

## 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>
int main()
{
    int i,n,a[100];
    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("%d ",a[i]);
    }
    printf("\n\nThe values store into the array in reverse are :\n");
    for(i=n-1;i>=0;i--)
    {
        printf("%d ",a[i]);
    }
    printf("\n\n");
    return 0;
}
```

**Output:**

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

**Input 4 number of elements in the array :**

**Element -0 : 2**

**Element -1 : 7**

**Element -2 : 8**

```
Element -3 :5
Element -4 :5
The values store into the array are :
2 7 8 5 5
The values store into the array in reverse are:
5 5 8 7 2
```

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

```
#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];
    }

    printf("Sum : %d\n\n", sum);

    return 0;
}
```

Output:

```
Input the number of elements to store in the array :4
Input 4 number of elements in the array :
Element -0 : 2
Element -1 : 7
Element -2 : 8
```

Element -3 :5  
Sum : 22

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

```
#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("%d ", arr1[i]);
    }
    printf("\n\n elements copied into the second array are :\n");
    for(i=0; i<n; i++)
    {
        printf("%d ", arr2[i]);
    }
    printf("\n\n");

    return 0;
}
```

Output:

Input the number of elements to be stored in the array: 3  
Input 3 number of elements in the array :  
Element -0 : 2

```
Element -1 : 7
Element -2 : 8
The elements stored in the first array are :
2 7 8
elements copied into the second array are:
2 7 8
```

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

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

int arr1[100], arr2[100], arr3[100];
int i,j,n,mm=1,ctr=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",&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++;
        }
    }
    printf("The number of duplicate elements found in the array is: %d\n", ctr);

    printf("\n\n");

    return 0;
}

```

**Output:**  
**Input the number of elements to be stored in the array: 5**  
**Input 3 number of elements in the array :**  
**Element -0 : 1**  
**Element -1 : 3**  
**Element -2 : 1**  
**Element -3 : 3**  
**Element -4 : 8**  
**The number of duplicate elements found in the array is: 2**

## **5. find the maximum and minimum element in an array.**

```

#include <stdio.h>
#define MAX_SIZE 100
int main()
{
    int arr[MAX_SIZE];
    int i, max, min, size;
    printf("Enter size of the array: ");
    scanf("%d", &size);
    printf("Enter elements in the array: ");
    for(i=0; i<size; i++)
    {
        scanf("%d", &arr[i]);
    }
}

```

```

max = arr[0];
min = arr[0];
for(i=1; i<size; i++)
{
    if(arr[i] > max)
    {
        max = arr[i];
    }
    if(arr[i] < min)
    {
        min = arr[i];
    }
}
printf("Maximum element = %d\n", max);
printf("Minimum element = %d", min);

return 0;
}

```

**Output:**

```

Enter size of the array:10
Enter elements in the array: 3 4 67 89 100 43 98 34 11 10
Maximum element = 100
Minimum element = 3

```

## 6. separate odd and even integers in separate arrays.

```

#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 :");
    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");

return 0;
}

```

**Output:**  
**Input the number of elements : 5**  
**Input 5 elements in the array :**  
**Element -0 : 1**  
**Element -1 : 4**  
**Element -2 : 8**  
**Element -3 : 3**  
**Element -4 : 14**  
**The Even elements are :**  
**4 8 14**  
**The Odd elements are :**  
**1 3**

## 7. insert New value in the array.

```

#include <stdio.h>
int main()
{
    int array[100], position, c, n, value;
    printf("Enter number of elements in array\n");
    scanf("%d", &n);

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

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

    printf("Enter the location where to insert element\n");
    scanf("%d", &position);

    printf("Enter the value to insert\n");
    scanf("%d", &value);

    for (c = n - 1; c >= position - 1; c--)
        array[c+1] = array[c];

    array[position-1] = value;

    printf("final array is\n");

    for (c = 0; c <= n; c++)
        printf("%d\n", array[c]);
    return 0;
}

```

**Output:**

```

Enter number of elements in array 5
Enter 5 elements
3 4 7 6 1
Enter the location where to insert element
2
Enter the value to insert
60
final array is
3
60
4
7

```



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

```
#include <stdio.h>
int main()
{
    int arr1[50],i,pos,n;
    printf("Input the size of array : ");
    scanf("%d", &n);
    printf("Input %d elements :\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("\n new array after deletion is : ");
    for(i=0;i<n;i++)
    {
        printf(" %d",arr1[i]);
    }
    printf("\n\n");
    return 0;
}
```

**Output:**

**Input the size of array : 5**

**Input 5 elements :**

**Element -0 : 1**

**Element -1 : 4**

```
Element -2 : 8
Element -3 : 3
Element -4 : 14
Input the position where to delete : 3
new array after deletion is : 1 4 3 14
```

