

ASSIGNMENT-7

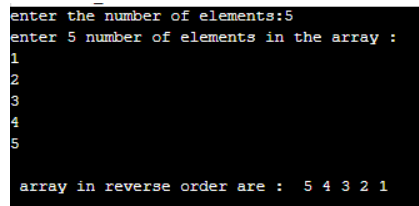
Write a C Program for the following problem statements .

1. Read n number of values in an array and display it in reverse order.

```
#include <stdio.h>
void main()
{
    int i,num,arr[50];
    printf("enter the number of elements:");
    scanf("%d",&num);

    printf("enter %d number of elements in the array :\n",num);
    for(i=0;i<num;i++)
    {
        scanf("%d",&arr[i]);
    }
    printf("\n array in reverse order are : ");
    for(i=num-1;i>=0;i--)
    {
        printf("%2d",arr[i]);
    }
}
```

Output:-



```
enter the number of elements:5
enter 5 number of elements in the array :
1
2
3
4
5

array in reverse order are :  5 4 3 2 1
```

2. Find the sum of all elements of the array.

```
#include <stdio.h>
int main()
{
    int arr[30],i,num,sum=0;

    printf("enter the number of elements in the array :");
    scanf("%d",&num);

    printf("Input elements in the array :\n");
    for(i=0;i<num;i++)
    {
```

```

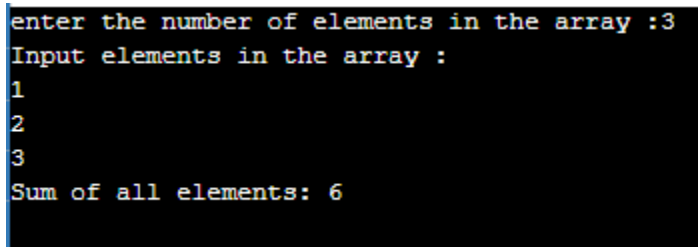
        scanf("%d",&arr[i]);
    }

    for(i=0; i<num; i++)
    {
        sum=sum+arr[i];
    }

    printf("Sum of all elements: %d\n", sum);
    return 0;
}

```

Output:-



```

enter the number of elements in the array :3
Input elements in the array :
1
2
3
Sum of all elements: 6

```

3. Copy the elements of one array into another array.

#include <stdio.h>

```

void main()
{
    int arr1[30], arr2[30],i,num;
    printf("enter the number of elements: ");
    scanf("%d",&num);

    for(i=0;i<num;i++)
    {
        scanf("%d",&arr1[i]);
    }
    for(i=0; i<num; i++)
    {
        arr2[i] = arr1[i];
    }
    printf("\nelements in the first array :\n");
    for(i=0; i<num; i++)
    {
        printf("% 2d", arr1[i]);
    }
}

```

```

    }
    printf("\ncopied into the second array :\n");
    for(i=0; i<num; i++)
    {
        printf("% 2d", arr2[i]);
    }

    printf("\n");
}

```

Output:-

```

enter the number of elements: 5
1 2 3 4 5

elements in the first array :
1 2 3 4 5
copied into the second array :
1 2 3 4 5

```

4. Count a total number of duplicate elements in an array.

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

```

void main()
{
    int arr1[30],arr2[30],arr3[30],num,a=1,count=0,i,j;

    printf("enter the number of elements :");
    scanf("%d",&num);

    printf("enter the elements in the array :\n");
    for(i=0;i<num;i++)
    {

        scanf("%d",&arr1[i]);
    }

    for(i=0;i<num; i++) //copy in other array
    {
        arr2[i]=arr1[i];
        arr3[i]=0;
    }

    for(i=0;i<num; i++) //duplicate array
    {
        for(j=0;j<num;j++)
        {

```

```

        if(arr1[i]==arr2[j])
        {
            arr3[j]=a;
            a++;
        }
        a=1;
    }

for(i=0; i<num; i++) //printing the array
{
    if(arr3[i]==2)
    {
        count++;
    }
}
printf("duplicate elements in the array are: %d\n", count);

printf("\n");
}

```

Output:-

```

enter the number of elements :6
enter the elements in the array :
1 2 2 4 5 4
duplicate elements in the array are: 2

