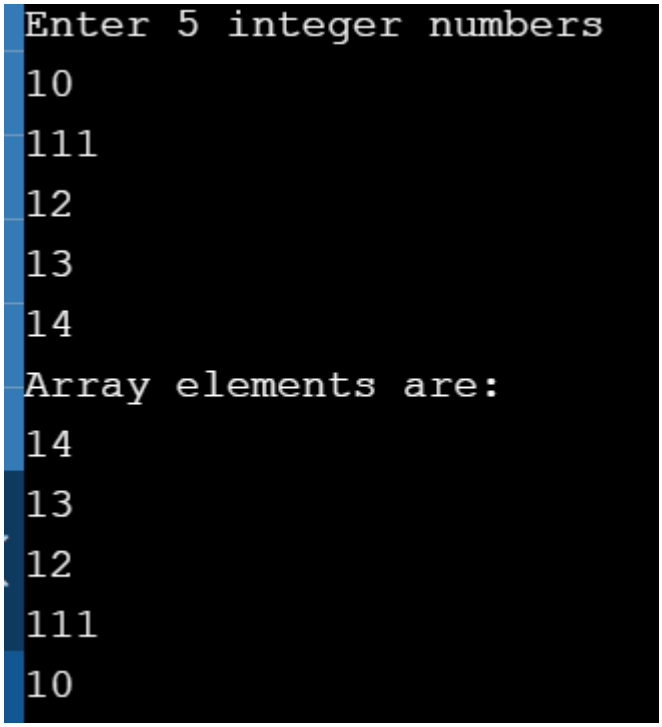
1: read n number of values in an array and display it in reverse order.

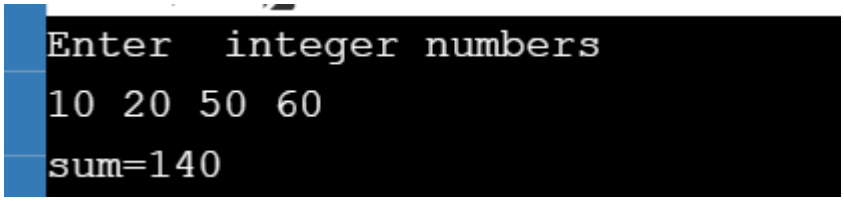
```
#include<stdio.h>
int main()
{
   int a[5], i;
   printf("Enter 5 integer numbers\n");
   for(i = 0; i < 5; i++)
      scanf("%d", &a[i]);
   printf("Array elements are:\n");
   for(i = 4; i >= 0; i--)
      printf("%d\n", a[i]);
   return 0;
}
```



2: find the sum of all elements of the array.

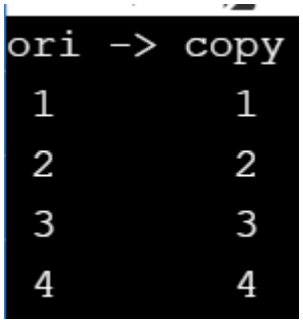
```
#include<stdio.h>
int main()
{
   int a[4], i,sum;
   printf("Enter  integer numbers\n");
```

```
    for(i=1;i<=4;i++)
    scanf("%d",&a[i]);
    for(i=1;i<=4;i++)
    sum+=a[i];
    printf("sum=%d\n",sum);
    return 0;
}
```



3: copy the elements of one array into another array.

```
#include<stdio.h>
int main()
{
    int ori[4]={1,2,3,4};
    int copy[4];
    int i;
    for(i=0;i<4;i++){
    copy[i]=ori[i];
    }

    printf("ori -> copy\n");
    for(i=0;i<4;i++){
    printf(" %d      %d\n",ori[i],copy[i]);
    }
    return 0;
}
```



4: count a total number of duplicate elements in an array.

```
#include<stdio.h>
int main()
{
   int a[5],i,j,size,count=0;
   printf("enter the size of array:");
   scanf("%d\n",&size);
   printf("enter the elements of array");
   for(i=0;i<5;i++)
   scanf("%d",&a[i]);
   {
   for(j=i+1;j<5;j++)
     {
    if (a[i]==a[j]);
       {
         count++;
          break;
        }
      }
    }

   printf("\ntotal no of duplicate elements in array= %d",count);
  return 0;
}
```

```
Please Enter Number of elements in an array  :   7

Please Enter 7 elements of an Array  :  66 8 4 90 66 6 89

Total Number of Duplicate Elements in this Array  =  1
```

