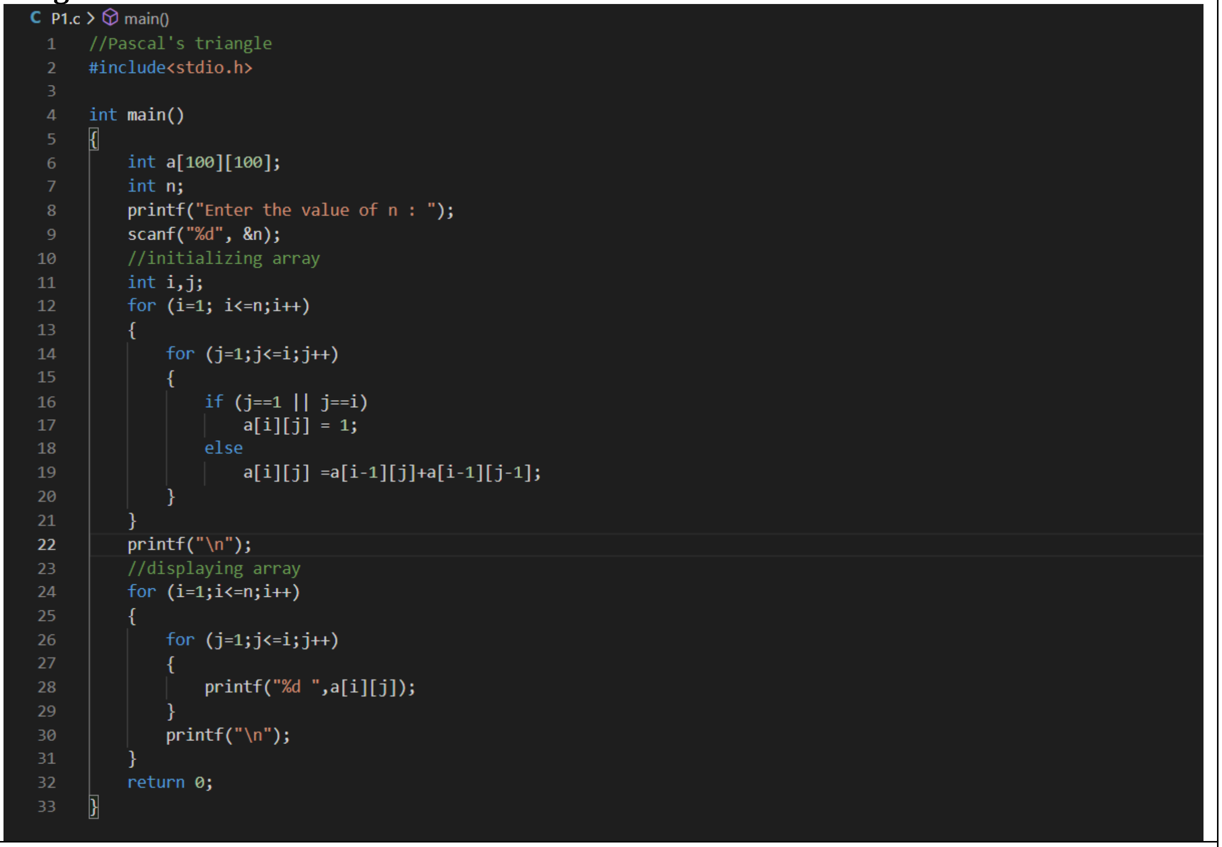
| Name: Renita Kurian | SRN: PES1UG20CS331 | Section: N |
|---|---|---|
| | Date: 9/06/21 | Week Number: 6 |

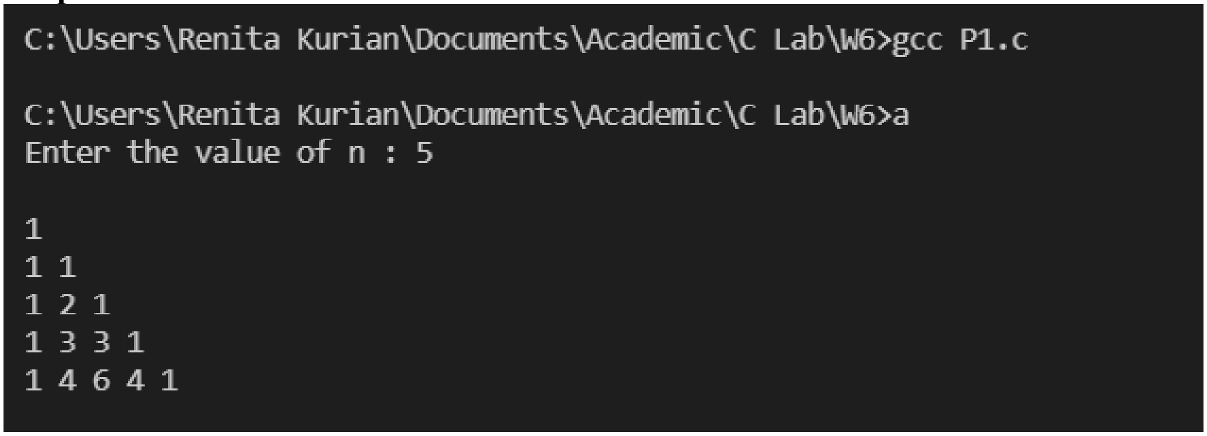
**1** Write a C program to generate Pascal triangle using a two-dimensional array

