**1.Array 20 input from user**

#include<stdio.h>

main()

{

int a[20];

printf("enter array elements:: ");

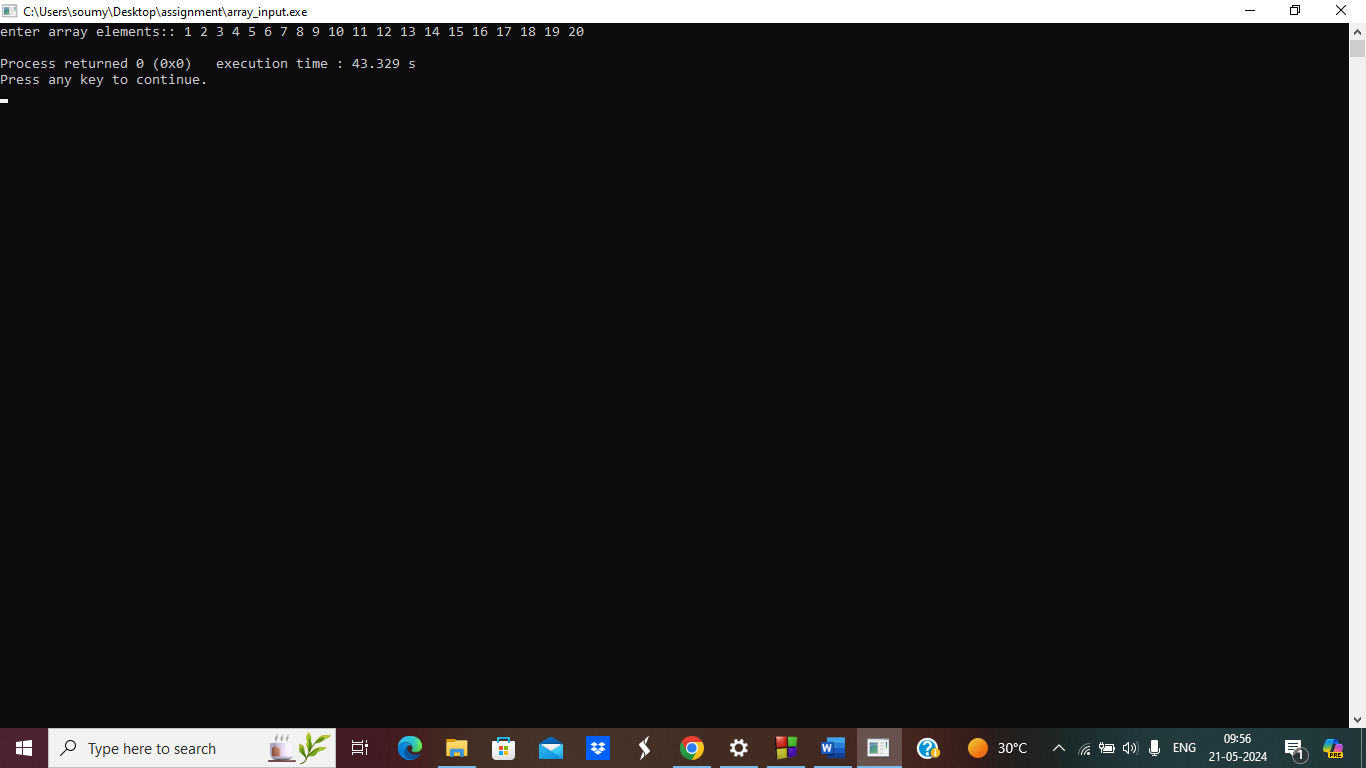
for(int i=0;i<20;i++)

{

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

}

}



**2.Print array**

#include<stdio.h>

main()

{

int n;

printf("enter no of array elements: ");

scanf("%d",&n);

int a[n];

