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

```c
#include <stdio.h>
int main(){
  int i,n,a[100];
    printf("Read n number of values in an array and display it in reverse order:\n");
  printf("Input the number of elements to store in the array :");
  scanf("%d",&n);
  printf("Input %d number of elements in the array :\n",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");
  return 0;
}
```

```
Read n number of values in an array and display it in reverse order:

Input the number of elements to store in the array :4

Input 4 number of elements in the array :

element - 0 : 1

element - 1 : 2

element - 2 : 3

element - 3 : 4


The values store into the array are :

    1    2    3    4


The values store into the array in reverse are :

    4    3    2    1
```

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

```c
#include <stdio.h>

int main(){

    int a[100];

    int i, n, sum=0;

        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",&a[i]);

            }

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

            sum += a[i];

        }
```

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

    return 0;

}
```

```
Input the number of elements to be stored in the array :4

Input 4 elements in the array :

element - 0 : 1

element - 1 : 2

element - 2 : 3

element - 3 : 4

Sum of all elements stored in the array is : 10
```

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

```c
#include <stdio.h>

int main(){

    int arr1[100], arr2[100];

    int i, 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");

        return 0;

}
```

```
Input the number of elements to be stored in the array :4

Input 4 elements in the array :

element - 0 : 1

element - 1 : 2

element - 2 : 3

element - 3 : 4


The elements stored in the first array are :

    1    2    3    4


The elements copied into the second array are :

    1    2    3    4
```

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

```
#include <stdio.h>

int main(){

    int arr1[100];

        int arr2[100];
```

```c
    int arr3[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",&arr1[i]);
    }
            for(i=0;i<n; i++){
            arr2[i]=arr1[i];
            arr3[i]=0;
    }
        for(i=0;i<n; i++) {
            for(j=0;j<n;j++){
                        if(arr1[i]==arr2[j]){
                        arr3[j]=mm;
                        mm++;
                        }
                }
                mm=1;
    }
for(i=0; i<n; i++) {
    if(arr3[i]==2){
ctr++;
}
```

```c
    }
      printf("The total number of duplicate elements found in the array is: %d \n", ctr);


          printf("\n\n");

          return 0;

}
```

```
Input the number of elements to be stored in the array :7

Input 7 elements in the array :

element - 0 : 1

element - 1 : 2

element - 2 : 3

element - 3 : 4

element - 4 : 5

element - 5 : 4

element - 6 : 2

The total number of duplicate elements found in the array is: 2
```

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

```c
#include <stdio.h>

int main(){

   int arr1[100];

   int i, mx, mn, n;

        printf("Input the number of elements to be stored in the array :");

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

```c
    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);

    return 0;

}
```

```
Input the number of elements to be stored in the array :4

Input 4 elements in the array :

element - 0 : 1

element - 1 : 2

element - 2 : 3

element - 3 : 4

Maximum element is : 4

Minimum element is : 1
```

Q.6 Separate odd and even integers in separate arrays.

```c
#include <stdio.h>

int main(){

  int arr1[10], arr2[10], arr3[10];

  int i,j=0,k=0,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]);}
```

```c
    printf("\nThe Odd elements are :\n");

    for(i=0;i<k;i++){

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

    printf("\n\n");

    return 0; }
```

```
Input the number of elements to be stored in the array :4

Input 4 elements in the array :

element - 0 : 1

element - 1 : 2

element - 2 : 3

element - 3 : 4


The Even elements are :

2 4

The Odd elements are :

1 3
```

Q.7 Insert new values in the array.

```c
#include <stdio.h>

int main()

{

  int arr1[100],i,n,p,x;

    printf("Input the size of array : ");

    scanf("%d", &n);

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

            printf("element - %d : ",i);

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

        }
```

```c
    printf("Input the value to be inserted : ");

    scanf("%d",&x);

    printf("Input the Position, where the value to be inserted :");

    scanf("%d",&p);

    printf("The current list of the array :\n");

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

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

    for(i=n;i>=p;i--)

      arr1[i]= arr1[i-1];

      arr1[p-1]=x;

    printf("\n\nAfter Insert the element the new list is :\n");

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

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

          printf("\n\n");

          return 0;

}
```

```
Input the size of array : 4

element - 0 : 1

element - 1 : 2

element - 2 : 3

element - 3 : 4

Input the value to be inserted : 7

Input the Position, where the value to be inserted :2

The current list of the array :

    1    2    3    4


After Insert the element the new list is :

    1    7    2    3    4
```