5: find the maximum and minimum element in an array.

```
#include<stdio.h>
int main()
{
  int arr[10],i,n,largest,smallest;
  printf("enter the size of array:");
  scanf("%d",&n);
  printf("enter the elements of the array:");
  for(i=0;i<n;i++)
  {
     scanf("%d",&arr[i]);
  }
  largest=arr[0];
  smallest=arr[0];
  for(i=0;i<n;i++)
  {
     if(arr[i]>largest)
     {

        largest=arr[i];
     }
     else if (arr[i]<smallest)
     {
        smallest=arr[i];
     }
  }
```

```
  printf("largest=%d,smallest=%d",largest,smallest);
 }
```

```
enter the size of array:6
enter the elements of the array:1 6 8 9 3 5
largest=9,smallest=1
```

6: separate odd and even integers in separate arrays.

```
#include<stdio.h>
   int main()
{
   int arr[10],i,num;
   printf("Enter size of the array\n");
   scanf("%d",&num);
   printf("Enter the elements of the array\n");
   for(i=0; i<num; i++){
      scanf("%d",&arr[i]);
   }
   printf("\nEven numbers of the array are \n");
    for(i=0; i<num; i++){
      if(arr[i]%2==0){
         printf("%d \t",arr[i]);
      }
   }
   printf("\nOdd numbers of the array are \n");
    for(i=0; i<=num; i++){
      if (arr[i]%2==1){
         printf("%d \t",arr[i]);
      }
   }
  return 0;
}
```

```
Enter size of the array
5
Enter the elements of the array
1 2 3 4 5
Even numbers of the array are
2   4
Odd numbers of the array are
1   3   5
```

7. insert New value in the array.

```
#include<stdio.h>
void main()
{
   int a[100],i,n,number,pos;

   printf("\nEnter no of elements\n");
   scanf("%d",&n);

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

   printf("Elements of array are\n");
   for(i=0;i<n;i++)
   {
      printf("a[%d] = %d\n",i,a[i]);
   }

   printf("Enter the number which you want to insert\n");
```

```
    scanf("%d",&number);
    printf("Enter the position where you want to insert the number\n");
    scanf("%d",&pos);
    for(i=n-1;i>=pos;i--)
    {
       a[i+1]=a[i];
    }
    n=n+1;
    a[pos]=number;
    printf("\nOn inserting new array we get is\n");
    for(i=0;i<n;i++)
    {
       printf("a[%d] = %d\n",i,a[i]);
    }
}
```

```
Enter no of elements
3
Enter the elements
1 2 3
Elements of array are
a[0] = 1
a[1] = 2
a[2] = 3
Enter the number which you want to insert
78
Enter the position where you want to insert the number
3
On inserting new array we get is
a[0] = 1
a[1] = 2
a[2] = 3
a[3] = 78
```

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

```
#include <stdio.h>
int main()
{
  int array[10], pos, a, n;

  printf("Enter number of elements in array:");
  scanf("%d", &n);

  printf("Enter %d elements:\n", n);

  for (a = 0; a < n; a++)
    scanf("%d", &array[a]);
```

```
  printf("Enter the location where you wish to delete element\n");
  scanf("%d", &pos);

  if (pos >= n+1)
    printf("Deletion not possible.\n");
  else
  {
    for (a = pos - 1; a < n - 1; a++)
      array[a] = array[a+1];

    printf("Resultant array:\n");

    for (a = 0; a < n - 1; a++)
      printf("%d\n", array[a]);
  }

  return 0;
}
```

```
Enter number of elements in array:3
Enter 3 elements:
5 6 7
Enter the location where you wish to delete element
2
Resultant array:
5
7
```

```
Enter number of elements in array:6
Enter 6 elements:
5 6 7 8 9 0
Enter the location where you wish to delete element
7
Deletion not possible.
```

9. find the second largest element in an array.

```
#include<stdio.h>
int main()
{
  int arr[10],i,n,largest,sec_largest;
  printf("enter the size of array:");
  scanf("%d",&n);
  printf("enter the elements of the array:");
  for(i=0;i<n;i++)
  {
     scanf("%d",&arr[i]);
  }
  largest=arr[0];
  sec_largest=arr[1];
  for(i=0;i<n;i++)
  {
     if(arr[i]>largest)
     {
        sec_largest=largest;
        largest=arr[i];
     }
     else if (arr[i]>sec_largest && arr[i]!=largest)
     {
        sec_largest=arr[i];
     }
  }
```

```
printf("largest=%d,sec_largest=%d",largest,sec_largest);
 }
```

```
enter the size of array:7
enter the elements of the array:6 7 8 9 0 3 5
largest=9,sec_largest=8
```