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

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

{

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

}

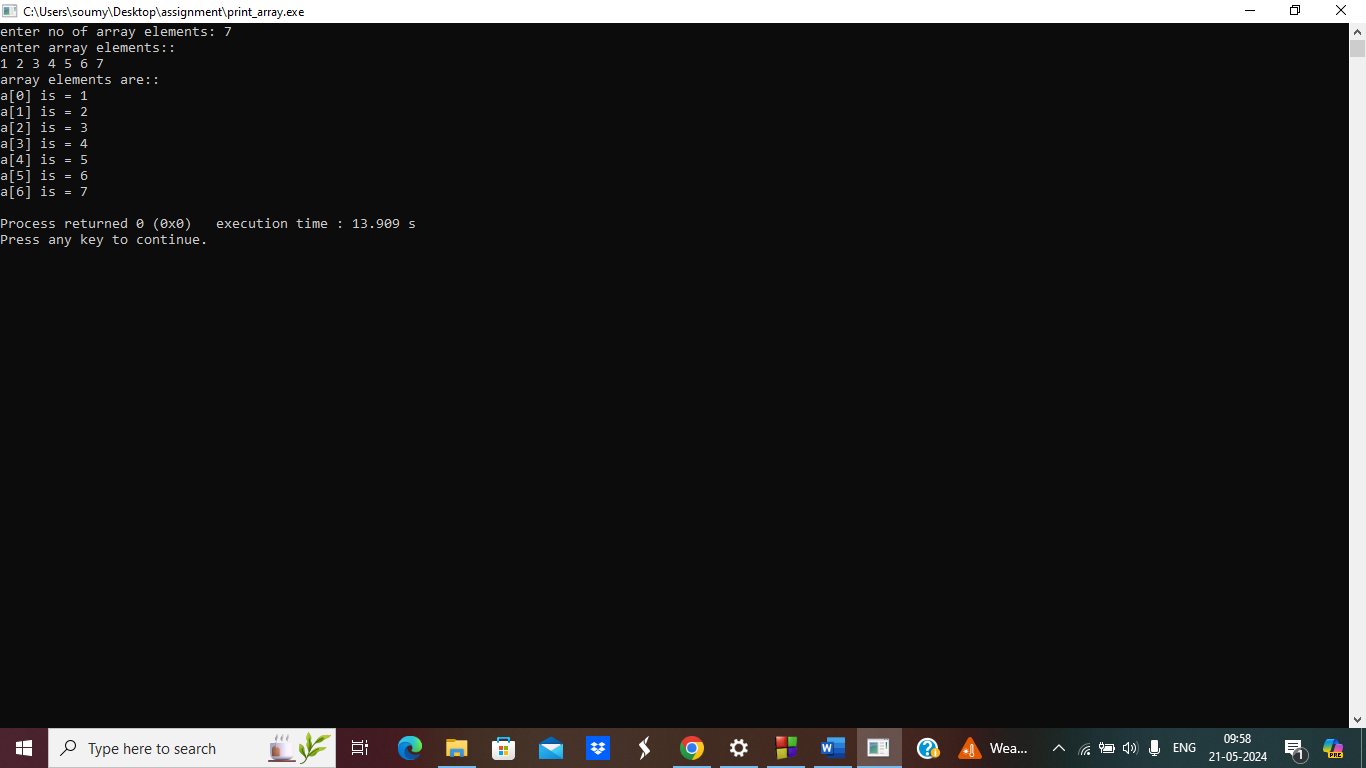
printf("array elements are::\n");

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

{

printf("a[%d] is = %d\n",i,a[i]);

} }



**3.delete an array element**

#include<stdio.h>

main()

{

int n,pos;

printf("enter no of array elements: ");

scanf("%d",&n);

int a[n];

printf("enter array elements:: ");

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

{

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

}

printf("array elements are::\n");

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

{

printf("a[%d] is = %d\n",i,a[i]);

}

printf("enter the element position you want to delete ::");

scanf("%d",&pos);

a[pos]=0;

