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

**#include <stdio.h>**

**int main()**

**{**

**int arr[10], i;**

**printf("enter the array elements: \n");**

**for(i=0; i<10; i++){**

**printf("enter a[%d]:\n",i);**

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

**}**

**for(i=10; i>=0; i--){**

**printf("%d", arr[i]);**

**}**

**return 0;**

**}**

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

**#include <stdio.h>**

**int main()**

**{**

**int arr[10], i, sum=0;**

**printf("enter the array elements: \n");**

**for(i=0; i<10; i++){**

**printf("enter a[%d]:",i+1);**

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

**sum+=arr[i];**

**}**

**for(i=0; i<10; i++){**

**printf("%d ", arr[i]);**

**}**

**printf("\nthe sum of elements is: %d", sum);**

**return 0;**

**}**

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

**#include <stdio.h>**

**int main()**

**{**

**int arr1[10],arr2[10],n, i;**

**printf("enter the number of array elements: \n");**

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

**printf("enter the elements for array1:\n");**

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

**printf("enter a[%d]:",i+1);**

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

**}**

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

**arr2[i]=arr1[i];**

**}**

**printf("the array 1 is: \n");**

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

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

**}**

**printf("\nthe array 2 is: \n");**

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

**printf("%d ", arr2[i]);**

**}**

**return 0;**

**}**

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

**#include <stdio.h>**

**int main()**

**{**

**int arr[100],n, i, j, count=0;**

**printf("enter the number of array elements: \n");**

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

**printf("enter the elements for array1:\n");**

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

**printf("enter a[%d]:",i+1);**

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

**}**

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

**{**

**for(j=i+1; j<n; j++)**

**{**

**if(arr[i] == arr[j])**

**{**

**count++;**

**break;**

**}**

**}**

**}**

**printf("the array 1 is: \n");**

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

**printf("%d ", arr[i]);**

**}**

**printf("\nthe number of duplicate elements are %d", count);**

**return 0;**

**}**

**5) Find the maximum and the minimum element in the array.**

**#include <stdio.h>**

**int main()**

**{**

**int arr[100],n, i, j, high, low;**

**printf("enter the number of array elements: \n");**

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

**printf("enter the elements for array:\n");**

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

**printf("enter a[%d]:",i+1);**

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

**}**

**high = arr[0];**

**for(i=1; i<n; i++)**

**{**

**if(high<arr[i]){**

**high = arr[i];**

**}**

**}**

**low = arr[0];**

**for(i=1; i<n; i++)**

**{**

**if(low>arr[i]){**

**low = arr[i];**

**}**

**}**

**printf("the array is: \n");**

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

**printf("%d ", arr[i]);**

**}**

**printf("\nthe highest element is %d", high);**

**printf("\nthe lowest element is %d", low);**

**return 0;**

**}**

**6) Separate even and odd elements from an array.**

**#include<stdio.h>**

**int main()**

**{**

**int arr[100],n, i, j=0,k=0, a\_even[100], a\_odd[100];**

**printf("enter the number of array elements: \n");**

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

**printf("enter the elements for array:\n");**

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

**printf("enter a[%d]:",i+1);**

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

**}**

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

**{**

**if(arr[i] % 2==0){**

**a\_even[j] = arr[i];**

**j++;**

**} else{**

**a\_odd[k] = arr[i];**

**k++;**

**}**

**}**

**printf("the array is: \n");**

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

**printf("%d ", arr[i]);**

**}**

**printf("\nthe even array is: \n");**

**for(i=0; i<j; i++){**

**printf("%d ", a\_even[i]);**

**}**

**printf("\nthe odd array is: \n");**

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

**printf("%d ", a\_odd[i]);**

**}**

**return 0;**

**}**

**7) Insert new value in the array.**

**#include<stdio.h>**

**int main()**

**{**

**int arr[100],n, i, pos, num;**

**printf("enter the number of array elements: \n");**

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

**printf("enter the elements for array:\n");**

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

**printf("enter a[%d]:",i+1);**

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

**}**

**printf("the array is: \n");**

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

**printf("%d ", arr[i]);**

**}**

**printf("\nenetr the position between 0 to %d: ", n);**

**scanf("%d", &pos);**

**printf("\n enter the number to be inserted: ");**

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

**for(i=n-1; i>=pos; i--)**

**{**

**arr[i] = arr[i-1];**

**}**

**arr[pos-1] = num;**

**printf("the array is: \n");**

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

**printf("%d ", arr[i]);**

**}**

**return 0;**

**}**

8) **Delete an array element.**

**#include<stdio.h>**

**int main()**

**{**

**int arr[100],n, i, pos;**

**printf("enter the number of array elements: \n");**

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

**printf("enter the elements for array:\n");**

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

**printf("enter a[%d]:",i+1);**

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

**}**

**printf("the array is: \n");**

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

