CIT 596
Quicksort

QUICKSORT

Algorithm Idea:

1. Choose a "pivot" element x at random.
2. Compare all elements to the pivot.
3. Partition all elements into two sets:
   - S (elements smaller than x)
   - L (elements larger than x)
4. Arrange the elements so that all elements in S come before x and all elements in L come after x.
5. Recursively sort S and L. Let $|S| = k$.

QUICKSORT – RUNTIME

If we always get a bad partition (i.e. in the worst-case),

- The partition does not split array at all.
- At every step, $k = 1$ or $k = n - 1$
- Then $T(n) = T(1) + T(n-1) + n = O(n^2)$, similar to insertion sort.

If we always get a good partition:

- The partition splits array evenly at every step ($k = n/2$)
- Then $T(n) = T\left(\frac{n}{2}\right) + T\left(\frac{n}{2}\right) + O(n) = O(n\, log\, n)$, similar to merge sort.

Instead of analyzing either extreme, we analyze the expected time.

ANALYZING A RANDOMIZED ALGORITHM

- Remember that the algorithm's behavior is random.
- For each input, the number of steps it takes is a random variable.
- Our goal: bound the expectation of this random variable.
- This is the expected worst-case behavior of the algorithm.
- What is the philosophy? The input is not in the algorithm's control, but we expect to not be too unlucky with our coin tosses.

QUICKSORT – RUNTIME IN EXPECTATION

Key observations:

- Any two elements are never compared more than once.
- If p is the pivot, $x < p$, and $y > p$, then x and y are never compared. They "go separate ways" in the recursion.

QUICKSORT – RUNTIME IN EXPECTATION

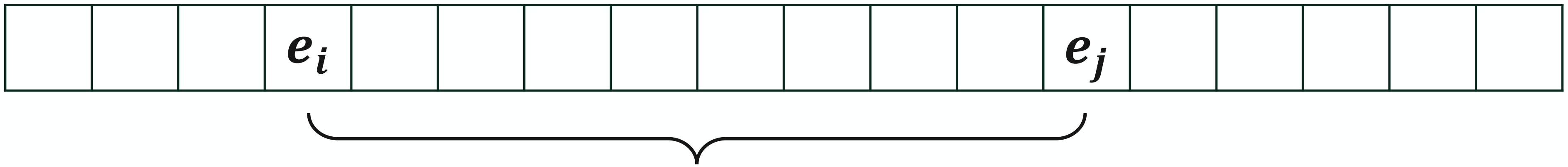
Analyze with random variables:

- Denote the k^{th} smallest element in the array as e_k
- $X =$ total number of comparisons
- $X_{ij} =$ indicator random variable for the event "e_i and e_j are compared"
- Then $X = \sum_{1 \le i < j \le n} X_{ij}$
- Linearity of expectation:

$$\mathrm{E}[X] = \sum_{1 \le i < j \le n} \mathrm{E}[X_{ij}] = \sum_{i=1}^{n} \sum_{j=i+1}^{n} \Pr[X_{ij} = 1]$$

To finish, we need to find $\Pr[X_{ij} = 1]$

QUICKSORT – $Pr[X_{ij} = 1]$



$j - i + 1$

- e_i and e_j will "go separate ways" when a pivot $p \in [i, j]$ is chosen for the first time.
- If p is strictly between i and j, then $X_{ij} = 0$.
- If $p = e_i$ or $p = e_j$, then $X_{ij} = 1$.
- Since pivots are chosen uniformly at random, $\Pr[X_{ij} = 1] = 2 \cdot \frac{1}{j-i+1}$

QUICKSORT – $Pr[X_{ij} = 1]$

(A bit more detail for the previous slide)

- Which pivots must be chosen for e_i and e_j to be compared?
  - Either e_i or e_j (2 total)
- Which pivots must be chosen for e_i and e_j to not (ever) be compared?
  - $e_{i+1}, e_{i+2}, \dots, e_{j-1}$ ($j - i - 1$ total)
- Elements are chosen as pivots randomly, so…
- $\mathrm{E}[X_{ij}] = 2 \cdot \frac{1}{(j-i-1)+2} = 2 \cdot \frac{1}{j-i+1}$

QUICKSORT – RUNTIME

$$\mathrm{E}[X] = \sum_{1 \le i < j \le n} \mathrm{E}[X_{ij}] = \sum_{1 \le i < j \le n} \Pr[X_{ij} = 1] = \sum_{i=1}^{n} \sum_{j=i+1}^{n} \frac{2}{j - i + 1}$$

- How to compute the sum? Group terms by value of $j - i$.

- How many terms with $j - i = d$? Exactly $n - d$.

- Each such term contributes $\frac{2}{d+1}$.
- So total is

$$\sum_{d=1}^{n-1} \frac{2(n-d)}{d+1} = 2n \sum_{d=1}^{n-1} \frac{1}{d+1} - \sum_{d=1}^{n-1} \frac{2d}{d+1}$$

QUICKSORT – RUNTIME

$$\sum_{d=1}^{n-1} \frac{2(n-d)}{d+1} = 2n \sum_{d=1}^{n-1} \frac{1}{d+1} - \sum_{d=1}^{n-1} \frac{2d}{d+1}$$

- Observe that the first sum is a Harmonic Series, which is $O(\log n)$.

- Every term in the second sum is at least 1, so the second sum is at least n.

- Thus,

$$\mathrm{E}[X] = O(n \log n)$$