**Program:**



```c
//Pascal's triangle
#include<stdio.h>

int main()
{
    int a[100][100];
    int n;
    printf("Enter the value of n : ");
    scanf("%d", &n);
    //initializing array
    int i,j;
    for (i=1; i<=n;i++)
    {
        for (j=1;j<=i;j++)
        {
            if (j==1 || j==i)
                a[i][j] = 1;
            else
                a[i][j] =a[i-1][j]+a[i-1][j-1];
        }
    }
    printf("\n");
    //displaying array
    for (i=1;i<=n;i++)
    {
        for (j=1;j<=i;j++)
        {
            printf("%d ",a[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

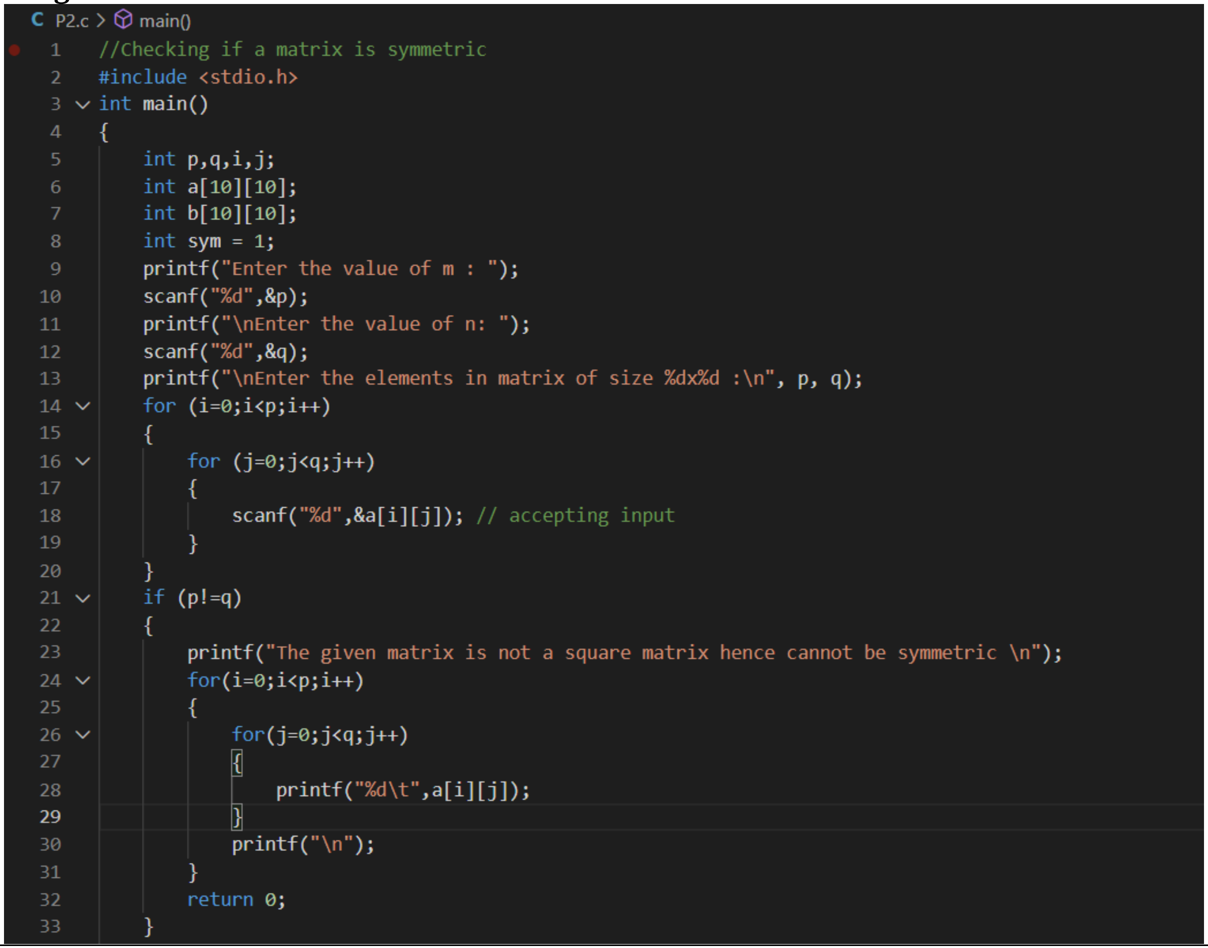
**Output Screenshot:**



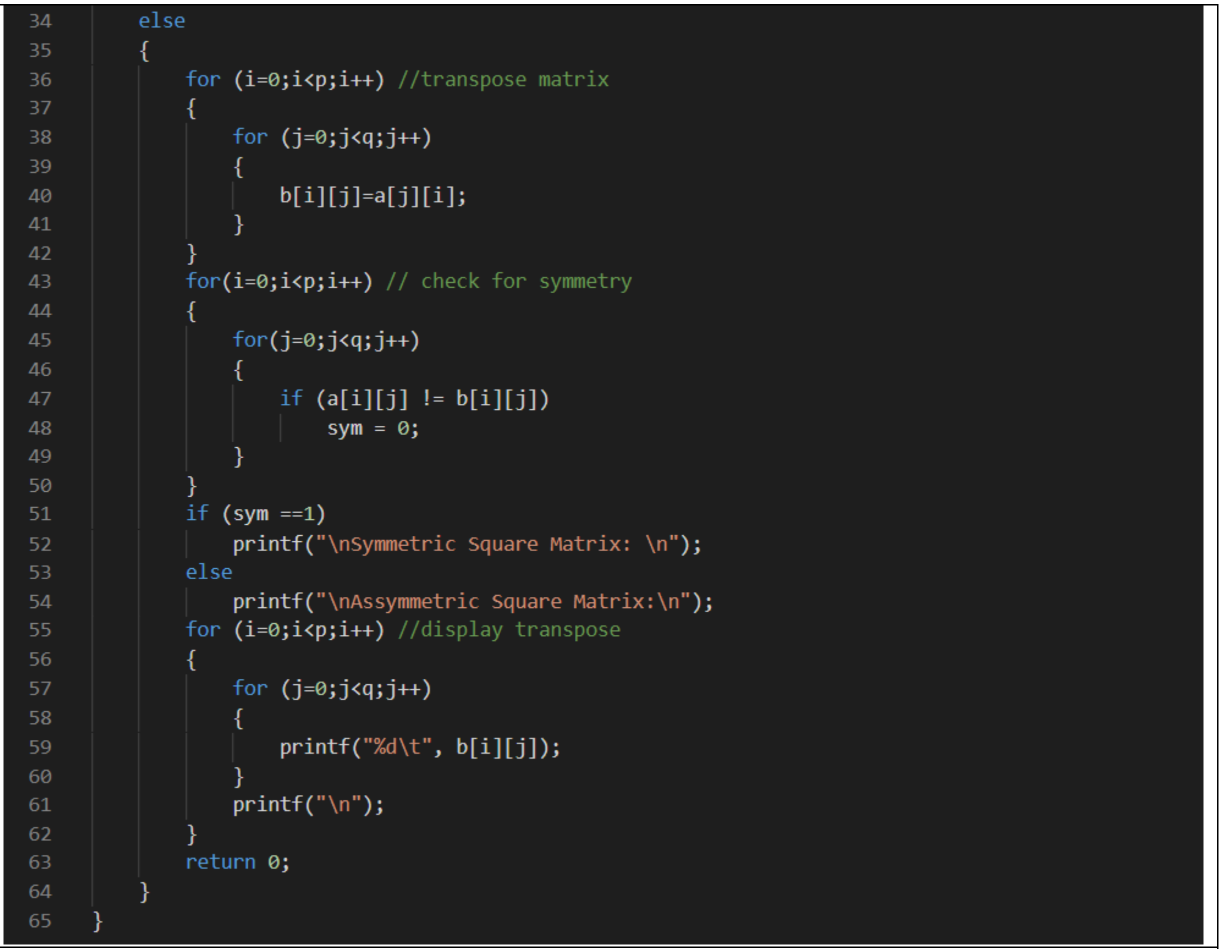
```
C:\Users\Renita Kurian\Documents\Academic\C Lab\W6>gcc P1.c