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");
```

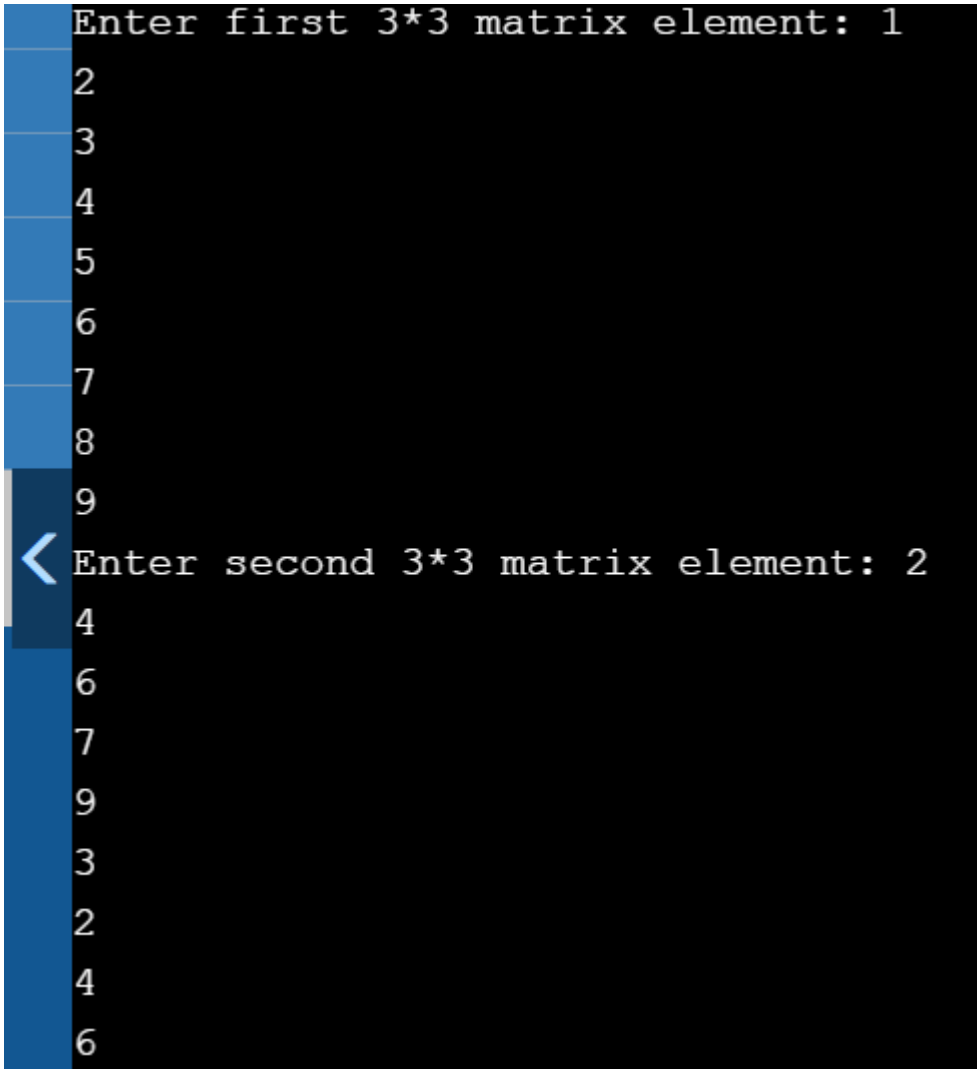
```
  printf("\nThe Median of the 2 sorted arrays is:
%d",median2SortedArrays(arr1, arr2, n));
  printf("\n");
  return 0;
}
```

