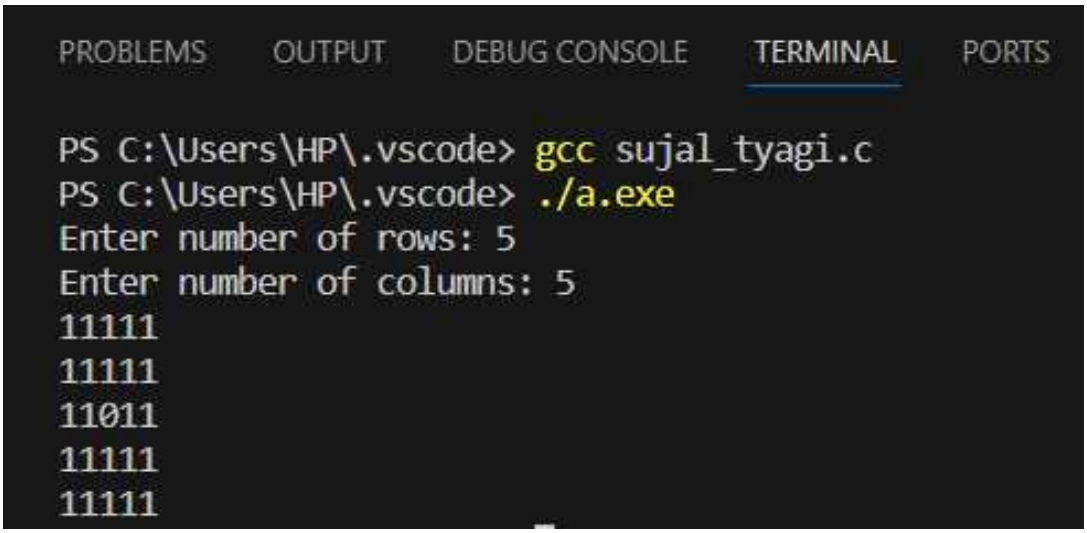
```
#include<stdio.h>
int main(){

   int i,j,rows,space;

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

   for(i=1;i<=rows;i++){
      for(j=1;j<=rows;j++){
         if(i==1 || i==rows || j==1 || j==rows){
            printf("1");
         }else{
            printf("0");
         }
      }printf("\n");
   }
}
```



Program 84: Square Pattern 5.

```
#include <stdio.h>

int main()
{
   int rows, cols, i, j;
   int centerRow, centerCol;

   /* Input rows and columns from user */
   printf("Enter number of rows: ");
   scanf("%d", &rows);
   printf("Enter number of columns: ");
   scanf("%d", &cols);

   /* Find center row and column */
   centerRow = (rows + 1) / 2;
   centerCol = (cols + 1) / 2;

   for(i=1; i<=rows; i++)
   {
      for(j=1; j<=cols; j++)
      {
         if(centerCol == j && centerRow == i)
         {
            printf("0");
         }
         else if(cols%2 == 0 && centerCol+1 == j)
         {
            if(centerRow == i || (rows%2 == 0 && centerRow+1 == i))
               printf("0");
            else
```

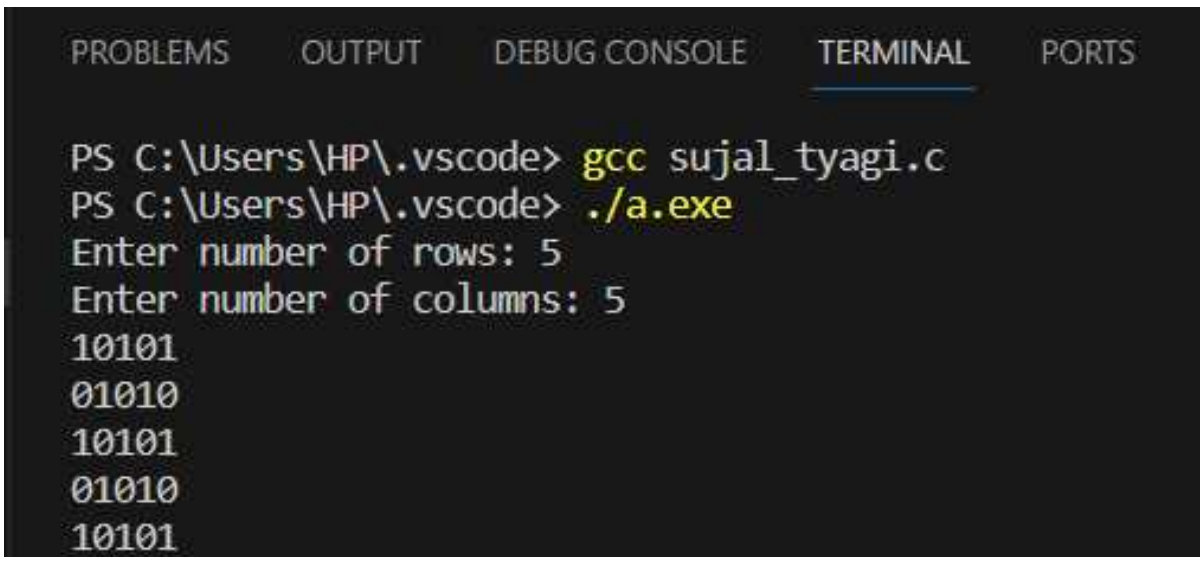
```
            printf("1");
        }
        else if(rows%2 == 0 && centerRow+1 == i)
        {
            if(centerCol == j || (cols%2 == 0 && centerCol+1 == j))
                printf("0");
            else
                printf("1");
        }
        else
        {
            printf("1");
        }
    }

    printf("\n");
  }

  return 0;
}
```



Program 85: Square Pattern 6.

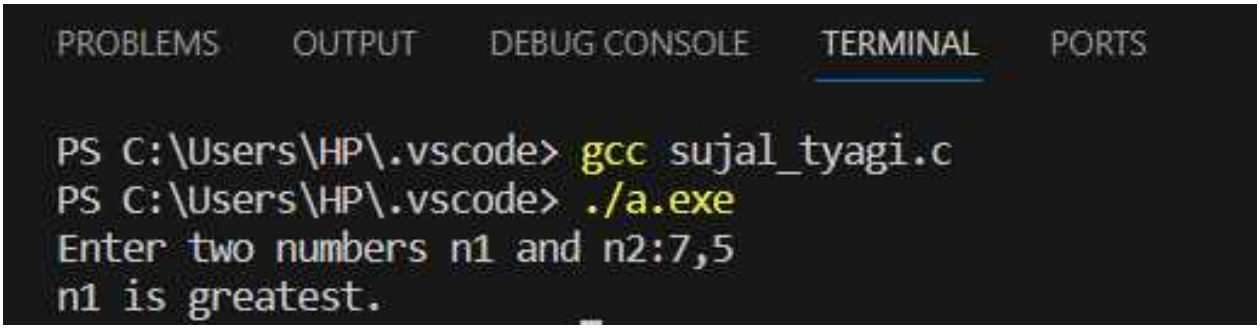
```
#include <stdio.h>

int main()
{
   int rows, cols, i, j, k;

   /* Input rows and columns from user */
   printf("Enter number of rows: ");
   scanf("%d", &rows);
   printf("Enter number of columns: ");
   scanf("%d", &cols);

   k = 1;

   for(i=1; i<=rows; i++)
   {
      for(j=1; j<=cols; j++)
      {
         if(k == 1)
         {
            printf("1");
         }
         else
         {
            printf("0");
         }

         // If k = 1  then k *= -1 => -1
         // If k = -1 then k *= -1 =>  1
         k *= -1;
```

```
        }

        if(cols % 2 == 0)
        {
            k *= -1;
        }

        printf("\n");
    }

    return 0;
}
```