```

5. Find the maximum and minimum element in an array.

```

#include <stdio.h>
int main()
{
    int a[30],max,min,num,i;

    printf("enter the number of elements :");
    scanf("%d",&num);

    printf("enter elements in the array :\n");
    for(i=0;i<num;i++)
    {
        scanf("%d",&a[i]);
    }
    max = a[0];

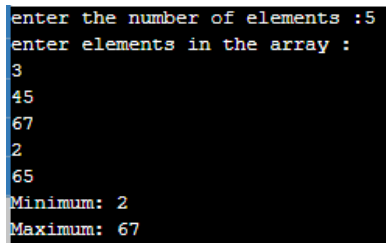
```

```

min = a[0];
for(i=1; i<num; i++)
{
    if(a[i]>max)
    {
        max = a[i];
    }
    if(a[i]<min)
    {
        min = a[i];
    }
}
printf("Minimum: %d\n", min);
printf("Maximum: %d\n", max);
return 0;
}

```

Output:-



```

enter the number of elements :5
enter elements in the array :
3
45
67
2
65
Minimum: 2
Maximum: 67

```

6. Separate odd and even integers in separate arrays.

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

```

int main()
{
    int arr1[20], arr2[20], arr3[20], r, c=0, m=0, num;
    printf("enter the number of elements :");
    scanf("%d", &num);

    printf("enter elements in the array :\n");
    for(r=0; r<num; r++)
    {
        scanf("%d", &arr1[r]);
    }

    for(r=0; r<num; r++)
    {

```

```

        if (arr1[r]%2 == 0)
        {
            arr2[c] = arr1[r];
            c++;
        }
        else
        {
            arr3[m] = arr1[r];
            m++;
        }
    }

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

    printf("\nOdd elements are :\n");
    for(r=0;r<m;r++)
    {
        printf("%d ", arr3[r]);
    }
    printf("\n");
}

```

Output:-

```

enter the number of elements :5
enter elements in the array :
1
2
4
5
7
Even elements are :
2 4
Odd elements are :
1 5 7

```

7. Insert new value in the array.

```

#include <stdio.h>
int main()
{
    int arr[30],i,num,p,start;

    printf("enter the size of array : ");
    scanf("%d", &num);
    printf("enter the elements in ascending order:\n");

```

```

        for(i=0;i<num;i++)
        {
            scanf("%d",&arr[i]);
        }
printf("enter the value to insert : ");
scanf("%d",&start);
printf("The exist array list is :\n ");
for(i=0;i<num;i++)
    printf("% 2d",arr[i]);
for(i=0;i<num;i++)
    if(start<arr[i])
    {
        p = i;
        break;
    }
for(i=num;i>=p;i--)
    arr[i]= arr[i-1];
    arr[p]=start;

    printf("\nAfter Inserting:\n ");
for(i=0;i<=num;i++)
    printf("% 2d",arr[i]);
    printf("\n");
    return 0;
}

```

Output:-

```

enter the size of array : 5
enter the elements in ascending order:
1
3
5
8
9
enter the value to insert : 4
The exist array list is :
 1 3 5 8 9
After Inserting:
 1 3 4 5 8 9

```

8. Delete an element at desired position from an array.

```

#include <stdio.h>
int main()
{
    int arr[50],i,position,num;

    printf("enter the size of array : ");
    scanf("%d", &num);
    for(i=0;i<num;i++)

```

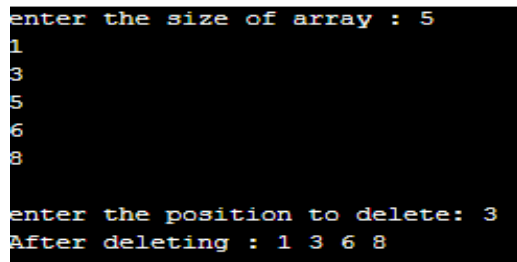
```

        {
            scanf("%d",&arr[i]);
        }

printf("\nenter the position to delete: ");
scanf("%d",&position);
i=0;
while(i!=position-1)
    i++;
while(i<num)
{
    arr[i]=arr[i+1];
    i++;
}
num--;
printf("After deleting :");
for(i=0;i<num;i++)
{
    printf(" %d",arr[i]);
}
printf("\n");
}

```

Output:-



```

enter the size of array : 5
1
3
5
6
8
enter the position to delete: 3
After deleting : 1 3 6 8

```

9. Find the second largest element in an array.

```

#include <stdio.h>
int main(){
int arr[30],num,i,j=0,large,sec_large;
    printf("enter the size of array: ");
    scanf("%d",&num);
    printf("enter elements in the array :\n");
    for(i=0;i<num;i++)
    {
        scanf("%d",&arr[i]);
    }
}