```
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 mat1[3][3], mat2[3][3], mat3[3][3], sum=0, i, j, k;
   printf("Enter first 3*3 matrix element: ");
   for(i=0; i<3; i++)
   {
      for(j=0; j<3; j++)
         scanf("%d", &mat1[i][j]);
   }
   printf("Enter second 3*3 matrix element: ");
   for(i=0; i<3; i++)
   {
      for(j=0; j<3; j++)
         scanf("%d", &mat2[i][j]);
   }
   printf("\nMultiplying two matrices...");
```

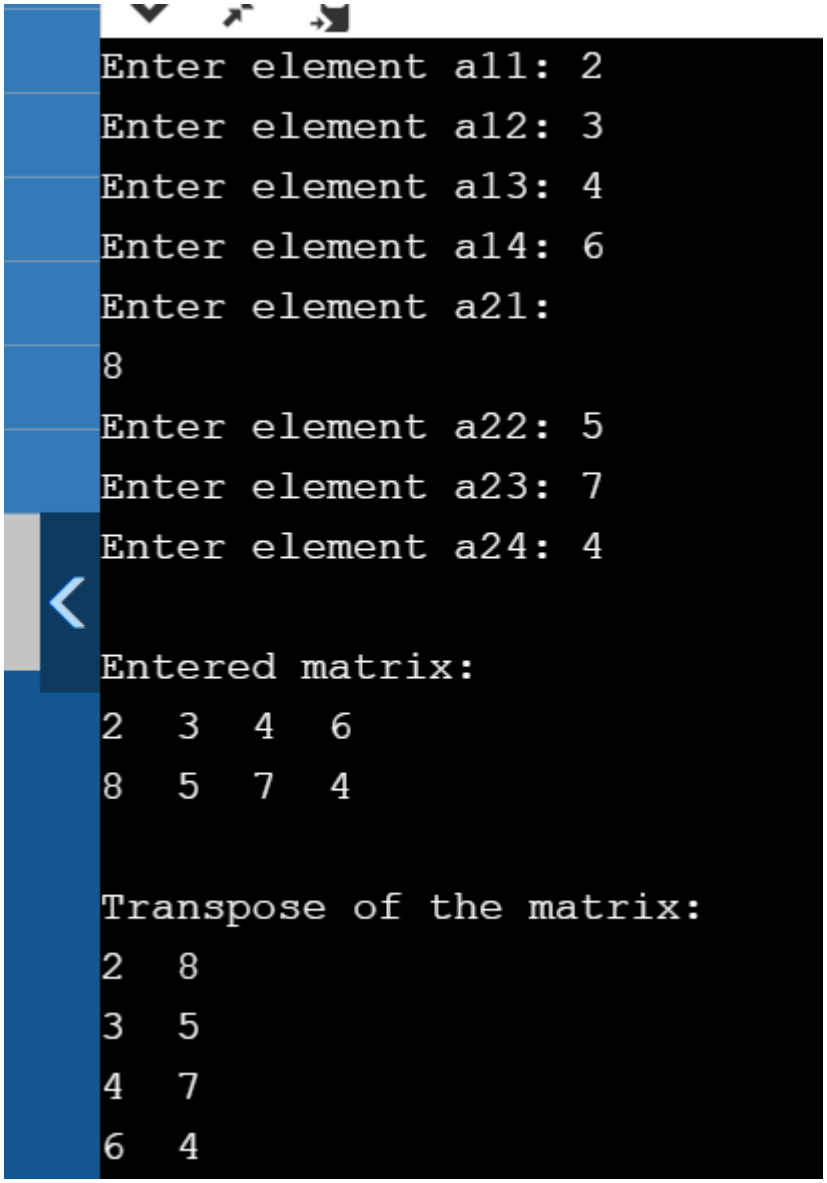
```
    for(i=0; i<3; i++)
    {
        for(j=0; j<3; j++)
        {
            sum=0;
            for(k=0; k<3; k++)
                sum = sum + mat1[i][k] * mat2[k][j];
            mat3[i][j] = sum;
        }
    }
    printf("\nMultiplication result of the two given Matrix is: \n");
    for(i=0; i<3; i++)
    {
        for(j=0; j<3; j++)
            printf("%d\t", mat3[i][j]);
        printf("\n");
    }

    return 0;
}
```



12. find transpose of a given matrix.