C:\Users\Renita Kurian\Documents\Academic\C Lab\W6>a
Enter the value of n : 5

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

| | |
|---|---|
| **2** | Write a C program to read elements in a matrix and check whether the given matrix is symmetric matrix or not. |

**Program:**

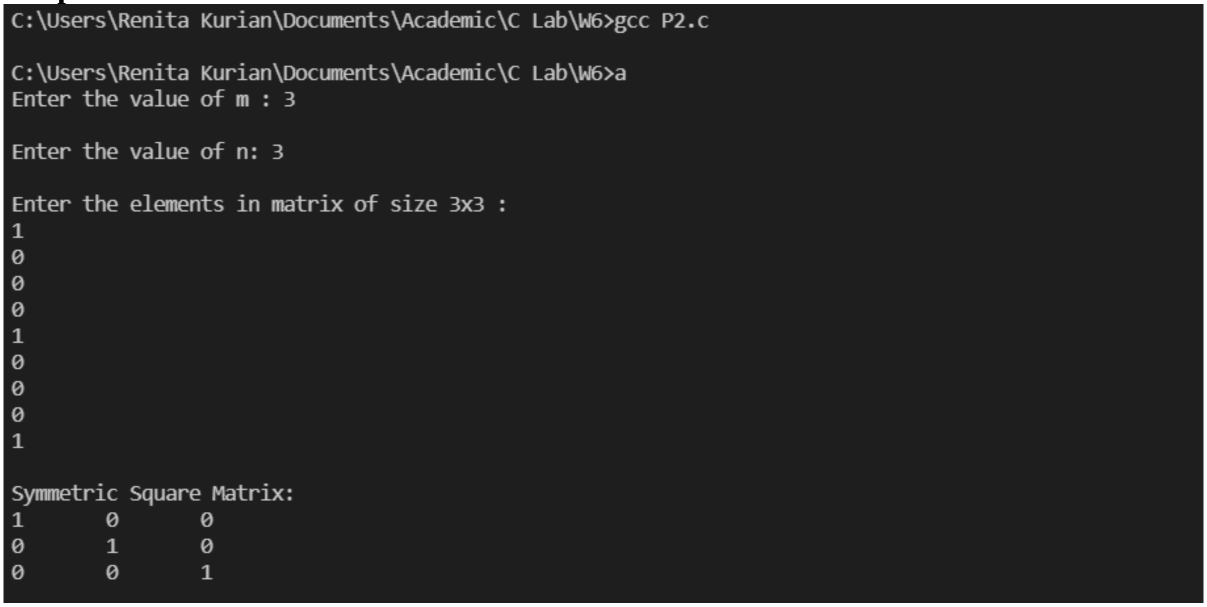


```c
//Checking if a matrix is symmetric
#include <stdio.h>
int main()
{
    int p,q,i,j;
    int a[10][10];
    int b[10][10];
    int sym = 1;
    printf("Enter the value of m : ");
    scanf("%d",&p);
    printf("\nEnter the value of n: ");
    scanf("%d",&q);
    printf("\nEnter the elements in matrix of size %dx%d :\n", p, q);
    for (i=0;i<p;i++)
    {
        for (j=0;j<q;j++)
        {
            scanf("%d",&a[i][j]); // accepting input
        }
    }
    if (p!=q)
    {
        printf("The given matrix is not a square matrix hence cannot be symmetric \n");
        for(i=0;i<p;i++)
        {
            for(j=0;j<q;j++)
            {
                printf("%d\t",a[i][j]);
            }
            printf("\n");
        }
        return 0;
    }
```



```c
    else
    {
        for (i=0;i<p;i++) //transpose matrix
        {
            for (j=0;j<q;j++)
            {
                b[i][j]=a[j][i];
            }
        }
        for(i=0;i<p;i++) // check for symmetry
        {
            for(j=0;j<q;j++)
            {
                if (a[i][j] != b[i][j])
                    sym = 0;
            }
        }
        if (sym ==1)
            printf("\nSymmetric Square Matrix: \n");
        else
            printf("\nAssymmetric Square Matrix:\n");
        for (i=0;i<p;i++) //display transpose
        {
            for (j=0;j<q;j++)
            {
                printf("%d\t", b[i][j]);
            }
            printf("\n");
        }
        return 0;
    }
}
```