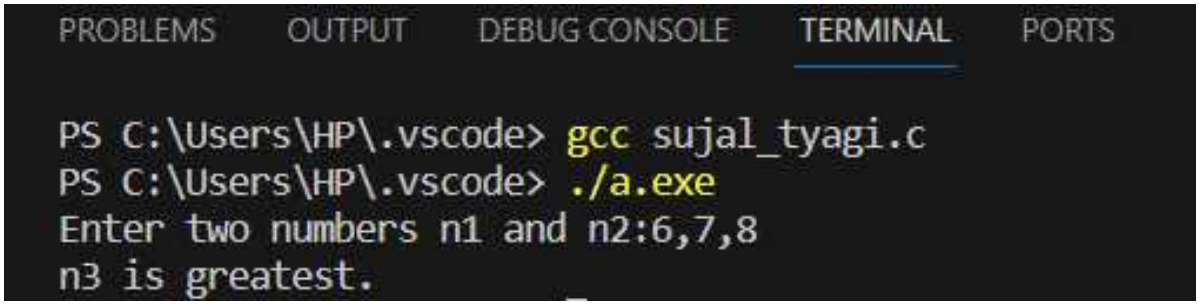
Program 86: Write a C program to find maximum between two numbers.

```
#include<stdio.h>
int main(){
   int n1,n2;

   printf("Enter two numbers n1 and n2:");
   scanf("%d,%d",&n1,&n2);

   if((n1>n2) && (n2<n1)){
      printf("n1 is greatest.");
   }
else if((n2>n1) && (n1<n2)){
      printf("n2 is greatest.");
   }
}
```



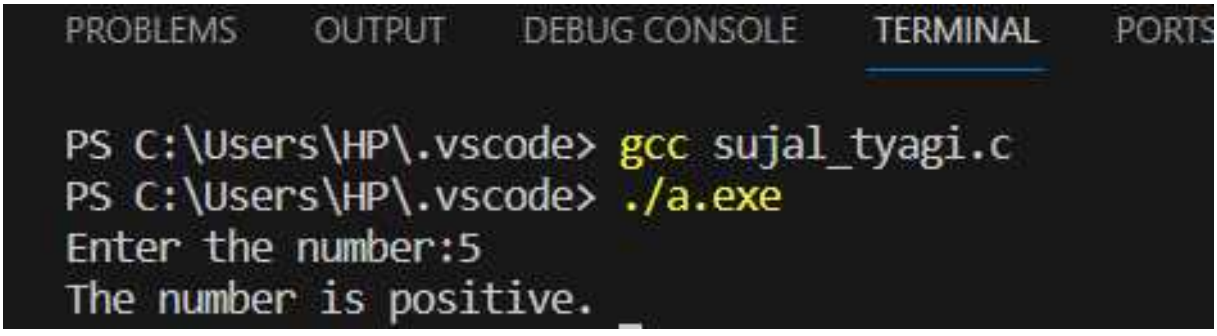
Program 87: Write a C program to find maximum between three numbers.

```
#include<stdio.h>
int main(){
   int n1,n2,n3;

   printf("Enter two numbers n1 and n2:");
   scanf("%d,%d,%d",&n1,&n2,&n3);

   if((n1>n2) && (n1>n3)){
      printf("n1 is greatest.");
   }
else if((n2>n1) && (n2>n3)){
      printf("n2 is greatest.");
   }
else if((n3>n1) && (n3>n2)){
      printf("n3 is greatest.");
   }
}
```



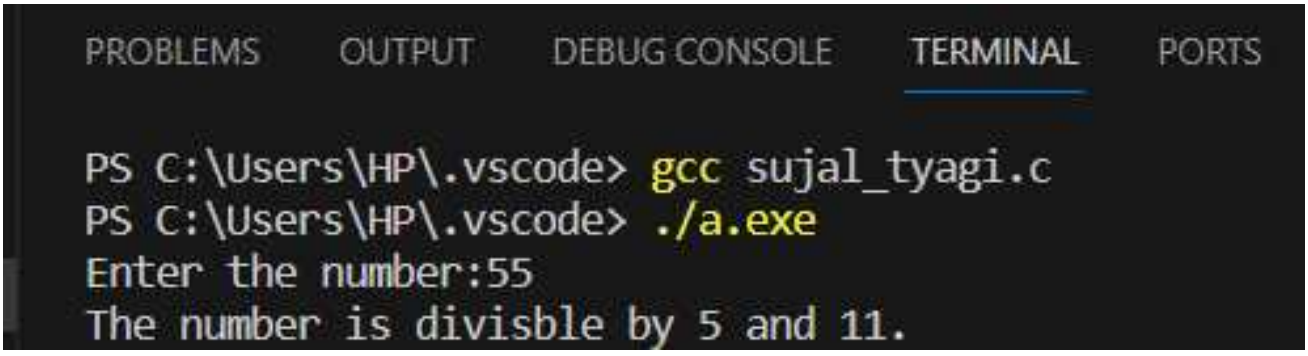
Program 88: Write a C program to check whether a number is negative, positive or zero.

```
#include<stdio.h>
int main(){
   int num;

   printf("Enter the number:");
   scanf("%d",&num);

   if(num>0){
      printf("The number is positive.");
   }
   else if(num<0){
      printf("The number is negative.");
   }
   else if(num==0){
      printf("The number is zero.");
   }
}
```



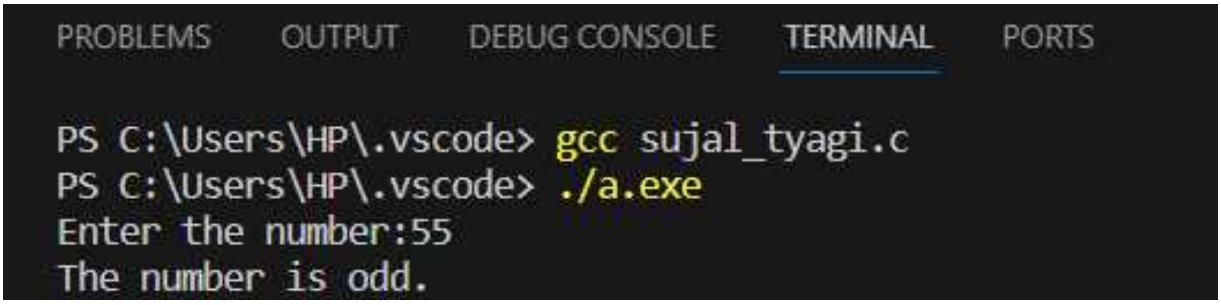
Program 89: Write a C program to check whether a number is divisible by 5 and 11 or not.

```
#include <stdio.h>
int main()
{
   int num;

   printf("Enter the number:");
   scanf("%d", &num);

   if (num / 5 && num / 11)
   {
      printf("The number is divisble by 5 and 11.");
   }
   else
   {
      printf("The number is not divisible by 5 and 11 both.");
   }
}
```



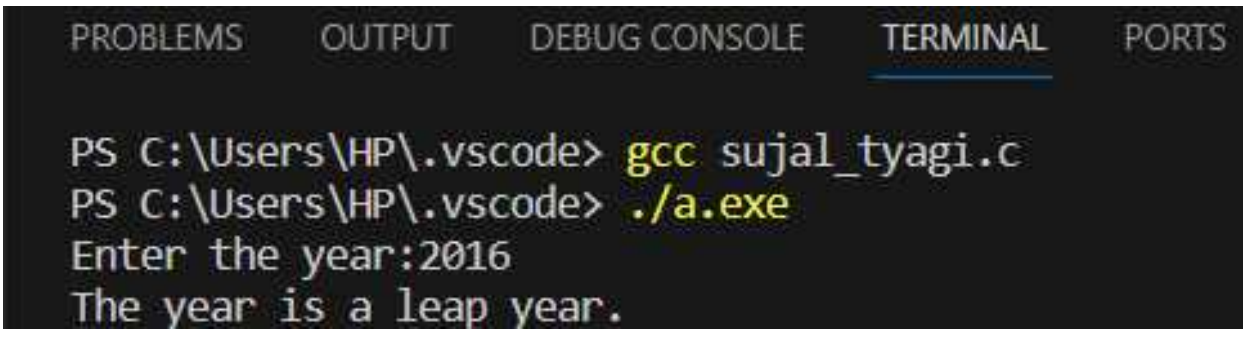
Program 90: Write a C program to check whether a number is even or odd.

```
#include <stdio.h>
int main()
{
   int num;

   printf("Enter the number:");
   scanf("%d", &num);

   if (num % 2 == 0)
   {
      printf("The number is even.");
   }
   else
   {
      printf("The number is odd.");
   }
}
```