```



```

    large=0;
    for(i=0;i<num;i++)
    {
        if(large<arr[i])
        {
            large=arr[i];
            j = i;
        }
    }
    sec_large=0;
    for(i=0;i<num;i++)
    {
        if(i==j)
        {
            i++;
            i--;
        }
        else
        {
            if(sec_large<arr[i])
            {
                sec_large=arr[i];
            }
        }
    }

    printf("Second largest element in the array is : %d\n", sec_large);
    return 0;
}

```

Output:-

```

enter the size of array: 5
enter elements in the array :
2
34
45
12
23
Second largest element in the array is : 34

```

10. Find the median of two sorted arrays of same size.

```

#include <stdio.h>
int max(int a, int b)
{
    return ((a > b) ? a : b);
}

```

```

int min(int a, int b)
{
    return ((a < b) ? a : b);
}

int median(int arr[], int size)
{
    if (size % 2 == 0)
        return (arr[size/2] + arr[size/2-1])/2;
    else
        return arr[size/2];
}

int median2SortedArrays(int arr1[], int arr2[], int size)
{
    int med1;
    int med2;
    if(size <= 0) return -1;
    if(size == 1) return (arr1[0] + arr2[0])/2;
    if (size == 2) return (max(arr1[0], arr2[0]) + min(arr1[1], arr2[1])) / 2;

    med1 = median(arr1, size);
    med2 = median(arr2, size);

    if(med1 == med2) return med1;

    if (med1 < med2)
    {
        return median2SortedArrays(arr1 + size/2, arr2, size - size/2);
    }
    else
    {
        return median2SortedArrays(arr2 + size/2, arr1, size - size/2);
    }
}

int main()
{
    int i,m,n;
    int arr1[] = {1, 5, 13, 24, 35};
    int arr2[] = {3, 8, 15, 17, 32};
    m = sizeof(arr1) / sizeof(arr1[0]);
    n = sizeof(arr2) / sizeof(arr2[0]);

```

```

        printf("The given array - 1 is : ");
        for(i = 0; i < m; i++)
        {
            printf("%d ", arr1[i]);
        }

        printf("\n");

        printf("The given array - 2 is : ");
        for(i = 0; i < n; i++)
        {
            printf("%d ", arr2[i]);
        }

        printf("\n");

        printf("\nThe Median of the 2 sorted arrays is: %d",median2SortedArrays(arr1, arr2, n));
        printf("\n");
        return 0;
    }

```

Output:-

```

The given array - 1 is : 1 5 13 24 35
The given array - 2 is : 3 8 15 17 32

The Median of the 2 sorted arrays is: 14

```

11. Multiplication of two square Matrices

```

#include<stdio.h>
int main()
{
    int a[20][20],b[20][20],multi[20][20],r,c,i,j,k;
    //system("cls");
    printf("enter the number of row: ");
    scanf("%d",&r);
    printf("enter the number of column: ");
    scanf("%d",&c);
    printf("enter the first matrix element: \n");
    for(i=0;i<r;i++)
    {
        for(j=0;j<c;j++)
        {
            scanf("%d",&a[i][j]);
        }
    }
}

```

```

}
printf("enter the element of second matrix: \n");
for(i=0;i<r;i++)
{
for(j=0;j<c;j++)
{
scanf("%d",&b[i][j]);
}
}
}

```

```

printf("multiply of the matrix: \n");
for(i=0;i<r;i++)
{
for(j=0;j<c;j++)
{
multi[i][j]=0;
for(k=0;k<c;k++)
{
multi[i][j]+=a[i][k]*b[k][j];
}
}
}
}

```

```

for(i=0;i<r;i++)
{
for(j=0;j<c;j++)
{
printf("%d\t",multi[i][j]);
}
printf("\n");
}
return 0;
}

```

Output:-

```

enter the number of row: 2
enter the number of column: 2
enter the first matrix element:
2
2
3
2
enter the element of second matrix:
2
4
3
2
multiply of the matrix:
10    12
12    16

