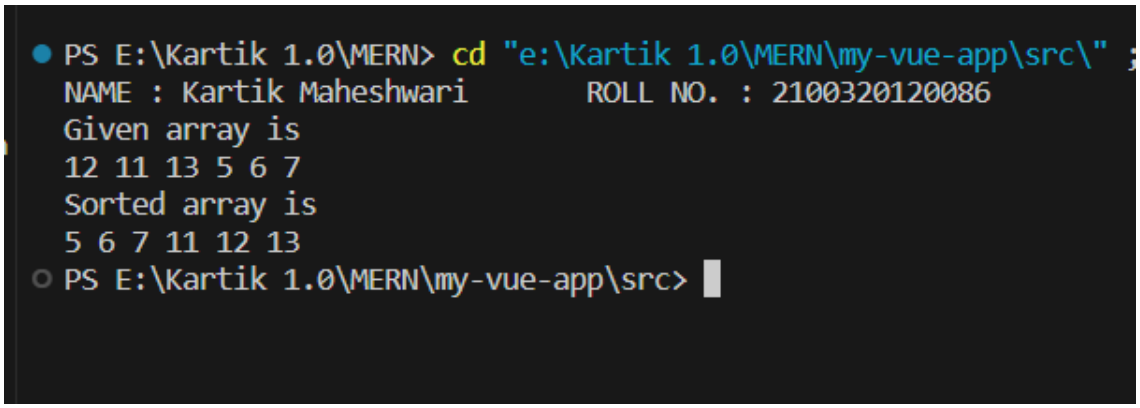
EXPERIMENT – 6

PROGRAM CODE:

```
#include <iostream>
using namespace std;
void merge(int array[], int const left, int const mid, int const right){
    auto const subArrayOne = mid - left + 1;
    auto const subArrayTwo = right - mid;
    auto *leftArray = new int[subArrayOne], *rightArray = new int[subArrayTwo];
    for (auto i = 0; i < subArrayOne; i++)
        leftArray[i] = array[left + i];
    for (auto j = 0; j < subArrayTwo; j++)
        rightArray[j] = array[mid + 1 + j];
    auto indexOfSubArrayOne = 0,
         indexOfSubArrayTwo = 0;
    int indexOfMergedArray = left;
    while (indexOfSubArrayOne < subArrayOne && indexOfSubArrayTwo < subArrayTwo){
        if (leftArray[indexOfSubArrayOne] <= rightArray[indexOfSubArrayTwo]){
            array[indexOfMergedArray] = leftArray[indexOfSubArrayOne];
            indexOfSubArrayOne++;
        }
        else{
            array[indexOfMergedArray] = rightArray[indexOfSubArrayTwo];
            indexOfSubArrayTwo++;
        }
        indexOfMergedArray++;
    }
    while (indexOfSubArrayOne < subArrayOne){
        array[indexOfMergedArray] = leftArray[indexOfSubArrayOne];
        indexOfSubArrayOne++;
        indexOfMergedArray++;
    }
    while (indexOfSubArrayTwo < subArrayTwo){
        array[indexOfMergedArray] = rightArray[indexOfSubArrayTwo];
        indexOfSubArrayTwo++;
        indexOfMergedArray++;
    }
}
void mergeSort(int array[], int const begin, int const end) {
    if (begin >= end)
        return;
    auto mid = begin + (end - begin) / 2;
    mergeSort(array, begin, mid);
    mergeSort(array, mid + 1, end);
    merge(array, begin, mid, end);
```

```
}
void printArray(int A[], int size) {
    for (auto i = 0; i < size; i++)
        cout << A[i] << " ";
    cout << endl;
}
int main() {
    cout << "NAME : Kartik Maheshwari       ROLL NO. : 2100320120086\n";
    int arr[] = {12, 11, 13, 5, 6, 7};
    auto arr_size = sizeof(arr) / sizeof(arr[0]);
    cout << "Given array is " << endl;
    printArray(arr, arr_size);
    mergeSort(arr, 0, arr_size - 1);
    cout << "Sorted array is " << endl;
    printArray(arr, arr_size);
    return 0;
}
```

OUTPUT :



EXPERIMENT –5

PROGRAM :

```
#include <iostream>
using namespace std;
int partition(int arr[], int start, int end) {
    int pivot = arr[start];
    int count = 0;
    for (int i = start + 1; i <= end; i++){
        if (arr[i] <= pivot)
            count++;
    }
    int pivotIndex = start + count;
    swap(arr[pivotIndex], arr[start]);
    int i = start, j = end;
    while (i < pivotIndex && j > pivotIndex){
        while (arr[i] <= pivot){
            i++;
        }
        while (arr[j] > pivot)
        {
            j--;
        }
        if (i < pivotIndex && j > pivotIndex)
        {
            swap(arr[i++], arr[j--]);
        }
    }
    return pivotIndex;
}
void quickSort(int arr[], int start, int end)
{
    if (start >= end)
        return;
    int p = partition(arr, start, end);
    quickSort(arr, start, p - 1);
    quickSort(arr, p + 1, end);
}
int main()
{
    cout << "NAME : Kartik Maheshwari    ROLL no. : 2100320120086n";
    int arr[] = {9, 3, 4, 2, 1, 8};
    int n = 6;
    quickSort(arr, 0, n - 1);
    cout << "SORTED ARRAY\n";
```

```
    for (int i = 0; i < n; i++)
    {
        cout << arr[i] << " ";
    }
    return 0;
}
```

OUTPUT :