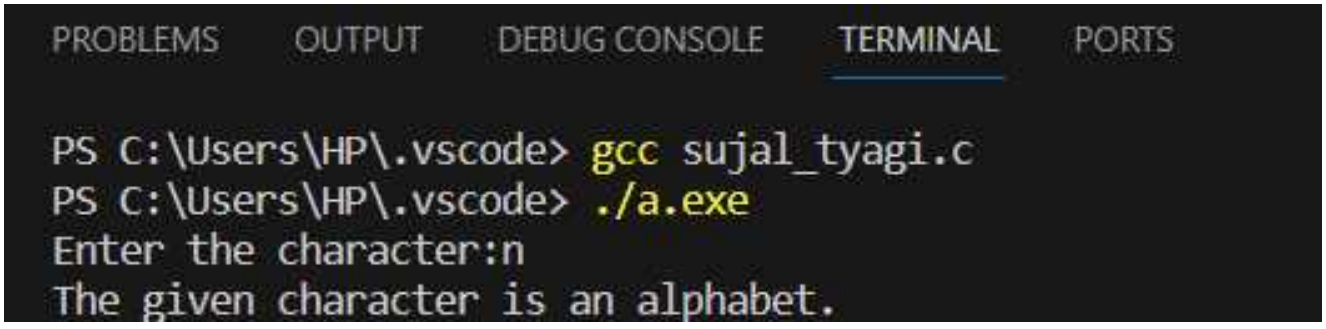
Program 91: Write a C program to check whether year is leap year or not.

```
#include <stdio.h>
int main()
{
   int year;

   printf("Enter the year:");
   scanf("%d", &year);

   if (year % 400 == 0)
   {
      printf("The year is a leap year.");
   }
   else if (year % 100 == 0)
   {
      printf("The year is not a leap year.");
   }
   else if (year % 4 == 0)
   {
      printf("The year is a leap year.");
   }
}
```



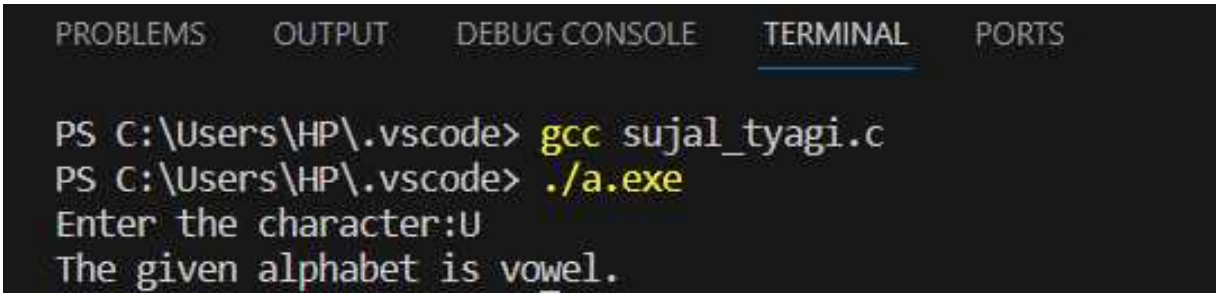
Program 92: Write a C program to check whether alphabet is or not.

```
#include <stdio.h>
int main()
{
   char ch;

   printf("Enter the character:");
   scanf("%c", &ch);

   if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))
   {
      printf("The given character is an alphabet.");
   }
   else
   {
      printf("The given character is not an alphabet.");
   }
}
```



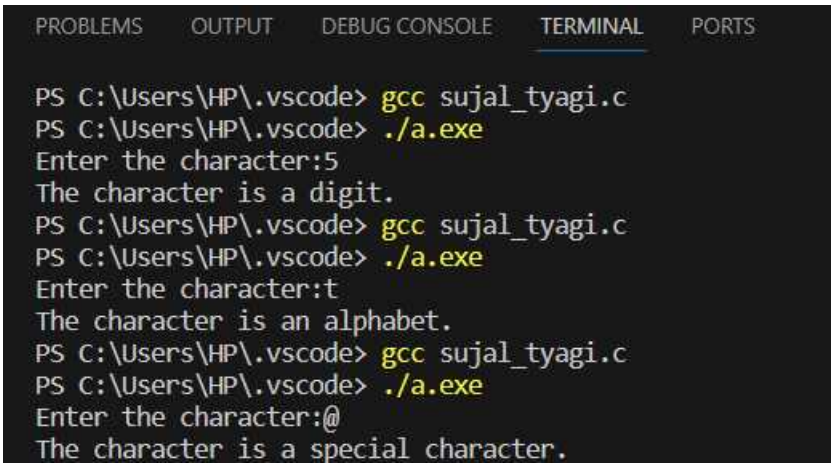
Program 93: Write a C program to check whether an alphabet is vowel or consonant.

```
#include <stdio.h>
int main()
{
   char ch;

   printf("Enter the character:");
   scanf("%c", &ch);

   if (ch == 'a' || ch == 'A' || ch == 'e' || ch == 'E' || ch == 'i' || ch == 'I' || ch == 'o' || ch == 'O' || ch == 'u' || ch == 'U')
   {
      printf("The given alphabet is vowel.");
   }
   else
   {
      printf("The alphabet is a consonant.");
   }
}
```



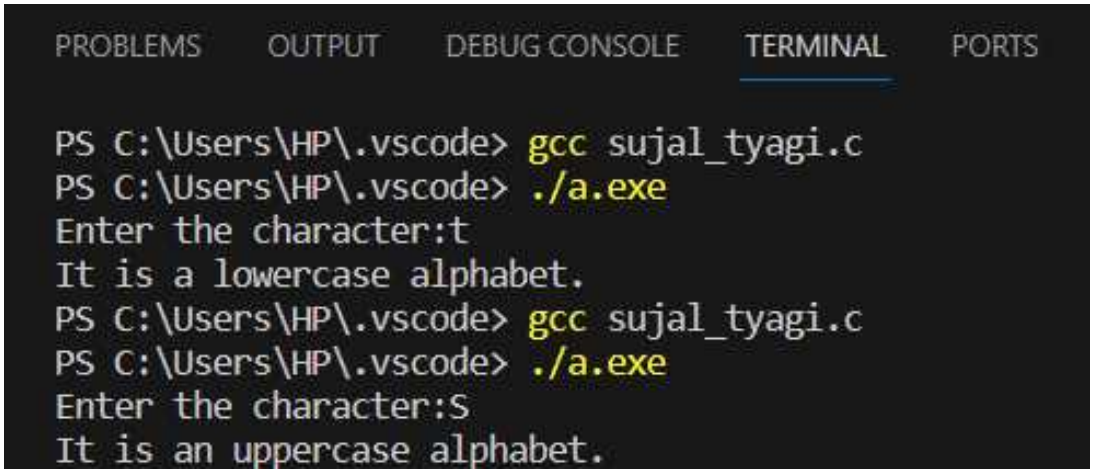
Program 94: Write a C program to input any character and check whether it is an alphabet, digit or a special character.

```
#include <stdio.h>
int main()
{
   char ch;
   printf("Enter the character:");
   scanf("%c", &ch);
if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))
   {
      printf("The character is an alphabet.\n");
   }
   else if (ch >= '0' && ch <= '9')
   {
      printf("The character is a digit.\n");
   }
   else
   {
      printf("The character is a special character.\n");
   }
}
```



Program 95: Write a C program to check whether an alphabet is uppercase or lowercase letter.

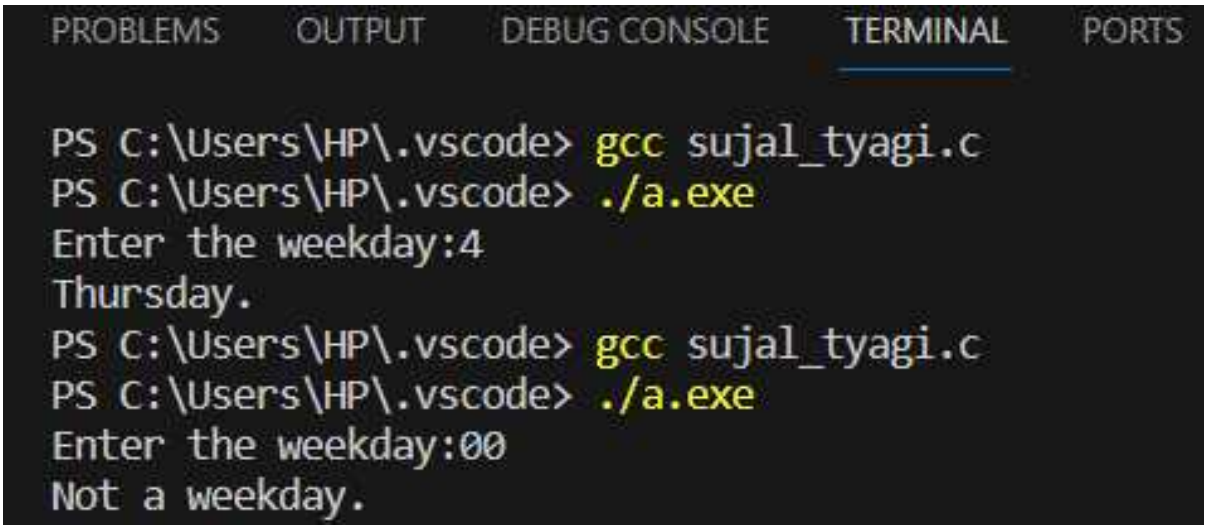
```
#include <stdio.h>
int main()
{
   char ch;

   printf("Enter the character:");
   scanf("%c", &ch);

   if (ch >= 'a' && ch <= 'z')
   {
      printf("It is a lowercase alphabet.");
   }
   else if (ch >= 'A' && ch <= 'Z')
   {
      printf("It is an uppercase alphabet.");
   }
}
```



