PaperAssignment4

Mikołaj Gawryś

March 2023

1 Assignment one

An array sorted in ascending order is in fact a min-heap (assuming we store the heap according to the typical convention), since it satisfies the property:

$$key(n) >= key(parent(n)) \tag{1}$$

for all the nodes apart from the root.

2 Assignment two

If the initial sequence is sorted and the pivot point is always chosen as the middle-point, then every recursive call splits the sequence such that it creates two subsequences of the same length (the items on the left side of the pivot go into one list, on the right to the other). Ergo the height of the tree becomes $O(log\, n)$, since we can split a given sequence like this $O(log\, n)$ times. At each level, it takes $O(n)$ time to come up with a partition (we still need to do comparison with the pivot point). In the end, the time complexity becomes $O(n\, log\, n)$, which happens to be the lower bound for comparison-based sorting algorithms.

3 Assignment three

This line helps identify whether the `compareTo()` method has been called on a String that is also passed in as an argument for the method - and those objects have the same memory address (i.e `s.compareTo(s)`. The

`this` keyword refers to the object we are calling the method on, and if the memory address of this object is the same as the address of the String, we return 0, since further comparisons and computations are redundant.

4 Assignment four

Sequence of values returned:

$$[R, R, P, O, T, Y, I, I, U, Q, E, U] \tag{2}$$