

**DSA Lab**

**Name: Siddhi Parekh**

**Reg No.: 221071047**

**Batch: C**

**SY Comps**

## **Experiment No.: 1**

### **AIM:**

The primary objective of this experiment is to develop and implement a modular C++ program for array manipulation. The program allows users to dynamically input, display, search for elements, sort, delete, and insert elements in an integer array.

### **THEORY:**

The input function prompts the user to enter elements into an array of a specified size, while the output function displays the array elements.

The search function enables users to input an element and checks if it exists in the array, providing the index if found.

The sort function utilizes the bubble sort algorithm to arrange the array elements in ascending order.

The del function deletes an element from the array based on the user-specified index, shifting subsequent elements to fill the gap.

Conversely, the insert function inserts a user-provided element at a specified index, shifting existing elements to accommodate the new entry.

## **ALGORITHM:**

Step 0: Start

Step 1: input(array, size)

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

Take input of array elements

Step 2: output(array, size)

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

Print all array elements

Step 3: search(array, size)

if(key==array\_element)

Print the index of that array element

else

Iterate till end of array

Step 4: sort(array, size)

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

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

if(array[i]>array[j]) swap the 2 values

display the elements of the sorted array \

Step 5: delete(array, size)

Take input of the index no. to be deleted.

From that index number till end of array, shift the numbers by 1 index towards the beginning.

Step 6: Insertion(array, size)

Can be done only after deletion

Take input of index no. where value is to be inserted

Counting from that index till end of array, shift the values by 1 index towards the end and insert the key value.

Step 7: Inside the int main

Make an array, take input of its size, and call the above functions in order.

## **EXAMPLE:**

Suppose you want to manipulate an array of integers. Here's an example interaction with the program:

1. Enter the size of the array: 5
2. Enter the elements of the array  
8 3 5 2 1
3. Displaying elements  
8 3 5 2 1
4. Enter the element you want to search: 5  
Element 5 is at index 2 of the array

5. The array sorted in ascending order is  
1 2 3 5 8
6. Enter the index no. of the array you want to delete: 3  
Displaying elements after deletion  
1 2 3 8
7. Enter the no. you want to insert: 6  
Enter the index no. where you want to insert: 2  
Displaying elements after insertion  
1 2 6 3 8

In this example, the user first specifies the size of the array (step 1), then inputs the elements of the array (step 2). The program displays the entered elements (step 3) and allows the user to search for a specific element (step 4).

After that, the program sorts the array in ascending order (step 5), deletes an element at a specified index (step 6), and finally inserts a new element at a specified index (step 7).

## **CONCLUSION:**

Thus we have successfully implemented simple array operations in this experiment.

## **CODE:**

```
#include<iostream>
```

```
using namespace std;
```

```
void input(int a[], int size)
```

```
{  
    cout<<"Enter the elements of the array"<<endl;  
    for(int i=0; i<size; i++)  
    {  
        cin>>a[i];  
    }  
}
```

```
void output(int b[],int size)
```

```
{  
    cout<<"Displaying elements"<<endl;  
    for(int i=0; i<size; i++)  
    {  
        cout<<b[i]<<" ";  
    }  
}
```

```
void search(int c[], int size)
```

```
{  
    int element;  
    int n=0;  
    int i = 0;  
    cout<<"\nEnter the element you want to search"<<endl;  
    cin>>element;  
  
    for(i=0; i<size; i++)  
    {  
        if(c[i] == element)  
        {  
            n=1;  
            break;  
        }  
  
        else
```

```

{
n=0;

}
}

if(n==1)
{
cout<<"Element "<<element<<" is at index "<<i<<" of the array"<<endl;
}

else
{
cout<<"No such element exists in the array"<<endl;
}
}

```

```

void sort(int d[],int size)
{
    cout<<"The array sorted in ascending array is"<<endl;

    for(int i=0; i<size; i++)
    {
        for(int j=i+1; j<size; j++)
        {
            if(d[i] > d[j])
            {
                int temp;
                temp = d[i];
                d[i]=d[j];
                d[j]=temp;
            }
        }
    }

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

```

```
        cout<<d[i]<<" ";
    }
}
```

```
void del(int e[], int size)
```

```
{
    // int num;
    // cout<<"Enter the number of elements you want to delete"<<endl;\
    // cin>>num;

    int index;
    cout<<"\nEnter the index no. of the array you want to delete"<<endl;
    cin>>index;

    for(int i=0; i<size-1; i++)
    {
        e[index+i] = e[index+i+1];
    }

    //size = size - 1;

    cout<<"Displaying elements"<<endl;
    for(int i=0; i<size-1; i++)
    {
        cout<<e[i]<<" ";
    }
}
```

```
void insert(int f[], int size)
```

```
{
    int index;
    int ele;

    cout<<"\nEnter the no. you want to insert"<<endl;
    cin>>ele;
    cout<<"Enter the index no. where you want to insert"<<endl;
    cin>>index;
    //size = size+1;
```

```

for(int i=1; i<=((size-1)-index); i++)
{
    f[size-i]=f[size-(i+1)];
}

f[index]=ele;

cout<<"Displaying elements after insertion"<<endl;
for(int i=0; i<size; i++)
{
    cout<<f[i]<<" ";
}

}

int main()
{
    int size=0;

    cout<<"Enter the size of the array"<<endl;
    cin>>size;

    int arr[size];

    input(arr,size);

    output(arr,size);
    search(arr,size);
    sort(arr,size);
    del(arr,size);
    insert(arr,size);

}

```



## OUTPUT:

```
•ys$ cd "/home/siddhi/dsa_lab_sy/expt1---arrays/" && g++ main.cpp -o main && "/home/siddhi/dsa_lab_sy/expt1---arrays/"main
Enter the size of the array
10
Enter the elements of the array
6 5 7 8 9 1 4 2 3 0
Displaying elements
6 5 7 8 9 1 4 2 3 0
Enter the element you want to search
10
No such element exists in the array
The array sorted in ascending array is
0 1 2 3 4 5 6 7 8 9
Enter the index no. of the array you want to delete
3
Displaying elements
0 1 2 4 5 6 7 8 9
Enter the no. you want to insert
10
Enter the index no. where you want to insert
3
Displaying elements after insertion
0 1 2 10 4 5 6 7 8 9 (base) siddhi@siddhi-Inspiron-3576:~/dsa_lab_sy/expt1---arra
o (base) siddhi@siddhi-Inspiron-3576:~/dsa_lab_sy/expt1---arrays$
```