```

12. Find transpose of a given matrix.

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

{
    int a[30][30],b[30][30],i,j,r,c;

    printf("\nEnter the rows and column : ");
    scanf("%d %d",&r,&c);

    printf("Enter the element of first Matrix :\n");
    for(i=0;i<r;i++)
    {
        for(j=0;j<c;j++)
        {
            scanf("%d",&a[i][j]);
        }
    }

    printf("\nGiven matrix is:\n");
    for(i=0;i<r;i++)
    {
        printf("\n");
        for(j=0;j<c;j++)
            printf("%d\t",a[i][j]);
    }

    for(i=0;i<r;i++)
    {
        for(j=0;j<c;j++)
        {
            b[j][i]=a[i][j];
        }
    }

    printf("\nThe transpose of a matrix is : ");
    for(i=0;i<c;i++)
    {
        printf("\n");
        for(j=0;j<r;j++)
        {
            printf("%d\t",b[i][j]);
        }
    }
}
```

```

    }
}
printf("\n");
return 0;
}

```

Output:-

```

enter the rows and column : 2
2
enter the element of first Matrix :
2
3
2
4
given matrix is:
2      3
2      4
The transpose of a matrix is :
2      2
3      4

```

13. Find the sum of left diagonals of a matrix.

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

```
int main()
```

```

{
    int i,j,a[20][20],sum=0,size,m=0;

    printf("enter the size of the square matrix : ");
    scanf("%d", &size);
    m=size;
    printf("enter elements of first matrix :\n");
    for(i=0;i<size;i++)
    {
        for(j=0;j<size;j++)
        {
            scanf("%d",&a[i][j]);
        }
    }

    printf("The matrix is :\n");
    for(i=0;i<size;i++)
    {
        for(j=0;j<size ;j++)
            printf("% 2d",a[i][j]);
        printf("\n");
    }
    for(i=0;i<size;i++)
    {

```

```

    m=m-1;
    for(j=0;j<size ;j++)
    {
        if (j==m)
        {
            sum= sum+a[i][j];
        }

    }
}
printf("Addition of the left Diagonal elements is :%d\n",sum);
return 0;
}

```

Output:-

```

enter the size of the square matrix : 2
enter elements of first matrix :
2
2
3
4
The matrix is :
2 2
3 4
Addition of the left Diagonal elements is :5

```

14. Check whether a given matrix is an identity matrix.

```

#include <stdio.h>
int main()
{
    int a[10][10],r,c,i,j,k=1;
    printf("enter the number of rows :");
    scanf("%d", &r);
    printf("enter the column of matrix :");
    scanf("%d",&c);
    printf("enter the elements of matrix :\n");
    for(i=0;i<r;i++)
    {
        for(j=0;j<c;j++)
        {
            scanf("%d",&a[i][j]);
        }
    }
    printf("given matrix is :\n");
    for(i=0;i<r;i++)
    {

```

```

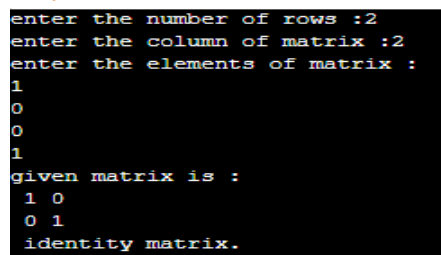
        for(j=0;j<c ;j++)
            printf("% 2d",a[i][j]);
        printf("\n");
    }

    for(i=0; i<r; i++)
    {
        for(j=0; j<c; j++)
        {
            if(a[i][j] != 1 && a[j][i] !=0)
            {
                k = 0;
                break;
            }
        }
    }

    if(k == 1 )
        printf(" identity matrix.\n");
    else
        printf(" not an identity matrix.\n");
    return 0;
}

```

Output:-



```

enter the number of rows :2
enter the column of matrix :2
enter the elements of matrix :
1
0
0
1
given matrix is :
1 0
0 1
identity matrix.

```

15. Search an element in a row wise and column wise sorted matrix.

```

#include <stdio.h>

int searchElement(int arr2D[4][4], int n, int x)
{
    int i = 0, j = n-1;
    while ( i < n && j >= 0 )
    {
        if ( arr2D[i][j] == x )
        {

```



```

        printf("\nThe element Found at the position in the matrix is: %d, %d", i, j);
        return 1;
    }
    if ( arr2D[i][j] < x )
        j--;
    else
        i++;
    }
    printf("\nThe given element not found in the 2D array.");
    return 0;
}

int main()
{
    int arr2D[4][4] = { {22, 20, 31, 39},
                        {18, 26, 36, 43},
                        {25, 18, 30, 32},
                        {30, 34, 29, 50},
                        };
    int i,j,v;
    v=20;

    printf("The given array in matrix form is : \n");
    for(i = 0; i < 4; i++)
    {
        for (j=0;j<4;j++)
        {
            printf("%d ", arr2D[i][j]);
        }
        printf("\n");
    }
    printf("The given value for searching is: %d",v);
    searchElement(arr2D, 4, v);
    return 0;
}

```

Output:-

```

The given array in matrix form is :
22 20 31 39
18 26 36 43
25 18 30 32
30 34 29 50
The given value for searching is: 20
The given element not found in the 2D array.

```

OPTIONAL

1. Print all unique elements in an array.