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

```c
#include <stdio.h>
int main(){
 int arr1[50],i,pos,n;
    printf("Input the size of array : ");
    scanf("%d", &n);
    printf("Input %d elements in the array in ascending order:\n",n);
    for(i=0;i<n;i++){
            printf("element - %d : ",i);
            scanf("%d",&arr1[i]);
        }
 printf("\nInput the position where to delete: ");
 scanf("%d",&pos);
 i=0;
 while(i!=pos-1)
      i++;
 while(i<n){
      arr1[i]=arr1[i+1];
      i++;
 }
 n--;
 printf("\nThe new list is : ");
 for(i=0;i<n;i++){
                printf(" %d",arr1[i]);
            }
      printf("\n\n");
  return 0;
```

```
}
```

```
Input the size of array : 4

Input 4 elements in the array in ascending order:

element - 0 : 1

element - 1 : 2

element - 2 : 3

element - 3 : 4


Input the position where to delete: 2


The new list is :   1   3   4
```

Q.9 Find the second largest element in an array.

#include <stdio.h>

int main(){

  int arr1[50],n,i,j=0,lrg,lrg2nd;

    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]);

        }

  lrg=0;

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

   if(lrg<arr1[i]){

```c
        lrg=arr1[i];

        j = i;

     }

  }

  lrg2nd=0;

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

   if(i==j){

      i++;  /* ignoring the largest element */

               i--;

     }

    else{

      if(lrg2nd<arr1[i]){

         lrg2nd=arr1[i];

       }

     }

  }

  printf("The Second largest element in the array is :  %d \n\n", lrg2nd);

  return 0;

}
```

```
Input the size of array : 4

Input 4 elements in the array :

element - 0 : 4

element - 1 : 3

element - 2 : 2

element - 3 : 1

The Second largest element in the array is :  3
```

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

```c
#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);
```

```c
    }
}


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
```

Q.11 Multiplication of two square matrixes.

```c
#include <stdio.h>
int main(){
 int arr1[50][50],brr1[50][50],crr1[50][50],i,j,k,r1,c1,r2,c2,sum=0;
 printf("\nInput the rows and columns of first matrix : ");
 scanf("%d %d",&r1,&c1);
 printf("\nInput the rows and columns of second matrix : ");
 scanf("%d %d",&r2,&c2);
 if(c1!=r2){
    printf("Mutiplication of Matrix is not possible.");
    printf("\nColumn of first matrix and row of second matrix must be same.");
 }
 else{
    printf("Input elements in the first matrix :\n");
    for(i=0;i<r1;i++){
       for(j=0;j<c1;j++){
             printf("element - [%d],[%d] : ",i,j);
             scanf("%d",&arr1[i][j]);
       }
     }
    printf("\n Input elements in the second matrix :\n");
    for(i=0;i<r2;i++){
       for(j=0;j<c2;j++){
             printf("element - [%d],[%d] : ",i,j);
             scanf("%d",&brr1[i][j]);
       }
     }
```

```c
        printf("\nThe First matrix is :\n");

            for(i=0;i<r1;i++){

            printf("\n");

            for(j=0;j<c1;j++)

        printf("%d\t",arr1[i][j]);

            }
        printf("\nThe Second matrix is :\n");

            for(i=0;i<r2;i++){

            printf("\n");

            for(j=0;j<c2;j++)

            printf("%d\t",brr1[i][j]);

            }
    for(i=0;i<r1;i++)

        for(j=0;j<c2;j++)

        crr1[i][j]=0;

        for(i=0;i<r1;i++)  {

            for(j=0;j<c2;j++){

            sum=0;

                for(k=0;k<c1;k++)

                sum=sum+arr1[i][k]*brr1[k][j];

                crr1[i][j]=sum;

            }

        }
printf("\nThe multiplication of two matrices is : \n");

for(i=0;i<r1;i++){

    printf("\n");

    for(j=0;j<c2;j++){
```

```c
            printf("%d\t",crr1[i][j]);

        }

    }

 }
printf("\n\n");

return 0;

}
```

```
Input the rows and columns of first matrix : 2

2


Input the rows and columns of second matrix : 1

2

Mutiplication of Matrix is not possible.

Column of first matrix and row of second matrix must be same.
```

