# WORKSHEET 5

**Student Name: Harsh Kumar** **UID: 22BCS15754**

**Branch: CSE** **Section/Group: FL_IOT_603'B'**

**Semester: 5th** **Date of Performance:15/08/24**

**Subject Name: Design and Analysis of Algorithms** **Subject Code: 22CSH-311**

1. **Aim**: Write a code to sort a given set of elements using the Quick sort.

2. **Objectives:** To implement Quick sort.

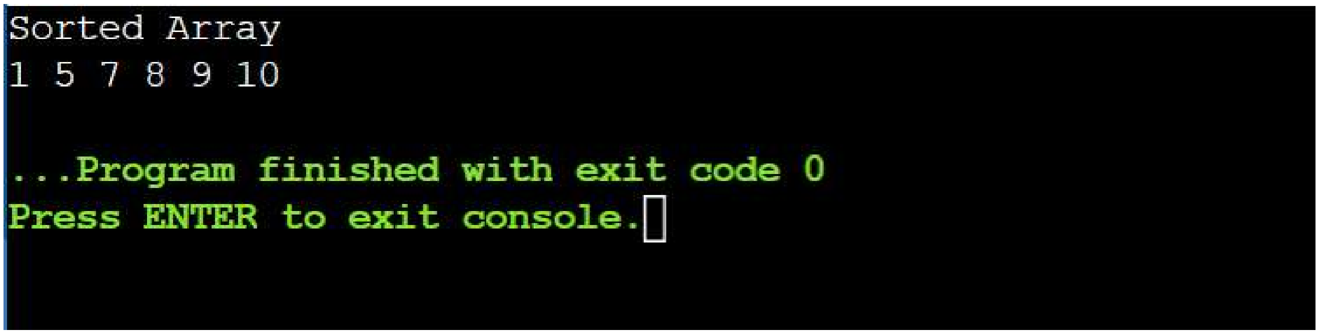
3. **Algorithm:**

- Pick any pivot, let's say the element at the first index value.
- Take two variables to point left and right of the list, excluding pivot.
- The left will point to the lower index, and the right will point to the higher index.
- Swap the pivot element with the first element of the higher values so that the pivot element lands in between the lower and higher values.
- Do the same operations (recursively) for the sub-arrays on the left and right side of the pivot element.

4. **Implementation/Code:**

```
#include <iostream>
#include <vector>
using namespace std;

int partition(vector<int>& arr, int low, int high) {

        int pivot = arr[low];
        int i = low;

        for (int j = low + 1; j <= high; j++) {
                if (arr[j] < pivot) {
                        i++;
                        swap(arr[i], arr[j]);
                }
        }

        swap(arr[low], arr[i]);
        return i;
}

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 = {10, 7, 8, 9, 1, 5};
        int n = arr.size();
        quickSort(arr, 0, n - 1);
        cout << "Sorted Array\n";
        for (int i = 0; i < n; i++) {
                cout << arr[i] << " ";
        }
        return 0;
}
```

**5. Output:**



```
Sorted Array
1 5 7 8 9 10

...Program finished with exit code 0
Press ENTER to exit console.
```

## 6. Time Complexity:

- Best Case : O (N log (N))
- Average Case: O(N log (N))
- Worst Case: O(n^2)

## 7. Learning Outcome:

1) Learnt how to use Divide and Conquer algorithm.
2) Learnt implementing Quick sort.