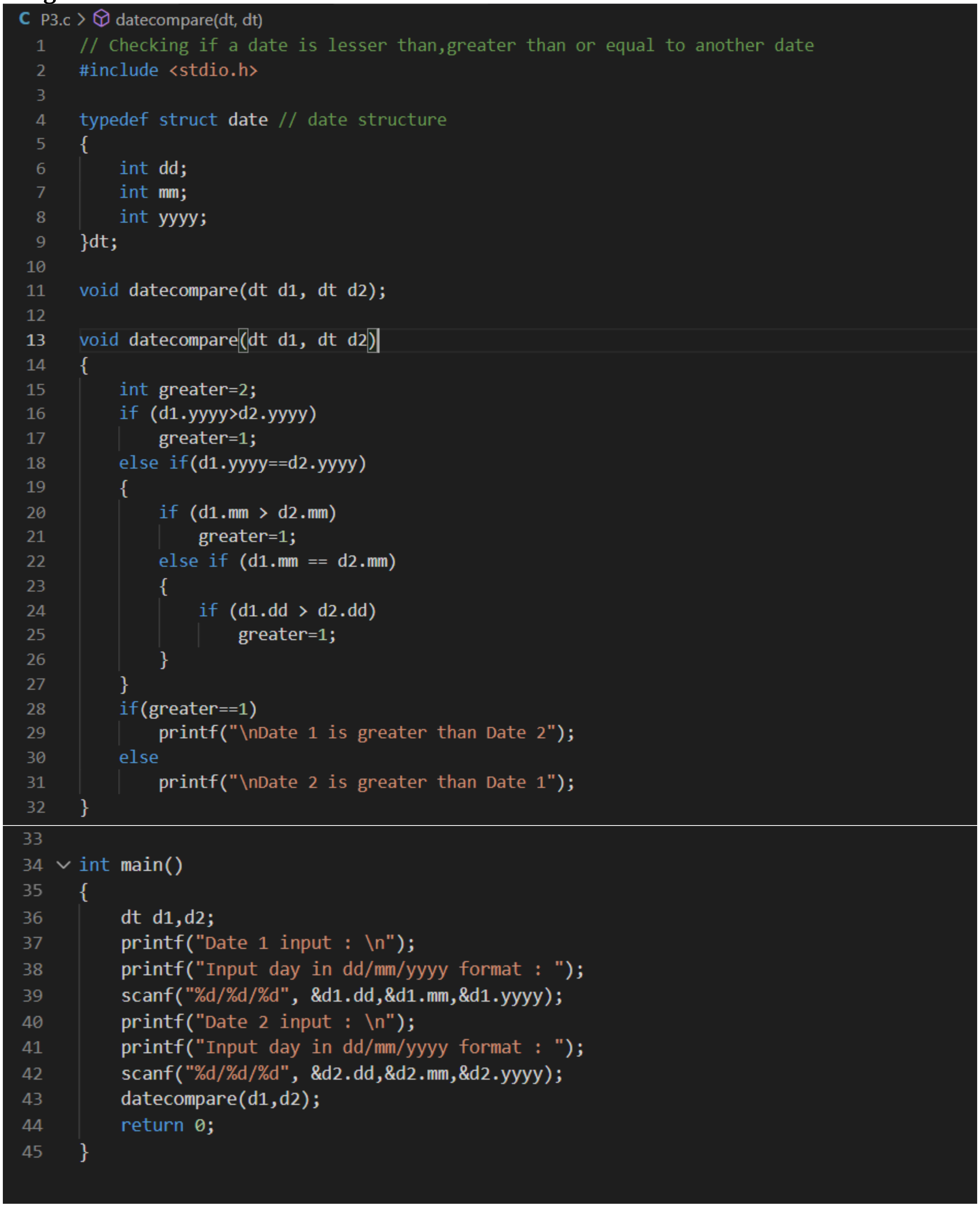
**Output Screenshot:**



```
C:\Users\Renita Kurian\Documents\Academic\C Lab\W6>gcc P2.c

C:\Users\Renita Kurian\Documents\Academic\C Lab\W6>a
Enter the value of m : 3

Enter the value of n: 3

Enter the elements in matrix of size 3x3 :
1
0
0
0
1
0
0
0
1

Symmetric Square Matrix:
1       0       0
0       1       0
0       0       1
```