Program 96: Write a C program to input week and print day.

```
#include <stdio.h>
int main()
{
   int week;

   printf("Enter the weekday:");
   scanf("%d", &week);

   switch (week)
   {
   case 1:
      printf("Monday.\n");
      break;
   case 2:
      printf("Tuesday.\n");
      break;
   case 3:
      printf("Wednesday.\n");
      break;
   case 4:
      printf("Thursday.\n");
      break;
   case 5:
      printf("Friday.\n");
      break;
   case 6:
      printf("Saturday.\n");
      break;
```

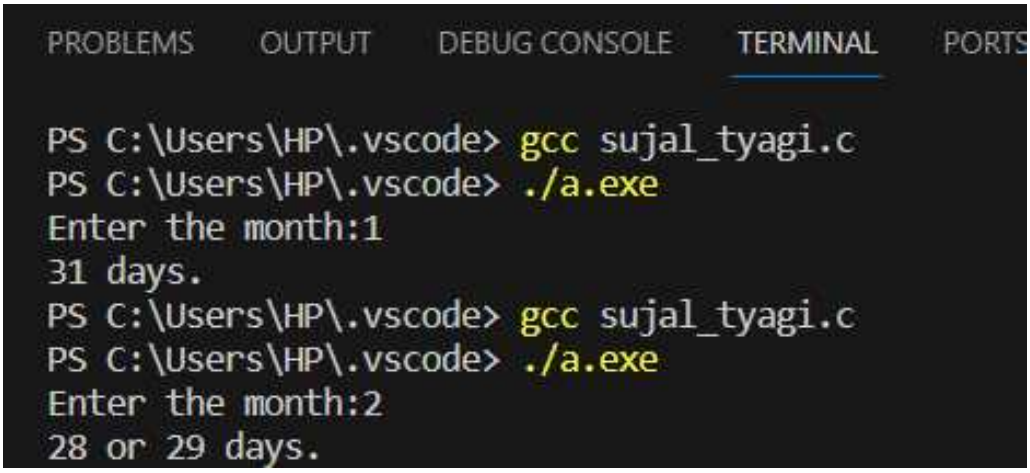
```
    case 7:
        printf("Sunday.\n");
        break;
    default:
        printf("Not a weekday.\n");
    }
}
```



Program 97: Write a C program to input month number and print number of days in that month.

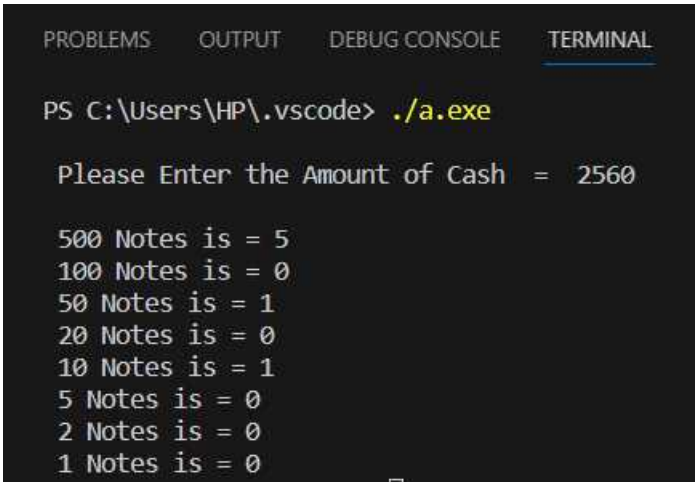
```
#include <stdio.h>
int main()
{
   int month;

   printf("Enter the month:");
   scanf("%d", &month);

   switch (month)
   {
   case 1:
      printf("31 days.\n");
      break;
   case 2:
      printf("28 or 29 days.\n");
      break;
   case 3:
      printf("31 days.\n");
      break;
   case 4:
      printf("30 days.\n");
      break;
   case 5:
      printf("31 days.\n");
      break;
   case 6:
      printf("30 days.\n");
      break;
```

```
    case 7:
        printf("31 days.\n");
        break;
    case 8:
        printf("31 days.\n");
        break;
    case 9:
        printf("30 days.\n");
        break;
    case 10:
        printf("31 days.\n");
        break;
    case 11:
        printf("30 days.\n");
        break;
    case 12:
        printf("31 days.\n");
        break;
    default:
        printf("Invalid month.\n");
    }
}
```



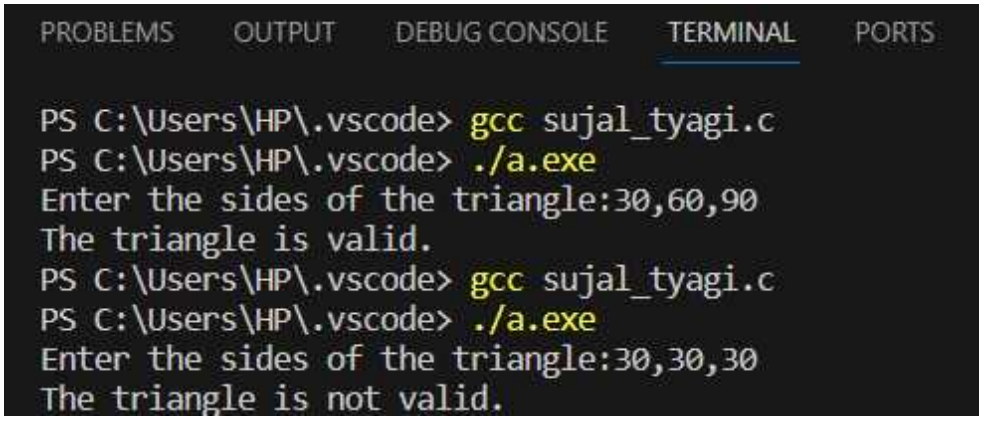
Program 98: Write a C program to count notes in given amount.

```
#include<stdio.h>

int main()
{
 int a[8] = {500, 100, 50, 20, 10, 5, 2, 1};
 int Amount, i, temp;

 printf("\n Please Enter the Amount of Cash  =  ");
 scanf("%d", &Amount);

 temp = Amount;

 for(i = 0; i < 8; i++)
 {
        printf("\n %d Notes is = %d", a[i], temp / a[i]);
        temp = temp % a[i];
 }

 return 0;
}
```



Program 99: Write a C program to input three angles of triangle and check whether it is valid or not.