```
#include <stdio.h>
int main()
{
    int a[30], num, count=0, i, j, k;
    printf("enter the number of elements: ");
    scanf("%d", &num);
    printf("enter elements in the array :\n");
    for(i=0; i<num; i++)
    {
        scanf("%d", &a[i]);
    }
    printf("\nunique elements in the array are: \n");
    for(i=0; i<num; i++)
    {
        count=0;
        for(j=0, k=num; j<k+1; j++)
        {
            if (i!=j)
            {
                if(a[i]==a[j])
                {
                    count++;
                }
            }
        }
        if(count==0)
        {
            printf("%d ", a[i]);
            return 0;
        }
    }
    printf("\n");
}
```

Output:-

```

enter the number of elements: 3
enter elements in the array :
1
2
2
unique elements in the array are:
1

```

3. Sort elements of the array in descending order.

```

#include <stdio.h>
int main()
{
    int a[30],num,i,j,temp;
    printf("enter the size of array : ");
    scanf("%d",&num);

    printf("enter elements in the array :\n");
    for(i=0;i<num;i++)
    {
        scanf("%d",&a[i]);
    }
    for(i=0; i<num; i++)
    {
        for(j=i+1; j<num; j++)
        {
            if(a[i] < a[j])
            {
                temp = a[i];
                a[i] = a[j];
                a[j] = temp;
            }
        }
    }

    printf("\nIn descending order:\n");
    for(i=0; i<num; i++)
    {
        printf("%d ", a[i]);

    }

    printf("\n");
    return 0;
}

```

Output:-

```

enter the size of array : 5
enter elements in the array :
3
2
6
8
4

In descending order:
8 6 4 3 2

```

4. Find the second smallest element in an array.

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

```
int main()
```

```
{
```

```
    int a[30],num,i,j=0,small,sec_small;
```

```
        printf("enter the size of array : ");
```

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

```
        printf("Input elements in the array:\n");
```

```
        for(i=0;i<num;i++)
```

```
        {
```

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

```
        }
```

```
        small=a[0];
```

```
        for(i=0;i<num;i++)
```

```
        {
```

```
            if(small>a[i])
```

```
            {
```

```
                small=a[i];
```

```
                j = i;
```

```
            }
```

```
        }
```

```
        for(i=0;i<num;i++)
```

```
        {
```

```
            if(i==j)
```

```
            {
```

```
                i++;
```

```
                i--;
```

```
            }
```

```
        else
```

```
        {
```

```
            if(sec_small>a[i])
```

```
            {
```

```
                sec_small=a[i];
```

```
            }
```

```

    }
}

```

```

printf("The Second smallest element in the array is : %d \n\n", sec_small);
return 0;

```

```

}

```

Output:-

```

enter the size of array : 5
Input elements in the array:
2
8
5
6
4
The Second smallest element in the array is : 4

```

6. Find numbers that occur odd number of times in an array.

```

#include <stdio.h>

```

```

int findOdd(int *arr, int num )

```

```

{
    int i, ResultXor = 0;
    for(i = 0; i < num; i++)
    {
        ResultXor = ResultXor ^ arr[i];
    }
    return ResultXor;
}

```

```

int main()

```

```

{
    int i;
    int arr[] = {3, 1, 8, 4, 1, 3, 1, 7, 3};

```

```

    int ctr = sizeof(arr)/sizeof(arr[0]);
    printf("The given array is : ");

```

```

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

```

```

printf("odd number occur : %d times.\n", findOdd(arr, ctr));

```

```
return 0;
}
```

Output:-

```
The given array is : 3 1 8 4 4 3 4 7 3
odd number occur : 9 times.
```

8. Subtraction of two Matrices.