```
#include <stdio.h>
int main() {
   int a[10][10], transpose[10][10], r, c, i, j;
   printf("Enter rows and columns: ");
   scanf("%d %d", &r, &c);
   printf("\nEnter matrix elements:\n");
   for (i = 0; i < r; ++i)
      for (j = 0; j < c; ++j) {
         printf("Enter element a%d%d: ", i + 1, j + 1);
         scanf("%d", &a[i][j]);
      }
```

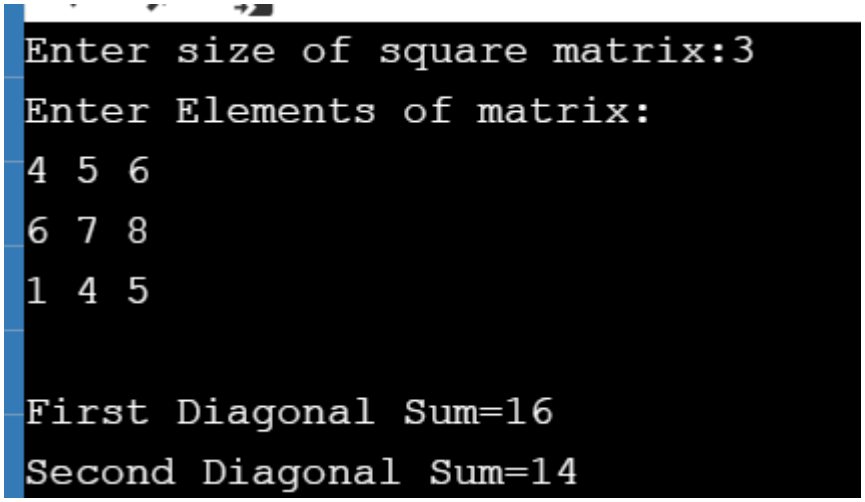
```
    printf("\nEntered matrix: \n");
    for (i = 0; i < r; ++i)
        for (j = 0; j < c; ++j) {
            printf("%d  ", a[i][j]);
            if (j == c - 1)
                printf("\n");
        }

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

    printf("\nTranspose of the matrix:\n");
    for (i = 0; i < c; ++i)
        for (j = 0; j < r; ++j) {
            printf("%d  ", transpose[i][j]);
            if (j == r - 1)
                printf("\n");
        }
    return 0;
}
```



13. find the sum of left diagonals of a matrix

```
#include<stdio.h>

int main()
{
        int i,j,n,d1=0,d2=0,a[5][5];
        printf("Enter size of square matrix:");
```

```
    scanf("%d",&n);
    printf("Enter Elements of matrix:\n");

    for(i=0;i<n;++i)
        for(j=0;j<n;++j)
        {
            scanf("%d",&a[i][j]);
            if(i==j)
                d1+=a[i][j];
            if((i+j)==(n-1))
                d2+=a[i][j];
        }

    printf("\nFirst Diagonal Sum=%d",d1);
    printf("\nSecond Diagonal Sum=%d",d2);

    return 0;
}
```



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

```
#include <stdio.h>
void main()
{
   int a[10][10];
   int i, j, row, column, flag = 1;
   printf("Enter the order of the matrix A \n");
   scanf("%d %d", &row, &column);
   printf("Enter the elements of matrix A \n");
   for (i = 0; i < row; i++)
   {
      for (j = 0; j < column; j++)
      {
         scanf("%d", &a[i][j]);
      }
   }
   printf("MATRIX A is \n");
   for (i = 0; i < row; i++)
   {
      for (j = 0; j < column; j++)
      {
         printf("%3d", a[i][j]);
      }
      printf("\n");
   }
```

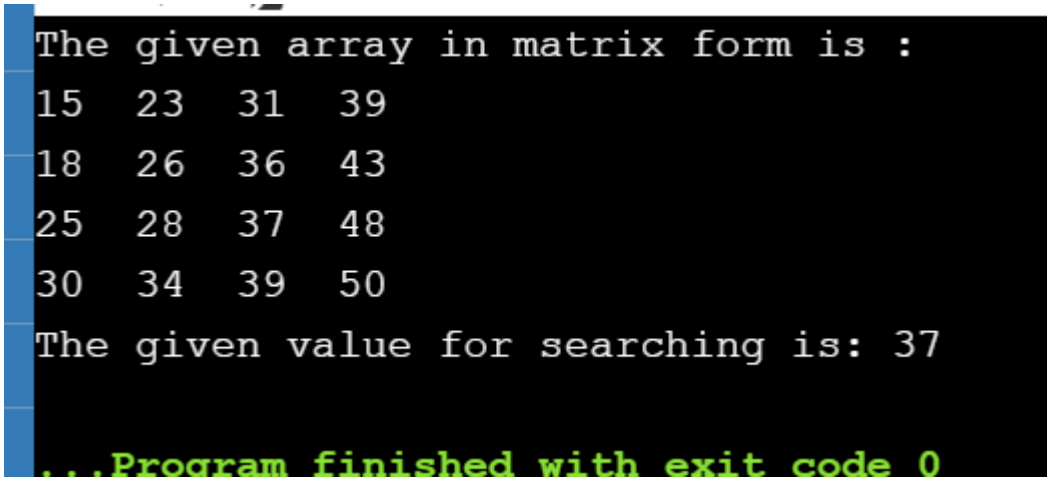
```
    for (i = 0; i < row; i++)
    {
        for (j = 0; j < column; j++)
        {
            if (a[i][j] != 1 && a[j][i] != 0)
            {
                flag = 0;
                break;
            }
        }
    }
    if (flag == 1 )
        printf("It is identity matrix \n");
    else
        printf("It is not a identity matrix \n");
}
```