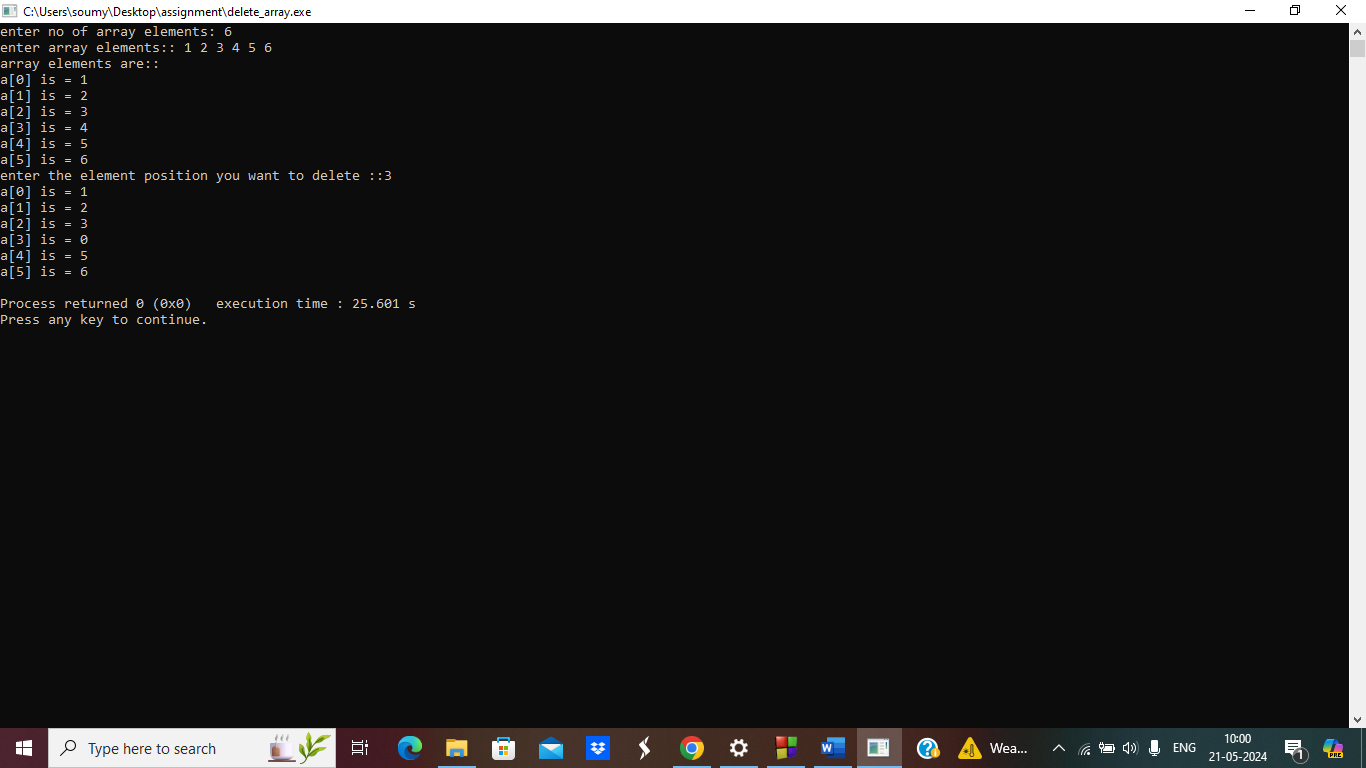
for(int i=0;i<n;i++)

{

printf("a[%d] is = %d\n",i,a[i]);

}

}



**4.Duplicate element in an array**

#include<stdio.h>

main()

{

int a[6],n;

printf("enter array elements:: ");

for(int i=0;i<6;i++)

{

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

}

printf("array elements are::");

for(int i=0;i<6;i++)

{

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

}

for(int i=0;i<6;i++)

{

for(int j=i+1;j<6;j++)