```
#include <stdio.h>
int main()
{
    int a[30][30],b[30][30],c[30][30],i,j,num;
    printf("enter the size of matrix ");
    scanf("%d", &num);
    printf("enter elements in the first matrix :\n");
    for(i=0;i<num;i++)
    {
        for(j=0;j<num;j++)
        {
            scanf("%d",&a[i][j]);
        }
    }

    printf("enter elements in the second matrix :\n");
    for(i=0;i<num;i++)
    {
        for(j=0;j<num;j++)
        {
            scanf("%d",&b[i][j]);
        }
    }
    printf("\nFirst matrix :\n");
    for(i=0;i<num;i++)
    {
        printf("\n");
        for(j=0;j<num;j++)
            printf("%d\t",a[i][j]);
    }

    printf("\nSecond matrix:\n");
    for(i=0;i<num;i++)
    {
```

```

        printf("\n");
        for(j=0;j<num;j++)
            printf("%d\t",b[i][j]);
    }

    for(i=0;i<num;i++)
        for(j=0;j<num;j++)
            c[i][j]=a[i][j]-b[i][j];

    printf("\nSubtraction of two matrix: \n");
    for(i=0;i<num;i++){
        printf("\n");
        for(j=0;j<num;j++)
            printf("%d\t",c[i][j]);
    }
    printf("\n");
}

```

Output:-

```

enter the size of matrix 2
enter elements in the first matrix :
2
3
4
5
enter elements in the second matrix :
1
2
3
4

First matrix :
2      3
4      5
Second matrix:
1      2
3      4
Subtraction of two matrix:
1      1
1      1

```

9. Find sum of right diagonals of a matrix.

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

```
int main()
```

```

{
    int i,j,a[30][30],sum=0,num;
    printf("enter the size of matrix: ");
    scanf("%d", &num);
    printf("enter the elements of matrix :\n");

```

```

for(i=0;i<num;i++)
{
    for(j=0;j<num;j++)
    {
        scanf("%d",&a[i][j]);
        if (i==j)
            sum= sum+a[i][j];
    }
}

printf("The matrix is :\n");
for(i=0;i<num;i++)
{
    for(j=0;j<num ;j++)
        printf("% 2d",a[i][j]);
    printf("\n");
}

printf("Addition of the right Diagonal elements :%d\n",sum);
return 0;
}

```

Output:-

```

enter the size of matrix: 2
enter the elements of matrix :
2
3
4
3
The matrix is :
 2 3
 4 3
Addition of the right Diagonal elements :5

```

10. Display the lower triangular of a given matrix.

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

```
int main()
```

```
{
```

```
int a[20][20],i,j,num;
```

```
printf("enter the size of the square matrix : ");
```

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

```
printf("enter elements in the matrix :\n");
```

```
for(i=0;i<num;i++)
```

```
{
```

```
for(j=0;j<num;j++)
```

```
{
```



```

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

printf("The matrix is :\n");
for(i=0;i<num;i++)
{
    for(j=0;j<num;j++)
        printf("% 2d",a[i][j]);
    printf("\n");
}

printf("\nset 0 in lower traingle matrix: \n");
for(i=0;i<num;i++){
    printf("\n");
    for(j=0;j<num;j++)
        if(i<=j)
            printf("% 2d",a[i][j]);
        else
            printf("% 2d",0);
}
printf("\n");
}

```

Output:-

```

enter the size of the square matrix : 3
enter elements in the matrix :
1
2
3
4
2
3
4
5
6
The matrix is :
 1 2 3
 4 2 3
 4 5 6

set 0 in lower traingle matrix:

 1 2 3
 0 2 3
 0 0 6

```

11. Calculate determinant of a 3 x 3 matrix.

```

#include <stdio.h>

int main()
{
    int a[10][10],i,j,n,det=0;
    printf("enter elements in the first matrix :\n");

```

```

for(i=0;i<3;i++)
{
    for(j=0;j<3;j++)
    {
        scanf("%d",&a[i][j]);
    }
}

printf("The matrix is :\n");
for(i=0;i<3;i++)
{
    for(j=0;j<3 ;j++)
        printf("% 2d",a[i][j]);
    printf("\n");
}

for(i=0;i<3;i++)
    det = det + (a[0][i]*(a[1][(i+1)%3]*a[2][(i+2)%3] - a[1][(i+2)%3]*a[2][(i+1)%3]));

printf("\nThe Determinant of the matrix is: %d\n\n",det);
return 0;
}

```

Output:-

```

enter elements in the first matrix :
3
4
2
4
3
6
3
4
3
The matrix is :
 3  4  2
 4  3  6
 3  4  3
The Determinant of the matrix is: -7

```