

# CS 214 – Algorithms and Complexity

Shanghai Jiaotong University, Fall 2016

- Handed out: Friday, 2016-10-14
- Due: Wednesday, 2016-10-19
- Feedback: Friday, 2016-10-21
- Revision due: Monday, 2016-10-24

## 2 Sorting Algorithms

### 2.1 largest, smallest, and second largest

**Exercise 1.** Let  $A$  be an array of size  $n$ , where  $n$  is even. Describe how to find both the minimum and the maximum with at most  $\frac{3}{2}n - 2$  comparisons. Make sure your solution is *simple*, in describe it in a clear and succinct way!

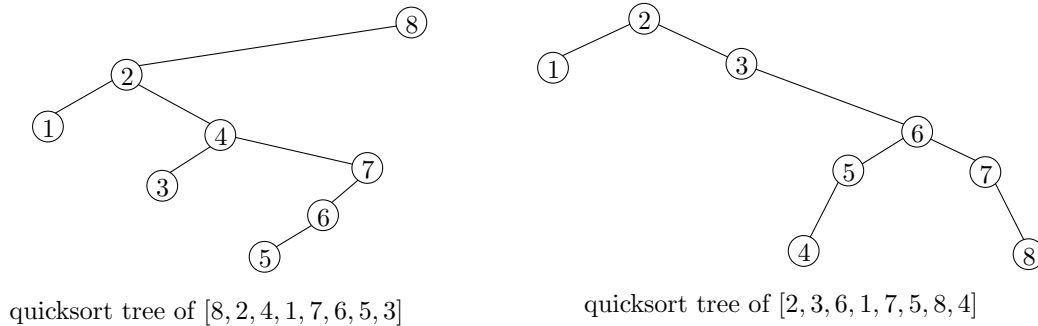
**Exercise 2.** Given an array  $A$  of size  $n = 2^k$ , find the second largest element with at most  $n + \log_2(n)$  comparisons. Again, your solution should be *simple*, and you should explain it in a clear and succinct way!

**Exercise 3.** [max needs at least  $n - 1$  comparisons] Suppose I propose an algorithm for finding the maximum of  $A$  using  $n - 2$  comparisons. Show that I am wrong, i.e., my algorithm must fail on certain inputs  $A$ .

**Remark.** The only way my algorithm can access the elements is to take two elements and *compare* them pairwise. That is, the algorithm is not allowed to check whether  $A[i] > n/2$  or something similar. Think of the elements of  $A$  as abstract objects, not necessarily numbers!

**Exercise 4.** In the lecture we proved that for  $n = 2^d$ , the number of comparisons performed by **mergesort** is at most  $n \log(n) - n + 1$ . Find a method that constructs, for every  $n = 2^d$ , an array  $A$  of  $n$  integers on which **mergesort** reaches this bound, i.e., performs exactly  $n \log(n) - n + 1$  when sorting  $A$ . For  $n = 8$  we have seen  $[0, 4, 2, 6, 1, 5, 3, 7]$  in class.

Recall the quicksort tree defined in the lecture.



We denote a specific list (ordering) by  $\pi$  and the tree by  $T(\pi)$ .  $A_{i,j}$  is an indicator variable which is 1 if  $i$  is an ancestor of  $j$  in the tree  $T(\pi)$ , and 0 otherwise. In the lecture, we have derived:

$$\mathbb{E}[A_{i,j}] = \frac{1}{|i - j| + 1}$$

$$\text{total number of comparisons} = \sum_{i \neq j} A_{i,j}.$$

From this we have deduced that the expected number of comparisons of quicksort on a random input is at most  $2nH_n$ , where  $H_n = \sum_{i=1}^n \frac{1}{i}$  is called the  $n^{\text{th}}$  harmonic number.

**Exercise 5.** Derive a precise formula for the expected number of comparisons made by quicksort. Your formula should be in “closed form”, i.e., not contain any  $\mathbb{E}$ ,  $\prod$  or  $\sum$ , but may contain  $H_n$ .

## 2.2 Quickselect

Remember the recursive algorithm **QUICKSELECT** from the lecture. I write it here in pseudocode. In analogy to quicksort we define **QuickSelect** deterministically and assume that the input array is random, or has been randomly

---

**Algorithm 1** Select the  $k^{\text{th}}$  smallest element from a list  $A$ 

---

```
1: procedure QUICKSELECT( $A, k$ )
2:   if  $|A| = 1$  then
3:     return  $A[1]$ 
4:   else:
5:      $p := A[1]$ 
6:      $B := [x \in A \mid x < p]$ 
7:      $C := [x \in A \mid x > p]$ 
8:     if  $|B| = k - 1$  then
9:       return  $p$ 
10:    else if  $|B| \geq k$  then
11:      return QUICKSELECT( $B, k$ )
12:    else
13:      Return QUICKSELECT( $C, k - |B| - 1$ )
14:    end if
15:  end if
16: end procedure
```

---

shuffled before QuickSelect is called. We assume that  $A$  consists of distinct elements and  $1 \leq k \leq |A|$ .

Let  $C$  be the number of comparison made by QUICKSELECT. In the lecture we proved that  $\mathbb{E}[C] \leq O(n)$  when we run it on a random input.

**Exercise 6.** Explain how QUICKSELECT can be viewed as a “partial execution” of quicksort with the random pivot selection rule. Draw an example quicksort tree and show which part of this tree is visited by QuickSelect.

Let  $B_{i,j,k}$  be an indicator variable which is 1 if  $i$  is a common ancestor of  $j$  and  $k$  in the quicksort tree. That is, if both  $j$  and  $k$  appear in the subtree of  $T(\pi)$  rooted at  $i$ .

**Exercise 7.** What is  $\mathbb{E}[B_{i,j,k}]$ ? Give a succinct formula for this.

**Exercise 8.** Let  $C(\pi)$  be the number of comparisons made by QUICKSELECT when given  $\pi$  as input. Design a formula for  $C(\pi)$  with the help of the indicator variables  $A_{i,j}$  and  $B_{i,j,k}$  (analogous to the formula  $\sum_{i \neq j} A_{i,j}$  for the number of comparisons made by quicksort).

**Exercise 9.** Derive a precise, closed formula for the expected number of comparisons made by QUICKSELECT when run on a random input. Again,

“closed formula” means no  $\mathbb{E}, \prod, \sum$ . The harmonic number  $H_n$ , however, is allowed.