DSA SERIES

- Learn Coding

Topic to be Covered today

Selection sorting

Quick Sorting

LETS START TODAY'S LECTURE

Selection sorting

- Simple sorting algorithm that works on the comparision.

- It selects the minimum element from the unsorted portion and places it at the beginning – one element at a time.

Repeat the process until the array is sorted.

Procedure :

Example :

[29 10 14 37 13]

[29 10 14 37 13]

[10 29 14 37 13]

[10 29 14 37 13]

[10 13 14 37 29]

[10 13 14 37 29]

[10 13 14 37 29]

[10 13 14 37 29]

[10 13 14 29 37]

```cpp
#include <iostream>

using namespace std;

int main()
{

    int arr[5] = {29, 10, 14, 37, 13};

    cout << "Printing the array before sorting : ";
    for (int i = 0; i < 5; i++)
    {
        cout << arr[i] << " ";
    }
    cout << endl;

    // Sorting logic

    int n = 5;
    //  for(int i = 0;i<n-1;i++){
    //     int minVal = arr[i];
    //     int minIndex = i;
```

```
//     for(int j = i+1;j<n;j++){
//         if(minVal > arr[j] ){
//             minVal = arr[j];
//             minIndex = j;
//         }
//     }

//     arr[minIndex]=arr[i];
//     arr[i]=minVal;
//  }

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

    for (int j = i + 1; j < n; j++)
    {
        if (arr[j] < arr[minIndex])
        {

            minIndex = j;
        }
```

```
        }
        swap(arr[i], arr[minIndex]);
    }

    cout << "Printing the array after sorting : ";
    for (int i = 0; i < 5; i++)
    {
        cout << arr[i] << " ";
    }
    cout << endl;

    return 0;
}
```

Quick Sort

Quick Sort is a **divide and conquer** algorithm that:

1.Selects a **pivot** element.

2.Partitions the array:

1. All elements smaller than the pivot go to the left.
2. All elements greater go to the right.

3.Recursively sorts the **left** and **right** subarrays.

```
#include<iostream>
#include<vector>

using namespace std;

int partition(vector<int> &arr , int low,int high){
    int pivot  = arr[high];
    int i = low-1;

    for(int j = low;j<high;j++){
        if(arr[j]<pivot){
            i++;
            swap(arr[i],arr[j]);

        }
    }

    swap(arr[i+1],arr[high]);
    return i+1;
}
```

```
void quickSort(vector<int> &arr , int low,int high){
    if(low<high){
        int pi = partition(arr,low,high);

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

int main(){

    vector<int> arr ={ 6,3,5,4,8,11,2,9};
    int n = arr.size();

    quickSort(arr,0,n-1);

    for(int i =0;i<arr.size();i++){
        cout<<arr[i]<<" ";
    }
    cout<<endl;


}
```

Learn coding

THANK YOU