**printf("%d ", arr[i]);**

**}**

**printf("\nenetr the position between 1 to %d: ", n);**

**scanf("%d", &pos);**

**if(pos>n){**

**printf("deletion not possible!!");**

**}**

**else{**

**for(i=pos-1; i<=n-1; i++)**

**{**

**arr[i] = arr[i+1];**

**}**

**}**

**printf("the array is: \n");**

**for(i=0; i<n-1; i++){**

**printf("%d ", arr[i]);**

**}**

**return 0;**

**}**

**9) Find the second largest element in an array.**

**#include <stdio.h>**

**int main()**

**{**

**int arr[100], n, i, j, high, second\_high;**

**printf("enter the size of array: \n");**

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

**printf("\nenter the array elements: \n");**

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

**printf("enter arr[%d]: ", i);**

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

**}**

**high=second\_high=arr[0];**

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

**for(j=i+1; j<n; j++){**

**if(high< arr[j]){**

**second\_high = high;**

**high = arr[j];**

**}**

**else if(arr[j]>second\_high && arr[j]< high){**

**second\_high = arr[j];**

**}**

**}**

**}**

**printf("the array: \n");**

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

**printf(" %d ",arr[i] );**

**}**

**printf("\n the highest number is = %d ", high);**

**printf("\n the second highest number is = %d ", second\_high);**

**return 0;**

**}**

**10) Find the median of sorted array**

**#include <stdio.h>**

**int main()**

**{**

**int arr1[100], arr2[100],merge[100], i, j,k,m, n, temp;**

**printf("enter the size of the array1: \n");**

**scanf("%d", &m);**

**printf("enter the size of the array2: \n");**

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

**printf("enter the elements for array1: \n");**

**for(i=0; i<m; i++){**

**printf("enter arr[%d]: ", i);**

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

**}**

**printf("enter the elements for array2: ");**

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

**printf("enter arr[%d]: ", i);**

**scanf("%d",& arr2[i]);**

**}**

**i = 0;**

**j = 0;**

**k = 0;**

**// merge two arrays**

**while(i<m && j<n){**

**if(arr1[i] <= arr2[j]){**

**merge[k] = arr1[i];**

**i++;**

**k++;**

**}**

**else{**

**merge[k] = arr2[j];**

**k++;**

**j++;**

**}**

**}**

**while(i<m){**

**merge[k] = arr1[i];**

**i++;**

**k++;**

**}**

**while(j<n){**

**merge[k] = arr2[j];**

**j++;**

**k++;**

**}**

**// print the merged array**

**printf("the merged array is: \n");**

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

**printf("%d ", merge[i]);**

**}**

**// sort the merged array**

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

**for(j=i+1; j<(m+n); j++){**

**if(merge[i] > merge[j]){**

**temp = merge[i];**

**merge[i] = merge[j];**

**merge[j] = temp;**

**}**

**}**

**}**

**// the array after sorting**

**printf("\nthe merged array after sorting is: \n");**

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

**printf("%d ", merge[i]);**

**}**

**// finding median**

**float median=0;**

**int length = m+n;**

**if(length%2 == 0)**

**median = (merge[(length-1)/2] + merge[length/2])/2.0;**

**else**

**median = merge[length/2];**

**printf("\nthe median is %f ", median);**

**return 0;**

**}**

**11) Matrix multiplication**

**#include<stdio.h>**

**#include<stdlib.h>**

**int main(){**

**int a[10][10],b[10][10],mul[10][10],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 second matrix element=\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++)**

**{**

**mul[i][j]=0;**

**for(k=0;k<c;k++)**

**{**

**mul[i][j]+=a[i][k]\*b[k][j];**

**}**

**}**

**}**

**//for printing result**

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

**{**

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

**{**

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

**}**

**printf("\n");**

**}**

**return 0;**

**}**

**12) Transpose of a 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("\nMatrix: \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) 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 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++)**

**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);**

**}**

**14) Check whether a matrix is identity matrix.**

**#include <stdio.h>**

**void main()**

**{**

**int i,j,arr1[50][50],sum=0,n,m, identity=1;**

**printf("Input the row size of the matrix : ");**

**scanf("%d", &m);**

**printf("Input the col size of the matrix : ");**

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

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

**for(i=0;i<m;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<m;i++)**

**{**

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

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

**printf("\n");**

**}**

**for(i=0; i<m; i++){**

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

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

**identity = 0;**

**break;**

**}**

**}**

**}**

**if(identity == 1){**

**printf("\n identical!!");**

**} else{**

**printf("not identical!");**

**}**

**return 0;**

**}**

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

**#include <stdio.h>**

**int searchElement(int arr[4][4], int n, int x)**

**{**

**int i = 0, j = n-1;**

**while ( i < n && j >= 0 )**

**{**

**if ( arr[i][j] == x )**

**{**

**printf("\nThe element Found at the position in the matrix is: %d, %d", i, j);**

**return 1;**

**}**

**if ( arr[i][j] < x )**

**j--;**

**else**

**i++;**

**}**

**printf("\nThe given element not found in the matrix.");**

**return 0;**

**}**

**int main()**

**{**

**int arr[4][4] = { {15, 23, 31, 39},**

**{18, 26, 36, 43},**

**{25, 28, 37, 48},**

**{30, 34, 39, 50},**

**};**

**int i,j,element=36;**

**printf("The given array in matrix form is : \n");**

**for(i = 0; i < 4; i++)**

**{**

**for (j=0;j<4;j++)**

**{**

**printf("%d ", arr[i][j]);**

**}**

**printf("\n");**

**}**

**printf("The given value for searching is: %d",element);**

**searchElement(arr, 4, element);**

**return 0;**

**}**