```
Enter the order of the matrix A
3
3
Enter the elements of matrix A
1 0 0
0 1 0
0 0 1
MATRIX A is
  1   0   0
  0   1   0
  0   0   1
It is 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++;
  }
  return 0;
}

int main(){
 int arr2D[4][4] = { {15, 23, 31, 39},
                {18, 26, 36, 43},
                {25, 28, 37, 48},
                {30, 34, 39, 50},
               };
int i,j,v;
v=37;
            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;}
```



Practice Questions [Optional]:

1. **print all unique elements in an array**.

```
#include<stdio.h>

int main() {
 int arr[10], n, i, j;

 printf("Enter number of elements in array\n");
 scanf("%d", &n);
 printf("Enter %d numbers\n", n);

     for(i = 0; i < n; i++){
      scanf("%d", &arr[i]);
     }

     printf("Unique Elements\n");
     for(i = 0; i < n; i++) {
      for (j=0; j<i; j++){
          if (arr[i] == arr[j])
           break;
           }

         if (i == j){

```

```
24.            printf("%d ", arr[i]);
25.         }
26.       }
27.
28.     return 0;
29.     }
```

```
Enter number of elements in array
5
Enter 5 numbers
3 4 5 3 5
Unique Elements
3 4 5
```

2. count the frequency of each element of an array.

```
#include <stdio.h>

int main()
{
   int arr[100], freq[100];
   int size, i, j, count;

   printf("Enter size of array: ");
   scanf("%d", &size);
   printf("Enter elements in array: ");
   for(i=0; i<size; i++)
   {
      scanf("%d", &arr[i]);
      freq[i] = -1;
   }


   for(i=0; i<size; i++)
   {
      count = 1;
      for(j=i+1; j<size; j++)
      {

         if(arr[i]==arr[j])
         {
            count++;

            freq[j] = 0;
```

```
        }
      }

      if(freq[i] != 0)
      {
        freq[i] = count;
      }
    }


    printf("\nFrequency of all elements of array : \n");
    for(i=0; i<size; i++)
    {
      if(freq[i] != 0)
      {
        printf("%d occurs %d times\n", arr[i], freq[i]);
      }
    }

    return 0;
}
```

```
Enter the value of N
6
Enter the numbers
2 5 3 7 6 4
The numbers arranged in descending order are given below
7
6
5
4
3
2
```

3: . sort elements of the array in descending order

```
#include <stdio.h>
    void main ()
    {
      int number[30];
      int i, j, a, n;
      printf("Enter the value of N\n");
      scanf("%d", &n);
      printf("Enter the numbers \n");
```

```
    for (i = 0; i < n; ++i)
            scanf("%d", &number[i]);
    for (i = 0; i < n; ++i)
    {
       for (j = i + 1; j < n; ++j)
       {
          if (number[i] < number[j])
          {
             a = number[i];
             number[i] = number[j];
             number[j] = a;
          }
       }
    }

    printf("The numbers arranged in descending order are given below\n");

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

  }
```

```
Enter the value of N
6
Enter the numbers
2 5 3 7 6 4
The numbers arranged in descending order are given below
7
6
5
4
3
2
```

4. find the second smallest element in an array

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

```
   int a[50];
   int n,i,small,sec_small;
   printf("\n Enter number of elements: ");
   scanf("%d",&n);

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

   small=sec_small=a[0];

   for(i=1;i<n;i++)
   {
      if(small>a[i])
      {
         sec_small=small;
         small=a[i];
      }
      else if(sec_small>a[i] && a[i]!=small)
      {
         sec_small=a[i];
      }
   }

   printf("\n The Second Smallest Element in the given Array: %d", sec_small);