{

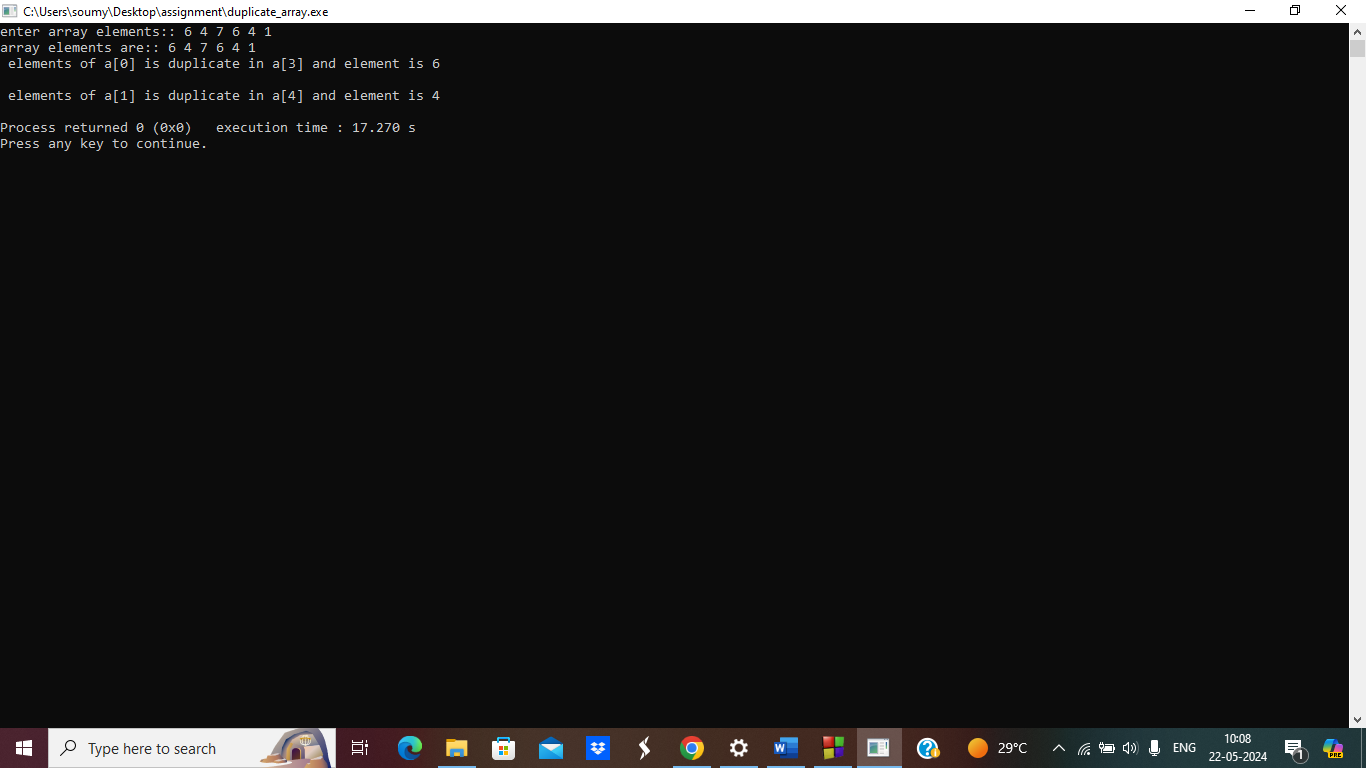
if(a[i]==a[j])

printf("\n elements of a[%d] is duplicate in a[%d] and element is %d\n",i,j,a[i]);

}

}

}



**5. Searching an element in an array**

#include<stdio.h>

main()

{

int a[10],element,flag=0;

printf("enter array elements:: ");

for(int i=0;i<10;i++)

{

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

}

printf("array elements are::\n");

for(int i=0;i<10;i++)

{

printf("a[%d] is = %d\n",i,a[i]);

}

printf("enter the element you want to search ::");

scanf("%d",&element);

for(int i=0;i<10;i++)

{

if(a[i]==element)

{

flag=1;