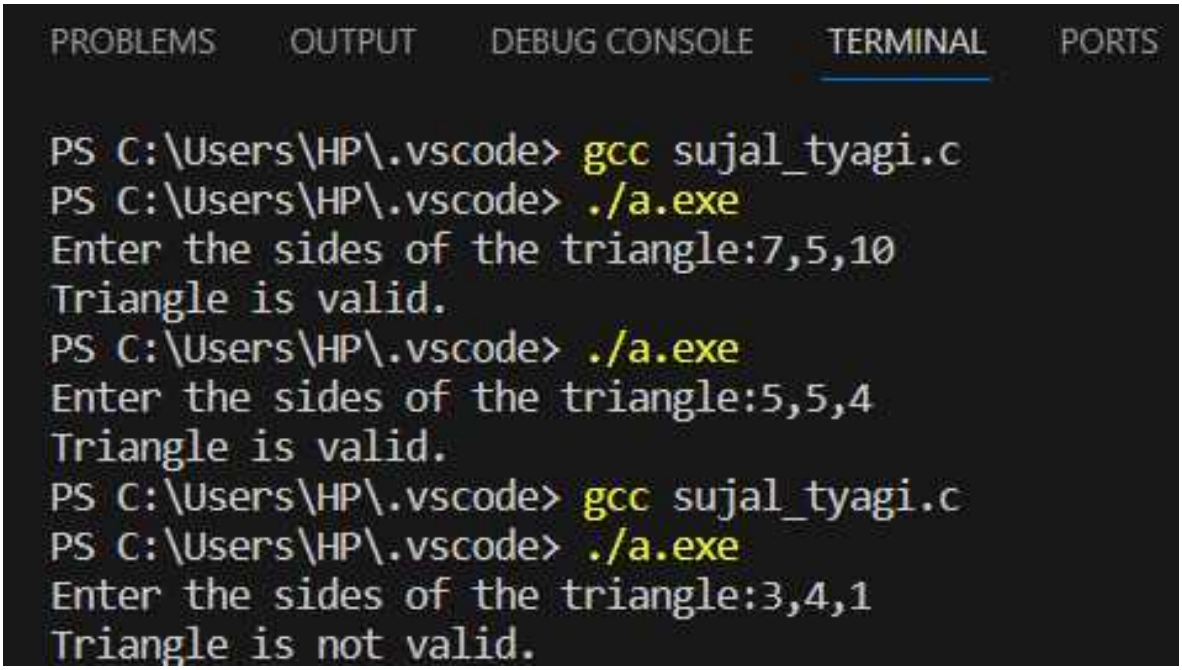
```
#include <stdio.h>

int main()
{
   int s1, s2, s3, sum;
   printf("Enter the sides of the triangle:");
   scanf("%d,%d,%d", &s1, &s2, &s3);

   sum = s1 + s2 + s3;
   if (sum == 180 && s1 > 0 && s2 > 0 && s3 > 0)
   {
      printf("The triangle is valid.\n");
   }
   else
   {
      printf("The triangle is not valid.\n");
   }
   return 0;
}
```



Program 100: Write a C program to input three sides of triangle and check whether it is valid or not.

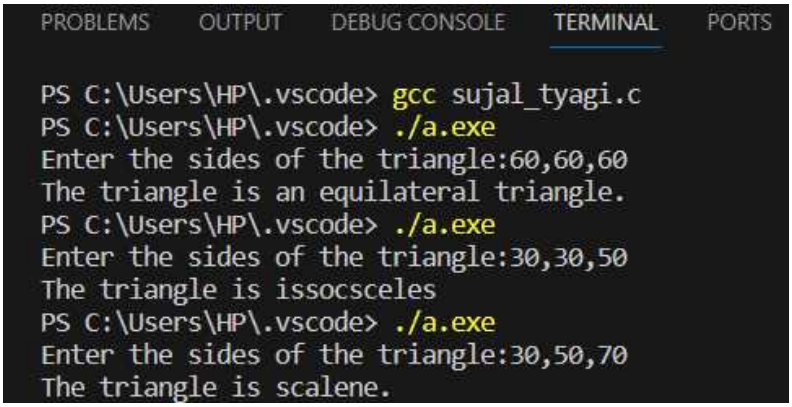
```
#include <stdio.h>

int main()
{
   int s1, s2, s3;
   printf("Enter the sides of the triangle:");
   scanf("%d,%d,%d", &s1, &s2, &s3);

   if ((s1 + s2) > s3)
   {
      if ((s2 + s3) > s1)
      {
         if ((s1 + s3) > s2)
         {
            printf("Triangle is valid.");
         }
         else
         {
            printf("Triangle is not valid.");
         }
      }
      else
      {
         printf("Triangle is not valid.");
      }
   }
   else
```

```
    {
        printf("Triangle is not valid.");
    }

    return 0;
}
```



Program 101: Write a C program to check whether triangle is equilateral, isosceles or scalene.

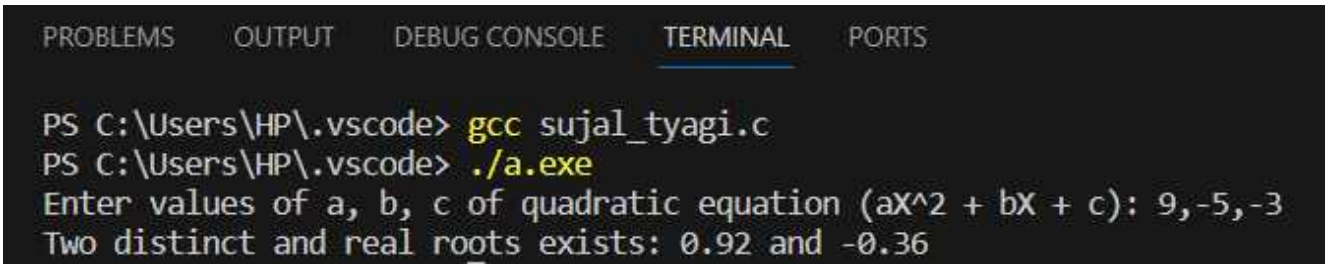
```
#include <stdio.h>

int main()
{
   int s1, s2, s3;
   printf("Enter the sides of the triangle:");
   scanf("%d,%d,%d", &s1, &s2, &s3);
if (s1 == s2 && s2 == s3)
   {
      printf("The triangle is an equilateral triangle.\n");
   }
   else if (s1 == s2 || s1 == s3 || s2 == s3)
   {
      printf("The triangle is issocsceles");
   }
   else if (s1 != s2 != s3)
   {
      printf("The triangle is scalene.\n");
   }
}
```



Program 102: Write a C program to find the roots of a quadratic equation.

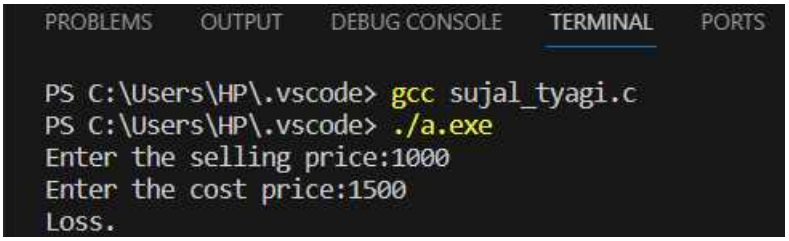
```
#include <stdio.h>
#include <math.h>

int main()
{
   float a, b, c;
   float root1, root2, imaginary;
   float discriminant;

   printf("Enter values of a, b, c of quadratic equation (aX^2 + bX + c): ");
   scanf("%f,%f,%f", &a, &b, &c);

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

   if(discriminant > 0)
   {
      root1 = (-b + sqrt(discriminant)) / (2*a);
      root2 = (-b - sqrt(discriminant)) / (2*a);

      printf("Two distinct and real roots exists: %.2f and %.2f", root1, root2);
   }
   else if(discriminant == 0)
   {
      root1 = root2 = -b / (2 * a);

      printf("Two equal and real roots exists: %.2f and %.2f", root1, root2);
   }
   else if(discriminant < 0)
```

```
    {
        root1 = root2 = -b / (2 * a);
        imaginary = sqrt(-discriminant) / (2 * a);

        printf("Two distinct complex roots exists: %.2f + i%.2f and %.2f - i%.2f",
               root1, imaginary, root2, imaginary);
    }

    return 0;
}
```



Program 103: Write a C program to find the profit or loss.