**9. find the second largest element in an array.**

```
#include <stdio.h>
#include <limits.h>
#define MAX_SIZE 1000
int main()
{
    int arr[MAX_SIZE], size, i;
    int max1, max2;
    printf("Enter size of the array : ");
    scanf("%d", &size);
    printf("Enter elements in the array: ");
    for(i=0; i<size; i++)
    {
        scanf("%d", &arr[i]);
    }

    max1 = max2 = INT_MIN;

    for(i=0; i<size; i++)
    {
        if(arr[i] > max1)
        {
            max2 = max1;
            max1 = arr[i];
        }
        else if(arr[i] > max2 && arr[i] < max1)
        {
            max2 = arr[i];
        }
    }
    printf("Second largest = %d", max2);
    return 0;}

```

```
output:
Enter size of the array : 4
```

Enter elements in the array:4 6 8 9  
Second largest = 8

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

### 11. multiplication of two square Matrices

```
#include <stdio.h>
int main()
{
    int
    arr1[50][50],arr2[50][50],arr3[50][50],i,j,k,row1,col1,row2,col2,sum
    =0;

    printf("\nInput the rows and columns of first matrix : ");
    scanf("%d %d",&row1,&col1);
    printf("\nInput the rows and columns of second matrix : ");
    scanf("%d %d",&row2,&col2);
    if(col1!=row2)
    {
        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<row1;i++)
        {
            for(j=0;j<col1;j++)
            {
                printf("element - [%d],[%d] : ",i,j);
                scanf("%d",&arr1[i][j]);
            }
        }
        printf("Input elements in the second matrix :\n");
        for(i=0;i<row2;i++)
        {
            for(j=0;j<col2;j++)
            {
                printf("element - [%d],[%d] : ",i,j);
                scanf("%d",&arr2[i][j]);
            }
        }
    }
}
```

```

    printf("\nThe First matrix is :\n");
    for(i=0;i<row1;i++)
    {
        printf("\n");
        for(j=0;j<col1;j++)
        printf("%d\t",arr1[i][j]);
    }

    printf("\nThe Second matrix is :\n");
    for(i=0;i<row2;i++)
    {
        printf("\n");
        for(j=0;j<col2;j++)
        printf("%d\t",arr2[i][j]);
    }

    for(i=0;i<row1;i++)
    for(j=0;j<col2;j++)
    arr3[i][j]=0;
    for(i=0;i<row1;i++)    {
        for(j=0;j<col2;j++)

            sum=0;
            for(k=0;k<col1;k++)
            sum=sum+arr1[i][k]*arr2[k][j];
            arr3[i][j]=sum;
        }
    }
    printf("\nThe multiplication of two matrices is : \n");
    for(i=0;i<row1;i++)
    {
        printf("\n");
        for(j=0;j<col2;j++)
        {
            printf("%d\t",arr3[i][j]);
        }
    }
    printf("\n\n");

    return 0;

```

}

**Output:**

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

**Input the rows and columns of second matrix : 2 3**

**Input elements in the first matrix :**

**Element - [0],[0] : 1 1**

**Element - [0],[1] : element-[1],[0] : 2 3**

**Element -[1],[1]:**

**Input element in the second matrix**

**Element - [0],[0] : 1 1**

**Element - [0],[1] : element-[1],[0] : 5 6**

**Element -[1],[1]: element -[1],[1]: 3 3**

**The First matrix is:**

**1 1**

**2 3**

**The Second matrix is :**

**1 1 5**

**6 3 3**

**The multiplication of two matrices is:**

**7 4 8**

**20 11 19**

## **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("\nEnter 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;
}

```

**Output:**

**Enter rows and columns: 2 3**

**Enter matrix elements:**

**Enter elements a11: 5**

**Enter elements a12: 8**

**Enter elements a13: 9**

**Enter elements a21: 1**

**Enter elements a22: 7**

**Enter elements a23: 89**

**Entered matrix:**

**5 8 9**

**1 7 89**

**Transpose of the matrix:**

**5 1**

**8 7**

**9 89**

### 13. find the sum of left diagonals of a matrix.

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

```
void main()
```

```
{
    int i,j,arr1[50][50],sum=0,n,m=0;
        printf("Input the size of the matrix : ");
    scanf("%d", &n);
        m=n;
        printf("Input elements in the 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++)
            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:%d\n",sum);
}
```

```
}
```

**Output:**

**Input the size of the matrix : 2 2**

**Input elements in the matrix :**

**Element -[0],[0] : element -[0],[1] :2 4**

**Element -[1],[0] : element -[1],[1] :6 2**

**The matrix is :**

**2 2**

**4 6**

**Addition of the left Diagonal:8**

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

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

```
void main()
```

```
{
    int arr1[10][10];
    int r1,c1,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");
}

```

**Output:**

```

Input number of Rows for the matrix :3
Input number of Columns for the matrix :3
Input elements in the first matrix :
element - [0],[0] : 1
element - [0],[1] : 0
element - [0],[2] : 0
element - [1],[0] : 0
element - [1],[1] : 1
element - [1],[2] : 0
element - [2],[0] : 0
element - [2],[1] : 0
element - [2],[2] : 1
The matrix is :
  1  0  0
  0  1  0
  0  0  1
The matrix is an 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] = { {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;

}

```