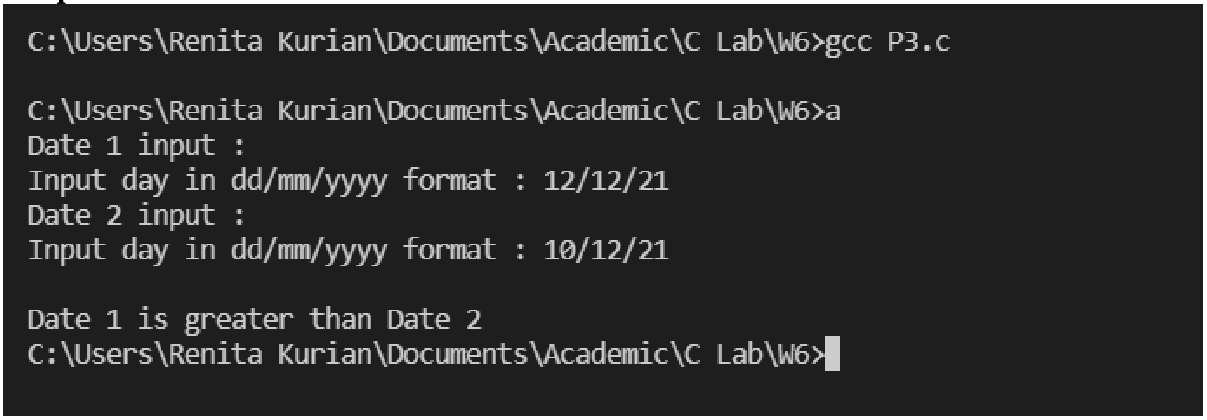
| | |
|---|---|
| **3** | Write a C program to compare 2 dates and print appropriate message using structures |

**Program:**



```c
// Checking if a date is lesser than,greater than or equal to another date
#include <stdio.h>

typedef struct date // date structure
{
    int dd;
    int mm;
    int yyyy;
}dt;

void datecompare(dt d1, dt d2);

void datecompare(dt d1, dt d2)
{
    int greater=2;
    if (d1.yyyy>d2.yyyy)
        greater=1;
    else if(d1.yyyy==d2.yyyy)
    {
        if (d1.mm > d2.mm)
            greater=1;
        else if (d1.mm == d2.mm)
        {
            if (d1.dd > d2.dd)
                greater=1;
        }
    }
    if(greater==1)
        printf("\nDate 1 is greater than Date 2");
    else
        printf("\nDate 2 is greater than Date 1");
}

int main()
{
    dt d1,d2;
    printf("Date 1 input : \n");
    printf("Input day in dd/mm/yyyy format : ");
    scanf("%d/%d/%d", &d1.dd,&d1.mm,&d1.yyyy);
    printf("Date 2 input : \n");
    printf("Input day in dd/mm/yyyy format : ");
    scanf("%d/%d/%d", &d2.dd,&d2.mm,&d2.yyyy);
    datecompare(d1,d2);
    return 0;
}
```

**Output Screenshot:**

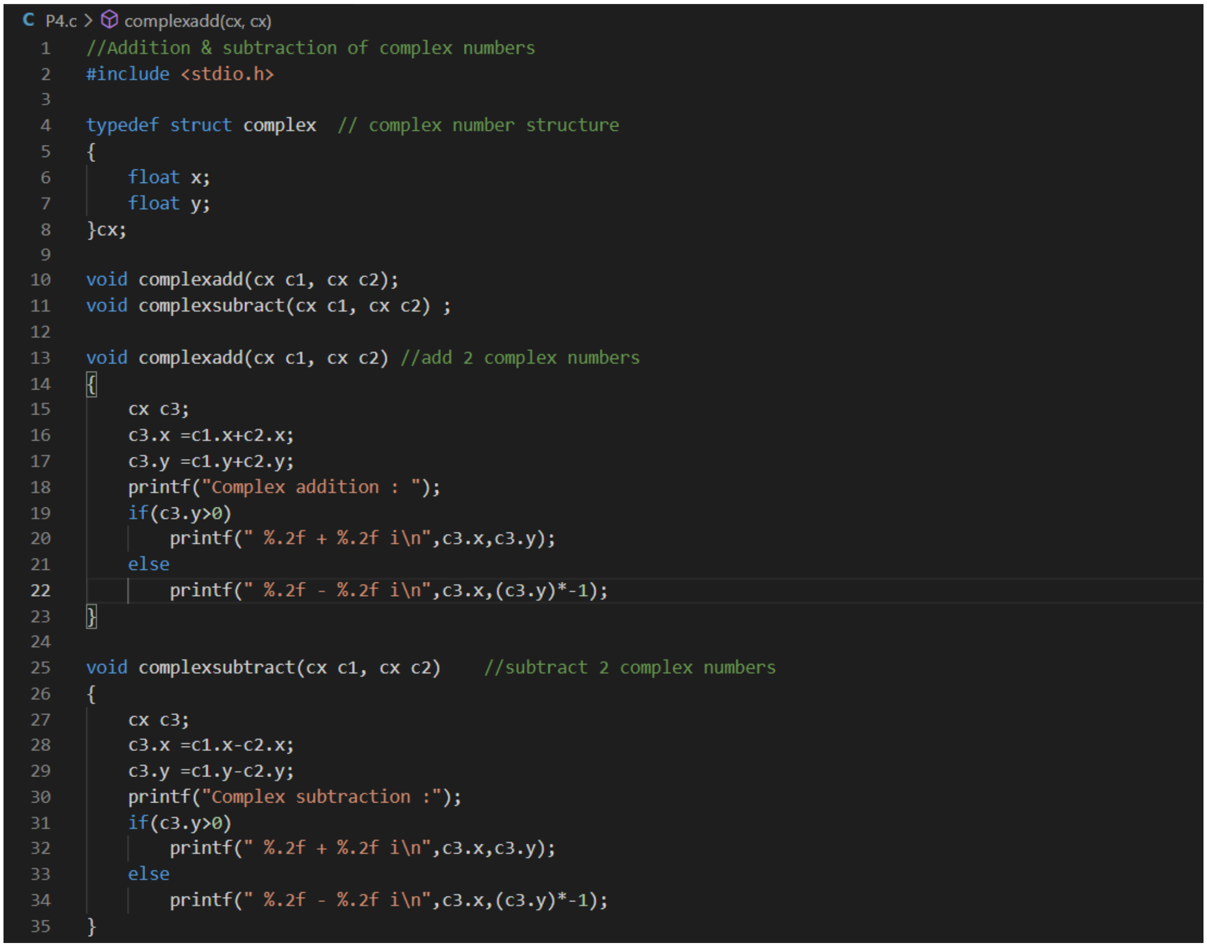


```
C:\Users\Renita Kurian\Documents\Academic\C Lab\W6>gcc P3.c

C:\Users\Renita Kurian\Documents\Academic\C Lab\W6>a
Date 1 input :
Input day in dd/mm/yyyy format : 12/12/21
Date 2 input :
Input day in dd/mm/yyyy format : 10/12/21

Date 1 is greater than Date 2
C:\Users\Renita Kurian\Documents\Academic\C Lab\W6>
```

