**Time Complexity Lab 3**

**Document Id 11**

**Questions**

1. Approximately how many compares will a Quicksort make when sorting an array of N items that are all equal?
2. Quicksort is a
3. A recursive algorithim
4. A Divide and Conqueor algorithim
5. A partitioning algorithim
6. All of a) through c)
7. None of the above

**Quicksort** is a fast sorting algorithm

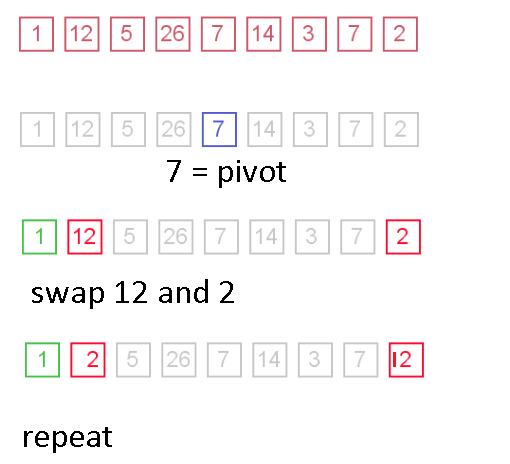
Using the pseudo code below

1) Choose an element, called pivot, from the list. Generally pivot can be the middle index element.

2) Reorder the list so that all elements with values less than the pivot come before the pivot, while all elements with values greater than the pivot come after it (equal values can go either way). After this partitioning, the pivot is in its final position.

This is called the partition operation.

3) Recursively apply the above steps to the sub-list of elements with smaller values and separately the sub-list of elements with greater values



**Practical Task Sorting Algorithm efficiency**

Create an implementation of a quick sort algorithim and compare it either bubble or select sort, show that the time complexity of quick sort is << time complexity of bubble or select sort