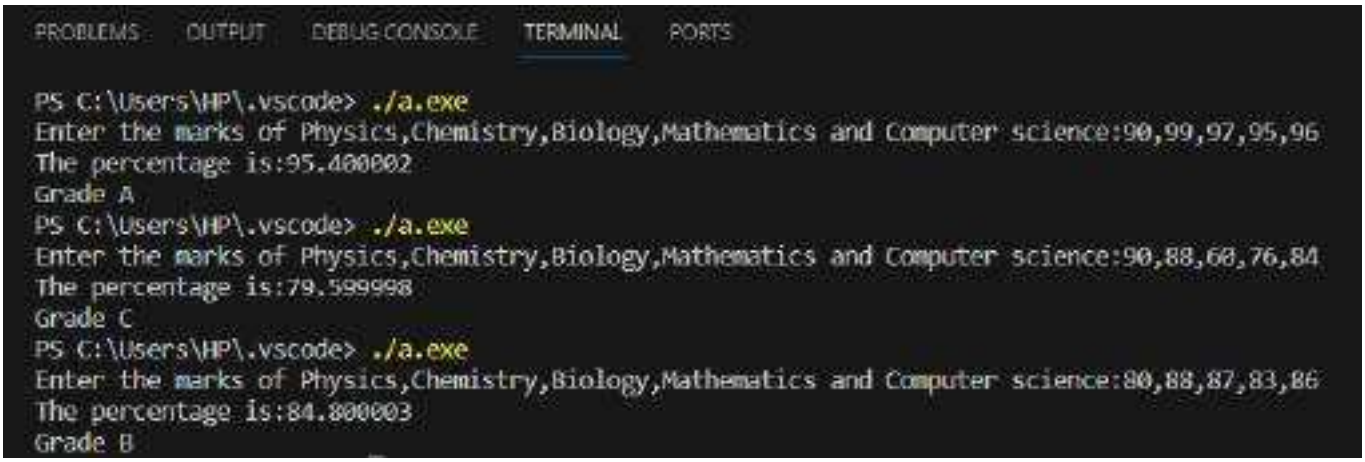
```
#include <stdio.h>
#include <math.h>

int main()
{
   int sellp, costp, amount;

   printf("Enter the cost price:");
   scanf("%d", &costp);
   printf("Enter the selling price:");
   scanf("%d", &sellp);
   amount = sellp - costp;
   if (sellp > costp)
   {
      printf("Profit.\n");
   }
   else if (costp > sellp)
   {
      printf("Loss.\n");
   }
   else
   { printf("Niether profit nor loss.\n");
   }
}
```



Program 104: Write a C program to give the grade.

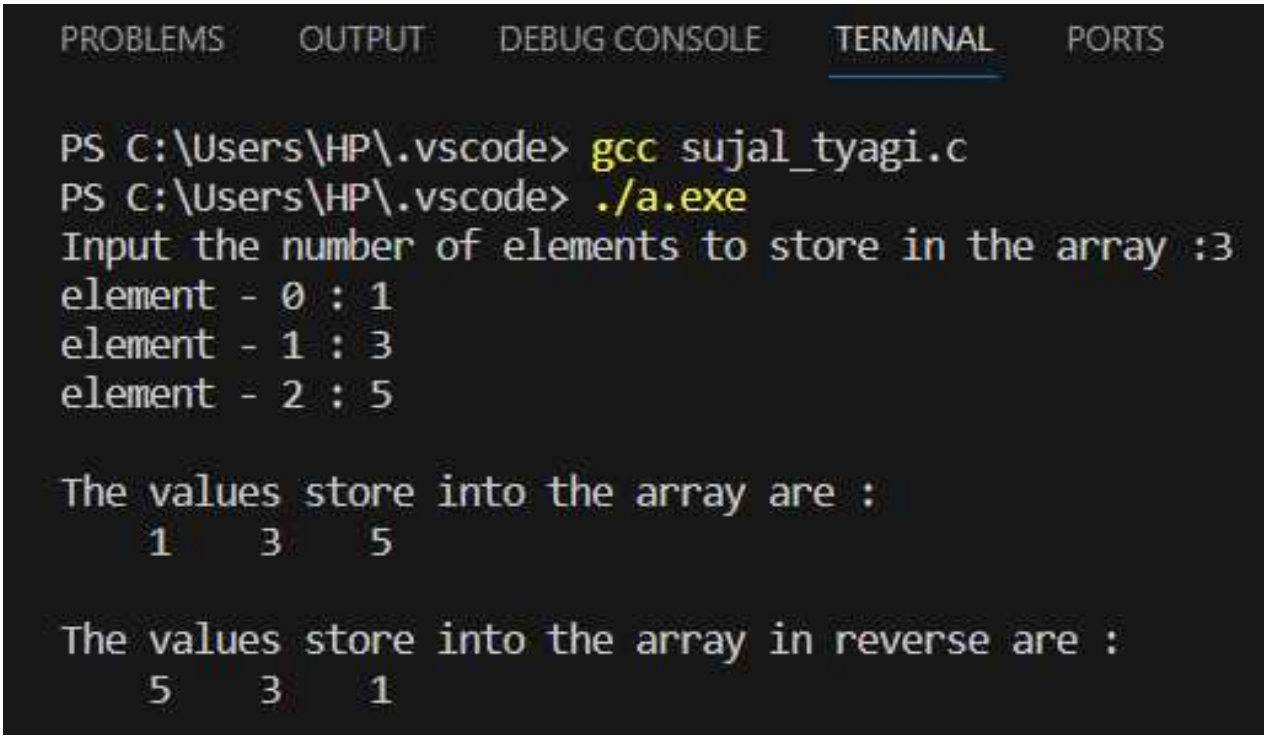
```
#include <stdio.h>

int main()
{
   int p, c, b, m, cs, tot, grade;
   float per;
   printf("Enter the marks of Physics,Chemistry,Biology,Mathematics and
Computer science:");
   scanf("%d,%d,%d,%d,%d", &p, &c, &b, &m, &cs);
   tot = p + c + b + m + cs;
   per = tot / 5.0;
   printf("The percentage is:%f\n", per);
   if (per >= 90)
   {
      printf("Grade A\n");
   }
   else if (per >= 80)
   {
      printf("Grade B\n");
   }
   else if (per >= 70)
   {
      printf("Grade C\n");
   }
   else if (per >= 60)
   {
      printf("Grade D\n");
   }
```

```
    else if (per >= 50)
    {
        printf("Grade E\n");
    }
    else if (per < 40)
    {
        printf("Grade F\n");
    }
    return 0;
}
```



Program 105: Write a C program to read n number of values in an array and display them in reverse order.

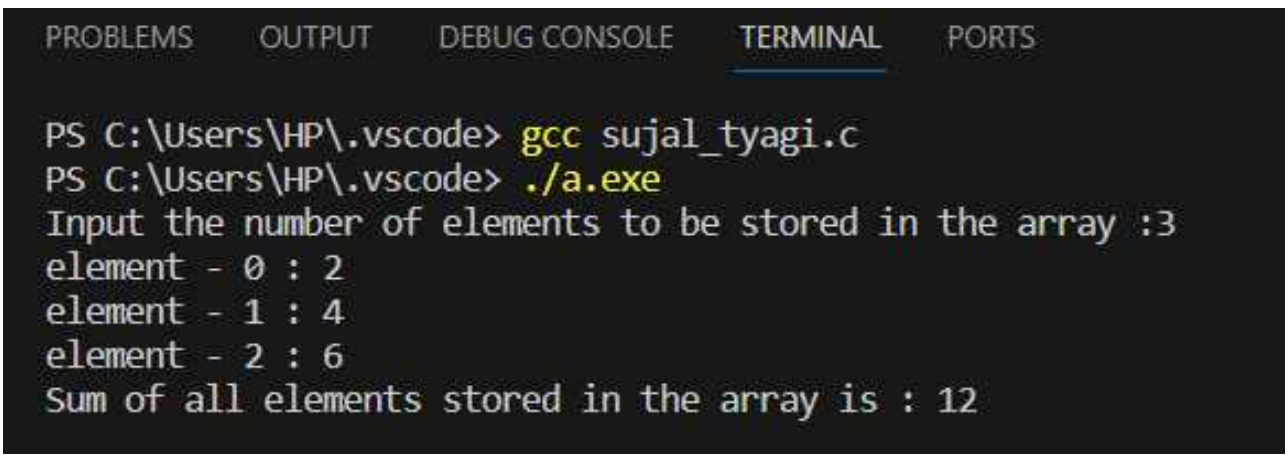
```
#include <stdio.h>

void main()
{
  int i,n,a[100];

  printf("Input the number of elements to store in the array :");
  scanf("%d",&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");
}
```



Program 106: Write a C program to find the sum of all elements in an array.

```
#include <stdio.h>

void main()
{
   int a[100];
   int i, n, sum=0;

      printf("Input the number of elements to be stored in the array :");
      scanf("%d",&n);
      for(i=0;i<n;i++)
       {
            printf("element - %d : ",i);
```

```
            scanf("%d",&a[i]);
          }

   for(i=0; i<n; i++)
   {
      sum += a[i];
   }

   printf("Sum of all elements stored in the array is : %d\n\n", sum);
}
```



Program 107: Write a C program to copy the elements of one array to another array.