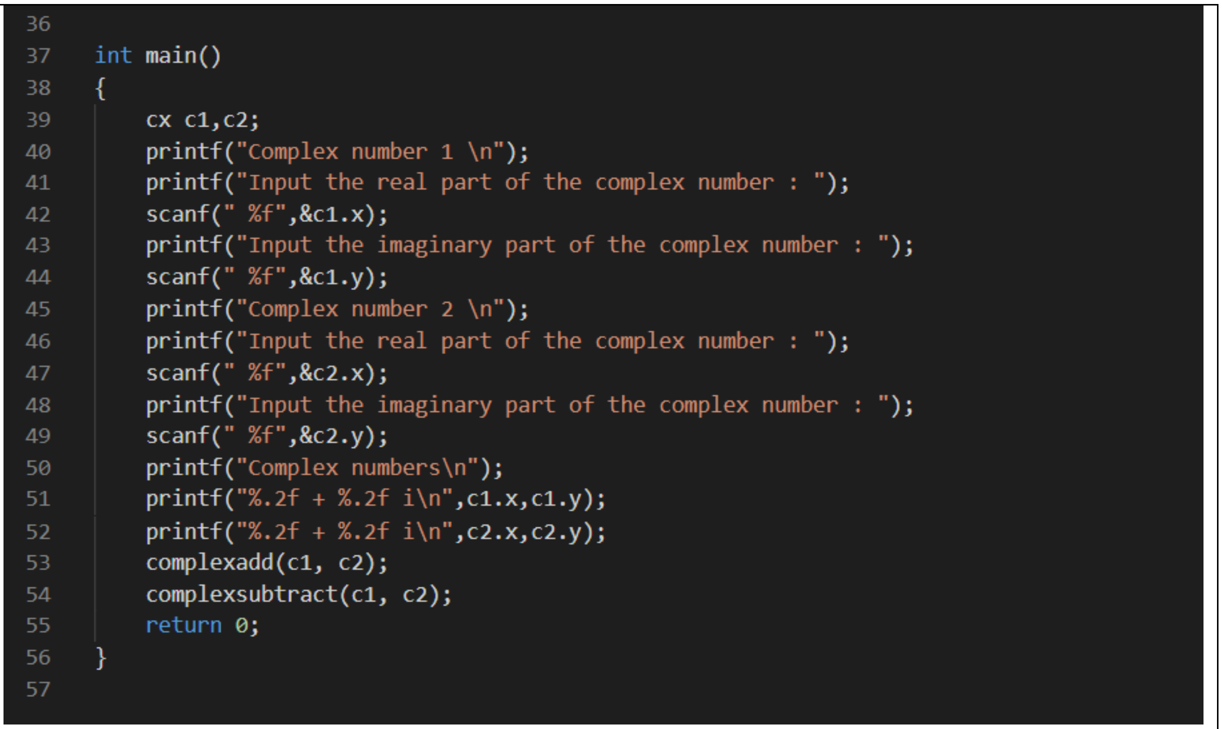
4

Write a C Program to Add and subtract two Complex Numbers by Passing Structure to a Function.

