**Quick sort**

Pseudo code

Make pivot is element the end of the list and compare pivot following rule the left of the pivot should be greater than and the right of the pivot should be less than.

Create 2 pointer the move left and right to compare following rules. With left start from first element of the list increase for every loop and right pointer start before first pivot and decrease index for every loop.

While value left pointer greater than pivot and then increase index of left pointer. And value right pointer less than pivot decrease index right pointer.

Else value left pointer less than pivot STOP at left index and right pointer greater than pivot STOP. And then it swap value left pointer and value right pointer.

If index left pointer and index right pointer equal and then it will swap left pointer with pivot. Finish step 1 we have left element of the pivot great than and right elements of the pivot less than. (1)

Loop the same (1) for left pointer loop from first element and right pointer will loop from before index of first pivot.

And check element the same (1) for elements of left hand side of the first pivot.

And check element the same(1) for elements of the right hand side of the pivot.

if(low >= high){

return;

}

T pivot=data[high];

int left=low;

int right=high;

while(left<right){

while(data[left].compareTo(pivot)>=0 && left < right){

left++;

}

while(data[right].compareTo(pivot)<=0 && left < right){

right--;

}

swap(left, right);//swap in case left > pivot and right < pivot

}

swap(left, high);//left right equal swap with pivot

QuickSortDescByHeight(low, left - 1);//left pointer is pivol so - 1 move before

QuickSortDescByHeight(left + 1, high);//move from left + 1. Before the pivot

Complexity Analysis:O(logn)