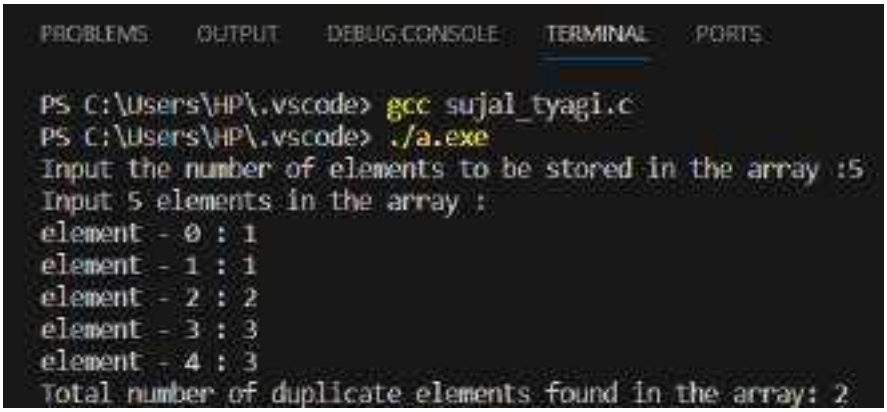
```
#include <stdio.h>

void main()
{
   int arr1[100], arr2[100];
   int i, n;


      printf("\n\nCopy the elements one array into another array :\n");
      printf("-------------------------------------------------\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("element - %d : ",i);
             scanf("%d",&arr1[i]);
           }
   for(i=0; i<n; i++)
   {
      arr2[i] = arr1[i];
   }

   printf("\nThe elements stored in the first array are :\n");
   for(i=0; i<n; i++)
```

```
    {
       printf("% 5d", arr1[i]);
    }

    printf("\n\nThe elements copied into the second array are :\n");
    for(i=0; i<n; i++)
    {
       printf("% 5d", arr2[i]);
    }
            printf("\n\n");
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

------------------------------------------------
Input the number of elements to be stored in the array :3
Input 3 elements in the array :
element - 0 : 5
element - 1 : 7
element - 2 : 9

The elements stored in the first array are :
    5    7    9

The elements copied into the second array are :
    5    7    9
```

Program 108: Write a C program to count the total number of duplicates in an array.

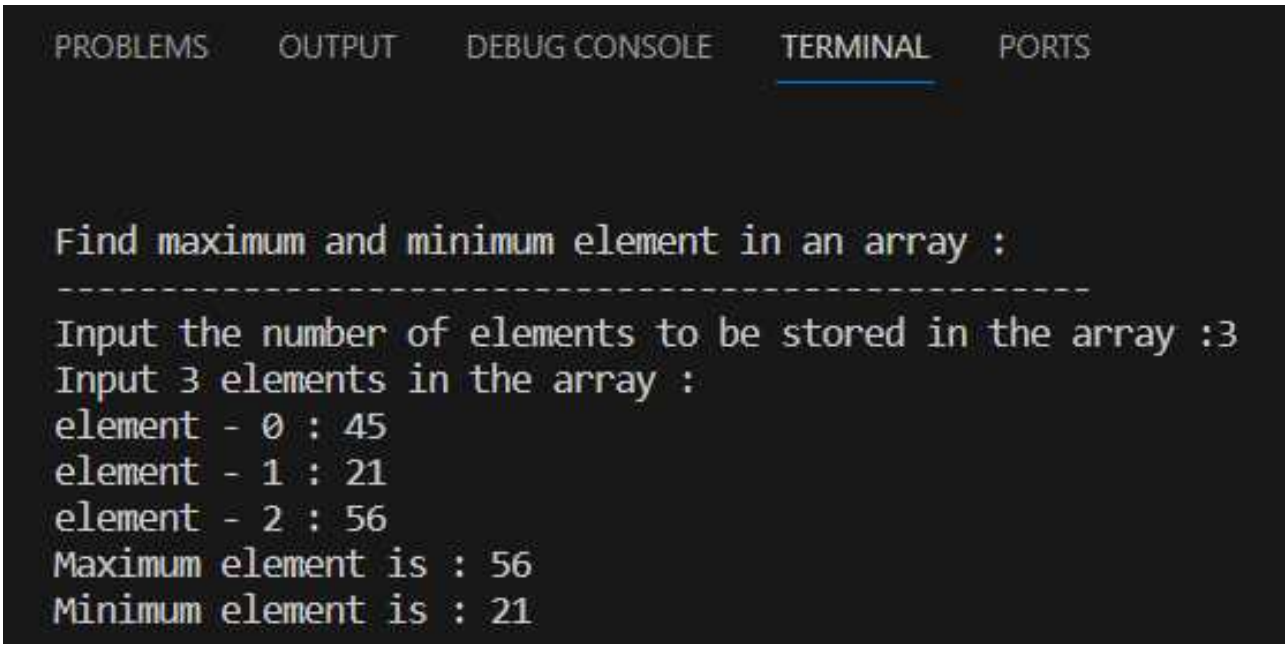
```
#include <stdio.h>
int main()
{
  int arr[100]; int n,mm=1,ctr=0; int i, j;
  printf("Input the number of elements to be stored in the array :");
  scanf("%d",&n);
  printf("Input %d elements in the array :\n",n);
  for(i=0;i<n;i++)
     {
          printf("element - %d : ",i);scanf("%d",&arr[i]);
        }
       for (i = 0; i < n; i++) {
    for (j = i + 1; j < n; j++) {
     if (arr[i] == arr[j]) {
       ctr++;
       break;
  }}}
 printf("Total number of duplicate elements found in the array: %d\n", ctr);
 return 0;
}
```



Program 109: Write a C program to find the maximum and minimum elements in an array.

```
#include <stdio.h>

void main()
{
   int arr1[100];
   int i, mx, mn, n;


     printf("\n\nFind maximum and minimum element in an array :\n");
     printf("-------------------------------------------------\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("element - %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("Maximum element is : %d\n", mx);

    printf("Minimum element is : %d\n\n", mn);

}
```



Program 110: Write a C program to sort the elements of an array in descending order.

```
#include <stdio.h>

void main()
{
   int arr1[100];
   int n, i, j, tmp;



     printf("\n\nsort elements of array in descending order :\n");
     printf("-----------------------------------------\n");

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

     printf("Input %d elements in the array :\n",n);
     for(i=0;i<n;i++)
         {
            printf("element - %d : ",i);
            scanf("%d",&arr1[i]);
          }
   for(i=0; i<n; i++)
   {
      for(j=i+1; j<n; j++)
      {
         if(arr1[i] < arr1[j])
         {
            tmp = arr1[i];
            arr1[i] = arr1[j];
```

```
            arr1[j] = tmp;
          }
        }
    }

     printf("\nElements of array is sorted in descending order:\n");

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

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

sort elements of array in descending order :
------------------------------------------------
Input the size of array : 3
Input 3 elements in the array :
element - 0 : 5
element - 1 : 8
element - 2 : 6

Elements of array is sorted in descending order:
8  6  5
```

Program 111: Write a C program to separate odd and even integers from an array.

```
#include <stdio.h>

void main()
 {
   int arr1[10], arr2[10], arr3[10];
   int i,j=0,k=0,n;



     printf("\n\nSeparate odd and even integers in separate arrays:\n");
     printf("---------------------------------------------------\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("element - %d : ",i);
            scanf("%d",&arr1[i]);
          }

   for(i=0;i<n;i++)
   {
       if (arr1[i]%2 == 0)
       {
          arr2[j] = arr1[i];
          j++;
       }
```

