CS/ECE 473 Fall 2020
Homework 3 Problem 3

Ryan Prendergast (ryanp4)
Noah Watson (nwatson3)
Lawson Probasco (lawsonp2)

3.a

Write down a description of randomized quick selection in pseudocode. Show that the expected depth of the recursion of randomized quick selection is $O(logn)$, and that the expected running time is $O(n)$.

Solution: We provide the following pseudocode for randomized quickselect on an array A of size n. It takes the arguments l,r as indices into A: $l \leq n$, $r \leq n$ and $r >= l$. We are looking to find the kth largest element in A. We name the function RQS, and use the standard partition subroutine which, given a left and right subarray, as well as a pivot, partitions elements into the appropriate subarray s.t. elements in the left subarray are less than the pivot, and elements in the right subarray are greater than the pivot.

Algorithm 1 RQS(l, r, k)

```
if l == r then
    return
end if
pivot ← RandInt(l..r)
partition(l, r, pivot)
if pivot == k then
    return A[k]
else
    return RQS(pivot + 1, r, k)
end if
```

We first define the recurrence $\bar{T}(n)$ to be the expected value of the amount of work on a problem of size n. We know that the partition function requires $n-1$ comparisons to solve a problem of size n.

$$\bar{T}(n) = n - 1 + max[\bar{T}(k-1), \bar{T}(n-k)]$$

We define a good pivot to be in the middle 50% of elements and a bad one to be in the lower 25 or upper 25. In the event of a good pivot,

$$\bar{T}(n) \leq n - 1 + \bar{T}(3/4 * n)$$

and in the event of a bad pivot

$$\bar{T}(n) \leq n - 1 + \bar{T}(n)$$

Because the pivot is selected uniformly randomly, there is equal probability of 1/2 of both. So,

$$\bar{T}(n) \leq 2n - 2 + .5 * \bar{T}(n) + .5 * \bar{T}(3/4 * n)$$
$$\bar{T}(n) \leq 4n - 4 + \bar{T}(3/4 * n)$$

By the master theorem, we know that the right side recurrence runs in $O(n)$ time, and this serves as an upper bound.

Next we define the recurrence $\bar{D}(n)$ to be the expected value of the depth of the recursion tree of a problem of size n. We know that each call to the function adds one to the recursion tree depth.

$$\bar{D}(n) = 1 + max[\bar{D}(k-1), \bar{D}(n-k)]$$

We will use the same definition of a good and bad pivot as previously: define a good pivot to be in the middle 50% of elements and a bad one to be in the lower 25 or upper 25. In the event of a good pivot,

$$\bar{D}(n) \leq 1 + \bar{D}(3/4 * n)$$

and in the event of a bad pivot

$$\bar{D}(n) \leq 1 + \bar{D}(n)$$

Because the pivot is selected uniformly randomly, there is equal probability of $1/2$ of both. So,

$$\bar{D}(n) \leq 2 + .5 * \bar{D}(n) + .5 * \bar{D}(3/4 * n)$$
$$\bar{D}(n) \leq 4 + \bar{D}(3/4 * n)$$

By the master theorem, we know that the right side recurrence runs in $O(logn)$ time, and this serves as an upper bound.

■