Q.12 Find transpose of a given matrix.

```c
#include <stdio.h>

int main(){

 int arr1[50][50],brr1[50][50],i,j,r,c;

    printf("\nInput the rows and columns of the matrix : ");

    scanf("%d %d",&r,&c);

    printf("Input elements in the first matrix :\n");

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

       for(j=0;j<c;j++){

                printf("element - [%d],[%d] : ",i,j);
```

```c
            scanf("%d",&arr1[i][j]);

        }

    }

        printf("\nThe matrix is :\n");

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

            printf("\n");

            for(j=0;j<c;j++)

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

   for(j=0;j<c;j++){

           brr1[j][i]=arr1[i][j];}

   }

   printf("\n\nThe transpose of a matrix is : ");

   for(i=0;i<c;i++){

   printf("\n");

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

      printf("%d\t",brr1[i][j]); }

 }

   printf("\n\n");

   return 0;

}
```

```
Input the rows and columns of the matrix : 2

2

Input elements in the first matrix :

element - [0],[0] : 1

element - [0],[1] : 2

element - [1],[0] : 3
```

```
element - [1],[1] : 4


The matrix is :



1        2

3        4



The transpose of a matrix is :

1        3

2        4
```

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

#include <stdio.h>

int main() {

   int i,j,arr1[50][50],sum=0,n,m=0;

      printf("Input the size of the square matrix : ");

  scanf("%d", &n);

   m=n;

      printf("Input elements in the first matrix :\n");

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

    for(j=0;j<n;j++){

        printf("element - [%d],[%d] : ",i,j);

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

    }

  }

     printf("The matrix is :\n");

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

     for(j=0;j<n ;j++)

```c
        printf("% 4d",arr1[i][j]);

         printf("\n");

      }

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

    m=m-1;

        for(j=0;j<n ;j++){

     if (j==m) {

       sum= sum+arr1[i][j];

      }

     }

      }

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

   return 0;

 }
```

```
Input the size of the square matrix : 2

Input elements in the first matrix :

element - [0],[0] : 1

element - [0],[1] : 2

element - [1],[0] : 3

element - [1],[1] : 4

The matrix is :

   1    2

   3    4

Addition of the  left Diagonal elements is :5
```

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

#include <stdio.h>

int main(){

```c
int arr1[10][10];

int r1,c1;

int i, j, yn =1;

printf("Input number of Rows for the matrix :");

scanf("%d", &r1);

printf("Input number of Columns for the matrix :");

scanf("%d",&c1);

        printf("Input elements in the first matrix :\n");

   for(i=0;i<r1;i++){

      for(j=0;j<c1;j++){

              printf("element - [%d],[%d] : ",i,j);

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

      }

   }

        printf("The matrix is :\n");

      for(i=0;i<r1;i++){

        for(j=0;j<c1 ;j++)

          printf("% 4d",arr1[i][j]);

         printf("\n");

      }

for(i=0; i<r1; i++){

 for(j=0; j<c1; j++){

      if(arr1[i][j] != 1 && arr1[j][i] !=0){

        yn = 0;

        break;

      }

      }
```

```
    }
  if(yn == 1 )

        printf(" The matrix is an identity matrix.\n\n");

  else

        printf(" The matrix is not an identity matrix.\n\n");

        return 0;

}
```

```
Input number of Rows for the matrix :2

Input number of Columns for the matrix :3

Input elements in the first matrix :

element - [0],[0] : 1

element - [0],[1] : 2

element - [0],[2] : 3

element - [1],[0] : 4

element - [1],[1] : 5

element - [1],[2] : 6

The matrix is :

   1    2    3

   4    5    6

 The matrix is not an identity matrix.
```

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

```c
#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;
```

```c
        }

      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;}
```

```
The given array in matrix form is :
```

```
15   23   31   39

18   26   36   43

25   28   37   48

30   34   39   50

The given value for searching is: 37
```