}

}

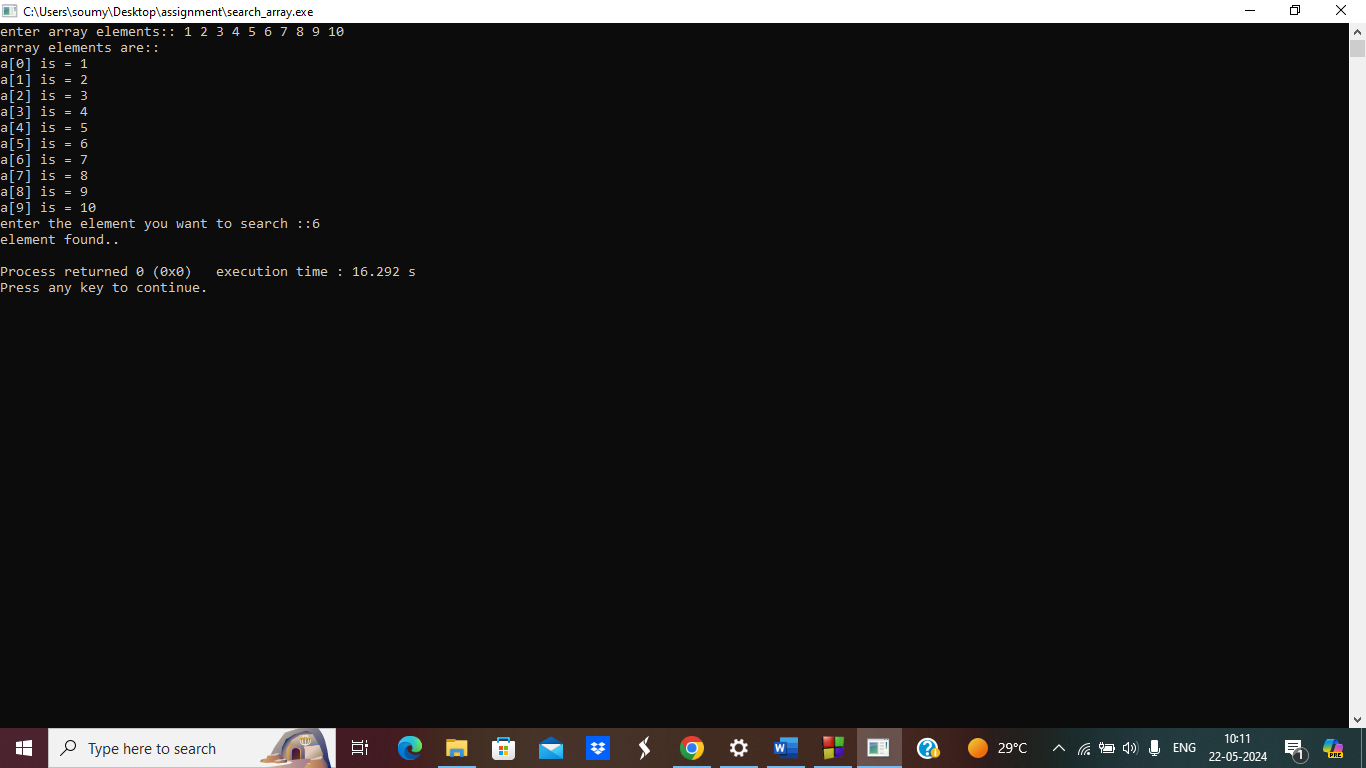
if(flag==1)

printf("element found..\n");

else

printf("element not found..\n");

}



**6.Implement bubble sort**

#include<stdio.h>

void bubble\_sort(int \*a,int n)

{

int i,j,temp;

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

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

{

if(\*(a+j)>\*(a+(j+1)))

{

temp=\*(a+j);

\*(a+j)=\*(a+(j+1));

\*(a+(j+1))=temp;

}

}

}

int main()

{

int n;

printf("enter no of elements in array--> ");

scanf("%d",&n);

int arr[n];

printf("enter array elements--> ");

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

{

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

}

bubble\_sort(arr,n);

printf("Elements after sorting --> ");