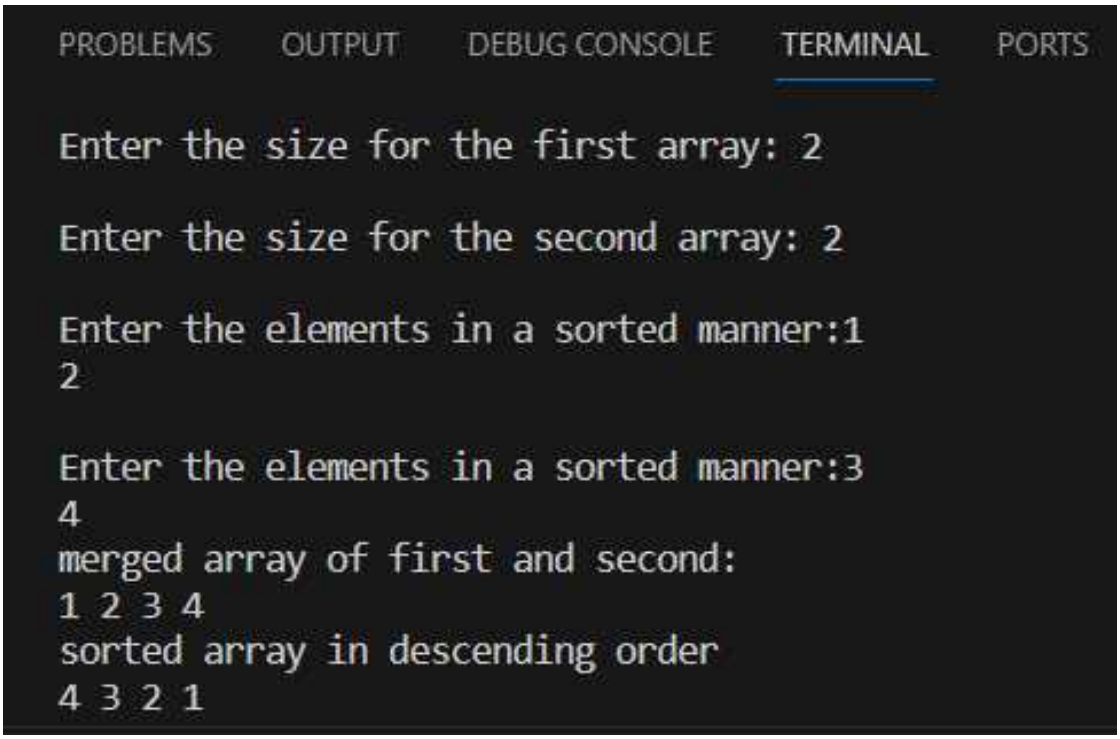
```
        else
        {
          arr3[k] = arr1[i];
          k++;
        }
    }

    printf("\nThe Even elements are : \n");
    for(i=0;i<j;i++)
    {
        printf("%d ",arr2[i]);
    }

    printf("\nThe Odd elements are :\n");
    for(i=0;i<k;i++)
    {
        printf("%d ", arr3[i]);
    }
    printf("\n\n");
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

-----------------------------------------------------
Input the number of elements to be stored in the array :3
Input 3 elements in the array :
element - 0 : 1
element - 1 : 3
element - 2 : 4

The Even elements are :
4
The Odd elements are :
1 3
```

Program 112: Write a C program to merge two arrays of same size sorted in descending order.

```
#include <stdio.h>
int main() {
  int size1, size2, size3;
  printf("\nEnter the size for the first array: ");
  scanf("%d", & size1);
  printf("\nEnter the size for the second array: ");
  scanf("%d", & size2);

  size3 = size1 + size2;
  printf("\nEnter the elements in a sorted manner:");

  int array1[size1], array2[size2], array3[size3];
  for (int i = 0; i < size1; i++) {
    scanf("%d", & array1[i]);
    array3[i] = array1[i];
  }
  int k = size1;
  printf("\nEnter the elements in a sorted manner:");
  for (int i = 0; i < size2; i++) {
    scanf("%d", & array2[i]);
    array3[k] = array2[i];
    k++;
  }
  printf("merged array of first and second:\n");
  for (int i = 0; i < size3; i++)
    printf("%d ", array3[i]);

  printf("\nsorted array in descending order\n");
```

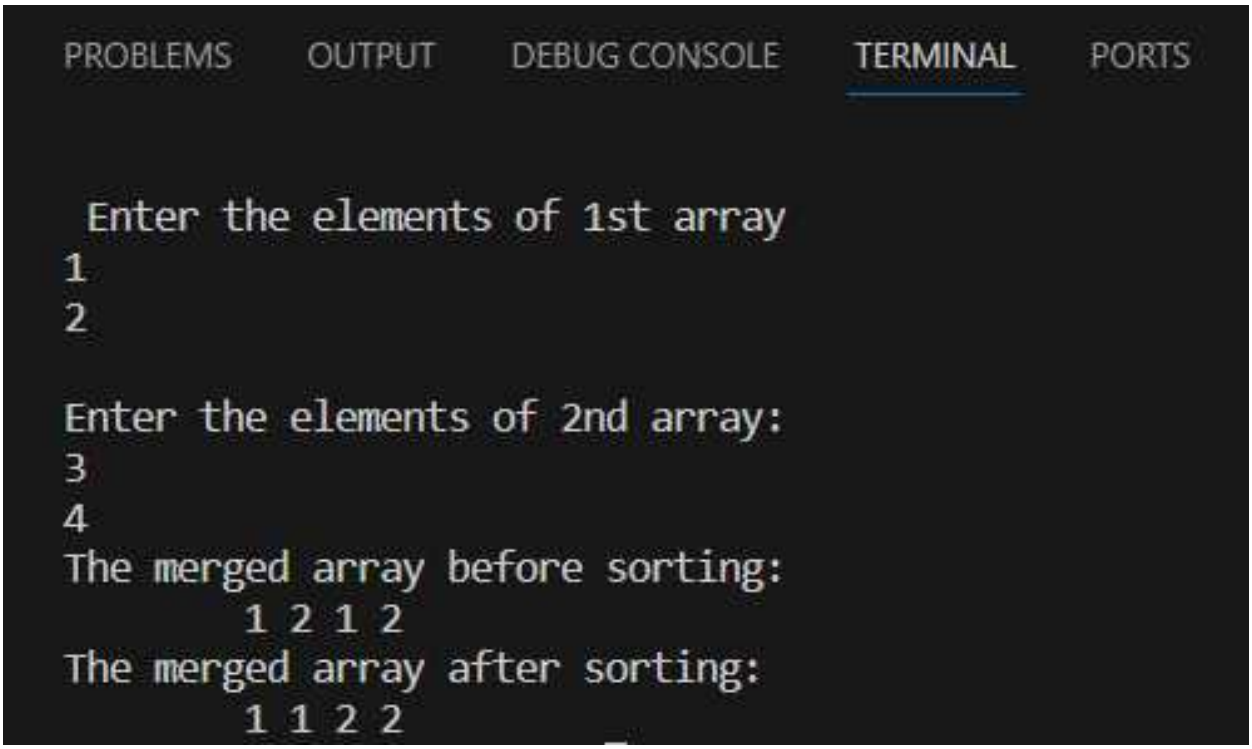
```
  for (int i = 0; i < size3; i++) {
    int temp;
    for (int j = i + 1; j < size3; j++) {
      if (array3[i] < array3[j]) {
        temp = array3[i];
        array3[i] = array3[j];
        array3[j] = temp;
      }
    }
  }
  for (int i = 0; i < size3; i++) {
    printf("%d ", array3[i]);
  }
  return 0;
}
```



Program 113: Write a C program to merge two arrays of same size sorted in ascending order.

```
#include <stdio.h>


int main() {
  int s1, s2, s3;
  printf("\n Enter the size of 1st array  ");
  scanf("%d", & s1);
  printf("\n Enter the size of 2nd array ");
  scanf("%d", & s2);



  s3 = s1 + s2;
  printf("\n Enter the elements of 1st array\n");
  int arr1[s1], arr2[s2], arr3[s3];
  for (int i = 0; i < s1; i++) {
    scanf("%d", & arr1[i]);
    arr3[i] = arr1[i];
  }
  int k = s1;
  printf("\nEnter the elements of 2nd array:\n");
  for (int i = 0; i < s2; i++)
  {
    scanf("%d", & arr2[i]);
    arr3[k] = arr3[i];
    k++;
  }
  printf("The merged array before sorting:\n\t");
  for (int i = 0; i < s3; i++)
```

```
    printf("%d ", arr3[i]);


  printf("\nThe merged array after sorting:\n\t");
  for (int i = 0; i < s3; i++)
  {
    int tem;
    for (int j = i + 1; j < s3; j++) {
      if (arr3[i] > arr3[j]) {
        tem = arr3[i];
        arr3[i] = arr3[j];
        arr3[j] = tem;
      }
    }
  }

  for (int i = 0; i < s3; i++)
  {
    printf("%d ", arr3[i]);
  }
}
```


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Enter the elements of 1st array
1
2
Enter the elements of 2nd array:
3
4
The merged array before sorting:
1 2 1 2
The merged array after sorting:
1 1 2 2