

## 1. Write a program to implement Array.

```
#include <iostream.h>

#include <conio.h>

int main()
{
    int n;
    cout << "Enter the size of the array: " << endl;
    cin >> n;
    int values[n];
    cout << "Enter " << n << " integers: " << endl;
    for (int i = 0; i < n; ++i)
    {
        cin >> values[i];
    }

    cout << "The integers are: ";
    for (int i = 0; i < n; ++i)
    {
        cout << values[i] << " ";
    }
    return 0;
}
```

**Output:-**

```
Enter the size of the array: 5
Enter 5 integers: 12 14 34 45 67
The integers are: 12 14 34 45 67
```

## 2. Write a program to implement Bubble sort.

```
#include <iostream.h>

#include <conio.h>

void bubble(int arr[], int size)
{
    for (int i = 0; i < size; i++)
    {
        for (int j = 0; j < size - 1; j++)
        {
            if (arr[j] > arr[j + 1])
            {
                int temp;
                temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        }
    }
}

void main()
{
    clrscr();
    int arr[6] = {5, 3, 6, 1, 2, 4};
    bubble(arr, 6);
    for (int i = 0; i < 6; i++)
    {
        cout << arr[i] << " ";
    }
    getch();
}
```

}

**Output:-**

**1 2 3 4 5 6**

### 3. Write a program to implement Selection sort.

```
#include <iostream.h>

#include <conio.h>

void selc(int arr[], int size)
{
    for (int i = 0; i < size; i++)
    {
        int mini = i;
        for (int j = i + 1; j < size; j++)
        {
            if (arr[mini] > arr[j])
            {
                int temp;
                temp = arr[mini];
                arr[mini] = arr[j];
                arr[j] = temp;
            }
        }
    }
}

void main()
{
    clrscr();
    int arr[6] = {6, 4, 2, 1, 3, 5};
    selc(arr, 6);
    for (int i = 0; i < 6; i++)
    {
        cout << arr[i] << " ";
    }
}
```

```
}  
    getch();  
}
```

**Output:-**

**1 2 3 4 5 6**

#### 4. Write a program to implement Insertion sort.

```
#include <iostream.h>
#include <conio.h>
void inse(int arr[], int size)
{
    for (int i = 1; i < size; i++)
    {
        int temp = arr[i];
        int j = i - 1;
        while (j >= 0 && temp <= arr[j])
        {
            arr[j + 1] = arr[j];
            j = j - 1;
        }
        arr[j + 1] = temp;
    }
}
void main()
{
    clrscr();
    int arr[6] = {5, 1, 6, 3, 4, 2};
    inse(arr, 6);
    for (int i = 0; i < 6; i++)
    {
        cout << arr[i] << " ";
    }
    getch();
}
```

```
}
```

**Output:-**

**1 2 3 4 5 6**

## 5. Write a program to implement Quick sort.

```
#include <iostream.h>

#include <conio.h>

void swap(int* a, int* b)
{
    int t = *a;
    *a = *b;
    *b = t;
}

int partition (int arr[], int low, int high)
{
    int pivot = arr[high]; // pivot
    int i = (low - 1);

    for (int j = low; j <= high- 1; j++)
    {
        if (arr[j] <= pivot)
        {
            i++; // increment index of smaller element
            swap(&arr[i], &arr[j]);
        }
    }
    swap(&arr[i + 1], &arr[high]);
    return (i + 1);
}

void quickSort(int arr[], int low, int high)
{
    if (low < high)
    {
        int pivot = partition(arr, low, high);

        quickSort(arr, low, pivot - 1);
        quickSort(arr, pivot + 1, high);
    }
}

void displayArray(int arr[], int size)
```



```

{
    int i;
    for (i=0; i < size; i++)
        cout<<arr[i]<<"\t";

}

int main()
{
    int arr[] = {12,23,3,43,51,35,19,45};
    int n = sizeof(arr)/sizeof(arr[0]);
    cout<<"Input array"<<endl;
    displayArray(arr,n);
    cout<<endl;
    quickSort(arr, 0, n-1);
    cout<<"Array sorted with quick sort"<<endl;
    displayArray(arr,n);
    return 0;
}

```

**Output:-**

**Input array**

**12    23    3    43    51    35    19    45**

**Array sorted with quicksort**

**3    12    19    23    35    43    45    51**

## 6. Write a program to implement Linear Search.

```
#include <iostream.h>

#include <conio.h>

void linearSearch(int a[], int n) {

    int temp = -1;

    for (int i = 0; i < 5; i++) {

        if (a[i] == n) {

            cout << "Element found at position: " << i + 1 << endl;

            temp = 0;

            break;

        }

    }

    if (temp == -1) {

        cout << "No Element Found" << endl;

    }

}

int main() {

    int arr[5];

    cout << "Please enter 5 elements of the Array" << endl;

    for (int i = 0; i < 5; i++) {

        cin >> arr[i];

    }

    cout << "Please enter an element to search" << endl;

    int num;

    cin >> num;
```

```
linearSearch(arr, num);  
return 0;  
}
```

**Output:-**

```
Please enter 5 elements of the Array  
23 56 78 90 79  
Please enter an element to search  
78  
Element found at position: 3
```

```
Please enter 5 elements of the Array  
12 34 70 40 45  
Please enter an element to search  
50  
No Element Found
```

## 7. Write a program to implement Binary Search.

```
#include <iostream.h>

#include <conio.h>

int main()
{
    int i, arr[10], num, first, last, middle;
    cout<<"Enter 10 Elements (in ascending order): ";
    for(i=0; i<10; i++)
        cin>>arr[i];
    cout<<"\nEnter Element to be Search: ";
    cin>>num;
    first = 0;
    last = 9;
    middle = (first+last)/2;
    while(first <= last)
    {
        if(arr[middle]<num)
            first = middle+1;
        else if(arr[middle]==num)
        {
            cout<<"\nThe number, "<<num<<" found at Position "<<middle+1;
            break;
        }
        else
            last = middle-1;
        middle = (first+last)/2;
    }
    if(first>last)
        cout<<"\nThe number, "<<num<<" is not found in given Array";
    cout<<endl;
    return 0;
}
```

**Output:-**

```
Enter 10 Elements (in ascending order): 11 12 13 14 15 16 17 18 19 20
Enter Element to be Search: 13
The number, 13 found at Position 3
```

**Enter 10 Elements (in ascending order): 11 12 13 14 15 16 17 18 19 20**

**Enter Element to be Search: 22**

**The number, 22 is not found in given Array**