Program:

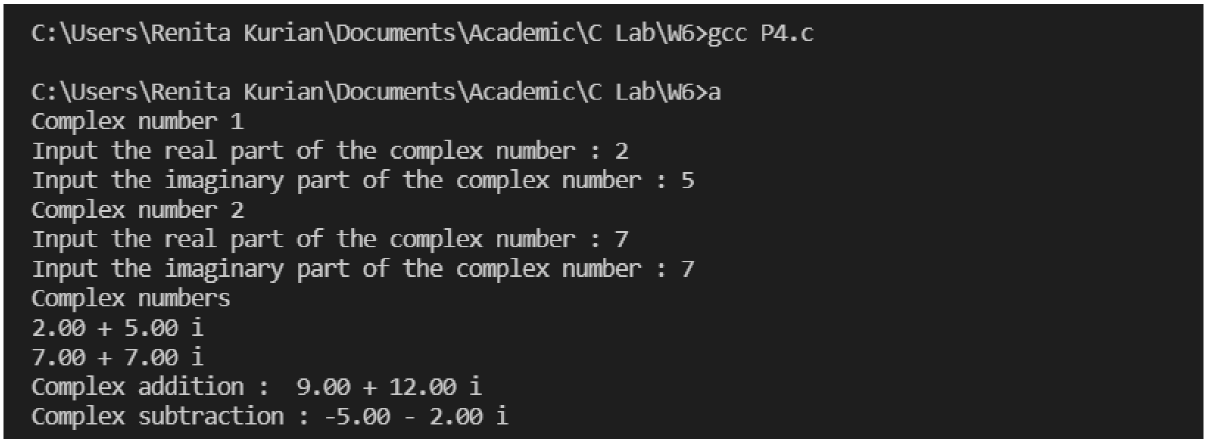


```c
//Addition & subtraction of complex numbers
#include <stdio.h>

typedef struct complex  // complex number structure
{
    float x;
    float y;
}cx;

void complexadd(cx c1, cx c2);
void complexsubract(cx c1, cx c2) ;

void complexadd(cx c1, cx c2) //add 2 complex numbers
{
    cx c3;
    c3.x =c1.x+c2.x;
    c3.y =c1.y+c2.y;
    printf("Complex addition : ");
    if(c3.y>0)
        printf(" %.2f + %.2f i\n",c3.x,c3.y);
    else
        printf(" %.2f - %.2f i\n",c3.x,(c3.y)*-1);
}

void complexsubtract(cx c1, cx c2)    //subtract 2 complex numbers
{
    cx c3;
    c3.x =c1.x-c2.x;
    c3.y =c1.y-c2.y;
    printf("Complex subtraction :");
    if(c3.y>0)
        printf(" %.2f + %.2f i\n",c3.x,c3.y);
    else
        printf(" %.2f - %.2f i\n",c3.x,(c3.y)*-1);
}
```



```c

int main()
{
    cx c1,c2;
    printf("Complex number 1 \n");
    printf("Input the real part of the complex number : ");
    scanf(" %f",&c1.x);
    printf("Input the imaginary part of the complex number : ");
    scanf(" %f",&c1.y);
    printf("Complex number 2 \n");
    printf("Input the real part of the complex number : ");
    scanf(" %f",&c2.x);
    printf("Input the imaginary part of the complex number : ");
    scanf(" %f",&c2.y);
    printf("Complex numbers\n");
    printf("%.2f + %.2f i\n",c1.x,c1.y);
    printf("%.2f + %.2f i\n",c2.x,c2.y);
    complexadd(c1, c2);
    complexsubtract(c1, c2);
    return 0;
}

```

Output Screenshot:



```
C:\Users\Renita Kurian\Documents\Academic\C Lab\W6>gcc P4.c

C:\Users\Renita Kurian\Documents\Academic\C Lab\W6>a
Complex number 1
Input the real part of the complex number : 2
Input the imaginary part of the complex number : 5
Complex number 2
Input the real part of the complex number : 7
Input the imaginary part of the complex number : 7
Complex numbers
2.00 + 5.00 i
7.00 + 7.00 i
Complex addition :  9.00 + 12.00 i
Complex subtraction : -5.00 - 2.00 i
```