General Sir John Kotelawala Defence University

Department of Computer Science

CS2062 – Data Structures and Algorithm II (Intake 40)

Lab Sheet 04

1. Sort the following array by using quick sort by implementing the following methods. Use the following algorithm for it. Use the rightmost element as the pivot.
   {4,10,2,19,13,15,8,11,14,5}

   Methods -> QuickSort (array A, start, end)
   Partition (array A, start, end)
   PrintArray()

   Quick Sort

```
QUICKSORT (array A, start, end)
{
 if (start < end)
 {
 p = partition(A, start, end)
 QUICKSORT (A, start, p - 1)
 QUICKSORT (A, p + 1, end)
 }
}
```

   Partition Algorithm:

   The partition algorithm rearranges the sub-arrays in a place.

```
PARTITION (array A, start, end)
{
 pivot ? A[end]
 i ? start-1
 for j ? start to end -1 {
 do if (A[j] < pivot) {
 then i ? i + 1
swap A[i] with A[j]
}}
swap A[i+1] with A[end]
return i+1
}
```

2. How does it differ from other sorting algorithms like Bubble Sort or Merge Sort?
3. Compare the space complexity of Merge Sort and Quick Sort. Which sorting algorithm is more space-efficient?

- Create a random array with a size of 100000.
- Create two separate methods for MergeSort() and QuickSort()