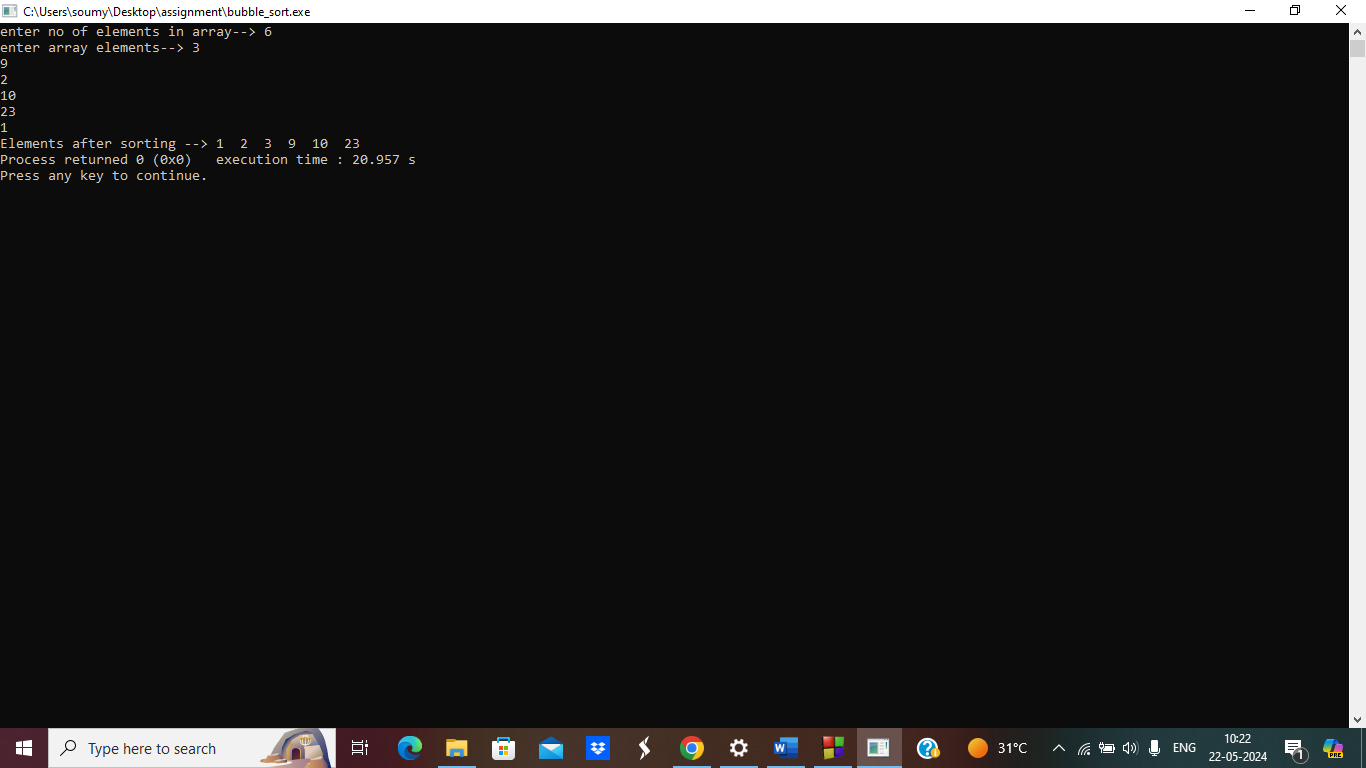
for(int i=0;i<n;i++)

{

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

}

}



**7.Implement merge sort**

#include <stdio.h>

#include <stdlib.h>

void Merge(int arr[], int left, int mid, int right)

{

int i, j, k;

int size1 = mid - left + 1;

int size2 = right - mid;

int Left[size1], Right[size2];

for (i = 0; i < size1; i++)

Left[i] = arr[left + i];

for (j = 0; j < size2; j++)

Right[j] = arr[mid + 1 + j];

i = 0;

j = 0;

k = left;

while (i < size1 && j < size2)

{

if (Left[i] <= Right[j])

{

arr[k] = Left[i];

i++;

}

else

{

arr[k] = Right[j];

j++;

}

k++;

}

while (i < size1)

{

arr[k] = Left[i];

i++;

k++;

}

while (j < size2)

{

arr[k] = Right[j];

j++;

k++;

}

}

void Merge\_Sort(int arr[], int left, int right)

{

if (left < right)

{

int mid = left + (right - left) / 2;

Merge\_Sort(arr, left, mid);

Merge\_Sort(arr, mid + 1, right);

Merge(arr, left, mid, right);

}

}

int main()

{

int size;

printf("Enter the size: ");

scanf("%d", &size);

int arr[size];

printf("Enter the elements of array: ");

for (int i = 0; i < size; i++)

{

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

}

Merge\_Sort(arr, 0, size - 1);

printf("The sorted array is: ");

for (int i = 0; i < size; i++)

{

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

}

printf("\n");

return 0;

}