}
```

```
Enter number of elements: 7
Enter 7 elements: 4 5 6 3 2 8 0

 The Second Smallest Element in the given Array: 2
```

8. subtraction of two Matrices

```
#include<stdio.h>


int main()
{
```

```
int i, j, rows, columns, a[10][10], b[10][10];
int Subtraction[10][10];

printf("\n Please Enter Number of rows and columns  :  ");
scanf("%d %d", &i, &j);

printf("\n Please Enter the First Matrix Elements\n");
for(rows = 0; rows < i; rows++)
{
        for(columns = 0;columns < j;columns++)
{
        scanf("%d", &a[rows][columns]);
}
}

printf("\n Please Enter the Second Matrix Elements\n");
for(rows = 0; rows < i; rows++)
{
        for(columns = 0;columns < j;columns++)
{
        scanf("%d", &b[rows][columns]);
}
}
```

```
        for(rows = 0; rows < i; rows++)
        {
                for(columns = 0;columns < j;columns++)
        {
                Subtraction[rows][columns] = a[rows][columns] -
b[rows][columns];
                }
        }

        printf("\n After Subtracting Matrix a from Matrix b = a - b \n");
        for(rows = 0; rows < i; rows++)
        {
                for(columns = 0; columns < j; columns++)
        {
                printf("%d \t ", Subtraction[rows][columns]);
        }
        printf("\n");
        }
        return 0;
}
```

```
 Please Enter Number of rows and columns  :  2 2

 Please Enter the First Matrix Elements
23 45
56 78

 Please Enter the Second Matrix Elements
6 7
7 9

 After Subtracting Matrix a from Matrix b = a - b
17        38
49        69
```

9. find sum of right diagonals of a matrix.

```
#include<stdio.h>

int main()
{
        int i,j,n,d1=0,d2=0,a[5][5];
        printf("Enter size of square matrix:");
        scanf("%d",&n);
        printf("Enter Elements of matrix:\n");

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

```
            for(j=0;j<n;++j)
            {
                scanf("%d",&a[i][j]);
                if(i==j)
                    d1+=a[i][j];
                if((i+j)==(n-1))
                    d2+=a[i][j];
            }

        printf("\nFirst Diagonal Sum=%d",d1);
        printf("\nSecond Diagonal Sum=%d",d2);

        return 0;
}
```

```
Enter size of square matrix:2 2
Enter Elements of matrix:
56 78
78 90

First Diagonal Sum=80
Second Diagonal Sum=134
```

10. display the lower triangular of a given matrix.

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

```
int main()
{
	int i, j, rows, columns, a[10][10];

	printf("\n Please Enter Number of rows and columns  :  ");
	scanf("%d %d", &i, &j);

	printf("\n Please Enter the Matrix Elements \n");
	for(rows = 0; rows < i; rows++)
	{
		for(columns = 0;columns < j;columns++)
	{
		scanf("%d", &a[rows][columns]);
	}
	}

	for(rows = 0; rows < i; rows++)
	{
		printf("\n");
		for(columns = 0; columns < j; columns++)
	{
		if(rows >= columns)
		{
```

```
                printf("%d ", a[rows][columns]);
                }
                else
                {
                        printf("0  ");
                }
            }
        }

        return 0;
}
```

```
 Please Enter Number of rows and columns  :  3 3

 Please Enter the Matrix Elements
3 4 5
6 7 8
9 0 5

3 0  0
6 7 0
9 0 5
```

11. calculate determinant of a 3 x 3 matrix.

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

```
int main(){

  int a[3][3], i, j;

  long determinant;
  printf("Enter the 9 elements of matrix: ");
  for(i = 0 ;i < 3;i++)
     for(j = 0;j < 3;j++)
          scanf("%d", &a[i][j]);

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

  determinant = a[0][0] * ((a[1][1]*a[2][2]) - (a[2][1]*a[1][2])) -a[0][1]
* (a[1][0]
   * a[2][2] - a[2][0] * a[1][2]) + a[0][2] * (a[1][0] * a[2][1] - a[2][0] *
a[1][1]);

  printf("\nDeterminant of 3X3 matrix: %ld", determinant);
```

```
  return 0;
}
```

```
Enter the 9 elements of matrix: 8 9 5 6 7 4 2 6 0

The matrix is

8       9       5
6       7       4
2       6       0
Determinant of 3X3 matrix: -10
```