```
#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++;
    }

    // Giving pivot element its correct position
    int pivotIndex = start + count;
    swap(arr[pivotIndex], arr[start]);

    // Sorting left and right parts of the pivot element
    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)
{

    // base case
    if (start >= end)
        return;

    // partitioning the array
    int p = partition(arr, start, end);

```

```cpp
    // Sorting the left part
    quickSort(arr, start, p - 1);

    // Sorting the right part
    quickSort(arr, p + 1, end);
}

int main()
{
    int size;
    std::cout << "Enter the size of the array: ";
    std::cin >> size;
    if (size <= 0) {
        std::cerr << "Invalid array size. Please enter a positive integer." << std::endl;
        return 1;
    }

    int array[size];

    std::cout << "Enter the elements of the array:" << std::endl;
    for (int i = 0; i < size; ++i) {
        std::cout << "Element " << i + 1 << ": ";
        std::cin >> array[i];
    }

    quickSort(array, 0, size - 1);

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

    return 0;
}
```