Data Structures and Algorithms
CSE 465

LECTURE 9
• Finishing QuickSort
• Randomized Selection

Sofya Raskhodnikova and Adam Smith
Reminder: QuickSort

Quicksort an \( n \)-element array:

1. **Divide:** Partition the array around a pivot \( x \) such that elements on left are \( \leq x \) and elements on right are \( \geq x \)

2. **Conquer:** Recursively sort the two subarrays.

3. **Combine:** Nothing!

**Key:** Linear-time partitioning subroutine.

Last lecture: use random element \( A[p] \) as pivot
Review

Recall: A split is “OK” when both pieces have \( \geq \lfloor n/4 \rfloor \) elements

– What is the probability of an OK split with a random pivot?  
  \( \text{(Answer} \approx \frac{1}{2}) \)

– Suppose we keep trying pivots and partitioning until we find an OK split.

  • What is the expected number of pivots we try before finding a good one?  
    \( \text{(Answer} \approx 2) \)

  • What is the expected running time it takes to find a good pivot this way?  
    \( \text{(Answer: } \Theta(n)) \)
Randomized quicksort analysis

Let $T(n)$ = the random variable for the running time of randomized quicksort on an input of size $n$, assuming random numbers are independent.

For $k = 0, 1, \ldots, n-1$, define the indicator random variable

$$X_k = \begin{cases} 
1 & \text{if PARTITION generates a } k:n-k-1 \text{ split,} \\
0 & \text{otherwise.}
\end{cases}$$

$E[X_k] = \Pr\{X_k = 1\} = 1/n$, since all splits are equally likely, assuming elements are distinct.
Analysis (continued)

\[ T(n) = \begin{cases} 
T(0) + T(n-1) + \Theta(n) & \text{if } 0 : n-1 \text{ split,} \\
T(1) + T(n-2) + \Theta(n) & \text{if } 1 : n-2 \text{ split,} \\
\vdots \\
T(n-1) + T(0) + \Theta(n) & \text{if } n-1 : 0 \text{ split,} 
\end{cases} \]

\[ = \sum_{k=0}^{n-1} X_k (T(k) + T(n - k - 1) + \Theta(n)) \]
Calculating expectation

\[ E[T(n)] = E \left[ \sum_{k=0}^{n-1} X_k (T(k) + T(n-k-1) + \Theta(n)) \right] \]

Take expectations of both sides.
Calculating expectation

\[ E[T(n)] = E \left[ \sum_{k=0}^{n-1} X_k (T(k) + T(n - k - 1) + \Theta(n)) \right] \]

\[ = \sum_{k=0}^{n-1} E[X_k (T(k) + T(n - k - 1) + \Theta(n))] \]

Linearity of expectation.
Calculating expectation

\[ E[T(n)] = E\left[ \sum_{k=0}^{n-1} X_k (T(k) + T(n - k - 1) + \Theta(n)) \right] \]

\[ = \sum_{k=0}^{n-1} E[X_k (T(k) + T(n - k - 1) + \Theta(n))] \]

\[ = \sum_{k=0}^{n-1} E[X_k] \cdot E[T(k) + T(n - k - 1) + \Theta(n)] \]

Independence of \( X_k \) from other random choices.
Calculating expectation

\[ E[T(n)] = E\left[ \sum_{k=0}^{n-1} X_k (T(k) + T(n-k-1) + \Theta(n)) \right] \]

\[ = \sum_{k=0}^{n-1} E[X_k (T(k) + T(n-k-1) + \Theta(n))] \]

\[ = \sum_{k=0}^{n-1} E[X_k] \cdot E[T(k) + T(n-k-1) + \Theta(n)] \]

\[ = \frac{1}{n} \sum_{k=0}^{n-1} E[T(k)] + \frac{1}{n} \sum_{k=0}^{n-1} E[T(n-k-1)] + \frac{1}{n} \sum_{k=0}^{n-1} \Theta(n) \]

Linearity of expectation; \( E[X_k] = 1/n \).
Calculating expectation

\[
E[T(n)] = E\left[ \sum_{k=0}^{n-1} X_k (T(k) + T(n-k-1) + \Theta(n)) \right]
\]

\[
= \sum_{k=0}^{n-1} E[X_k (T(k) + T(n-k-1) + \Theta(n))]
\]

\[
= \sum_{k=0}^{n-1} E[X_k] \cdot E[T(k) + T(n-k-1) + \Theta(n)]
\]

\[
= \frac{1}{n} \sum_{k=0}^{n-1} E[T(k)] + \frac{1}{n} \sum_{k=0}^{n-1} E[T(n-k-1)] + \frac{1}{n} \sum_{k=0}^{n-1} \Theta(n)
\]

\[
= \frac{2}{n} \sum_{k=1}^{n-1} E[T(k)] + \Theta(n)
\]

Summations have identical terms.
A recurrence on expected value

• We want to know the asymptotic behaviour of

\[ F(n) = E[T(n)] \]

• We know it satisfies

\[ F(n) = \frac{2}{n} \sum_{k=2}^{n-1} F(k) + \Theta(n) \]
Hairy recurrence

\[ F(n) = \frac{2}{n} \sum_{k=2}^{n-1} F(k) + \Theta(n) \]

(The \( k = 0, 1 \) terms can be absorbed in the \( \Theta(n) \).)

**Prove:** \( F(n) \leq a n \lg n \) for constant \( a > 0 \).

- Choose \( a \) large enough so that \( a n \lg n \) dominates \( E[T(n)] \) for sufficiently small \( n \geq 2 \).

**Use fact:** \( \sum_{k=2}^{n-1} k \lg k \leq \frac{1}{2} n^2 \lg n - \frac{1}{8} n^2 \) (exercise).
Substitution method

\[ F(n) \leq \frac{2}{n} \sum_{k=2}^{n-1} ak \lg k + \Theta(n) \]

Substitute inductive hypothesis.
Substitution method

\[ F(n) \leq \frac{2}{n} \sum_{k=2}^{n-1} a_k \lg k + \Theta(n) \]

\leq \frac{2a}{n} \left( \frac{1}{2} n^2 \lg n - \frac{1}{8} n^2 \right) + \Theta(n)

Use fact.
Substitution method

\[ F(n) \leq \frac{2}{n} \sum_{k=2}^{n-1} ak \log k + \Theta(n) \]

\[ \leq \frac{2a}{n} \left( \frac{1}{2} n^2 \log n - \frac{1}{8} n^2 \right) + \Theta(n) \]

\[ = an \log n - \left( \frac{an}{4} - \Theta(n) \right) \]

Express as \textit{desired} – \textit{residual}.
Substitution method

\[ F(n) \leq \frac{2}{n} \sum_{k=2}^{n-1} ak \log k + \Theta(n) \]

\[ = \frac{2a}{n} \left( \frac{1}{2} n^2 \log n - \frac{1}{8} n^2 \right) + \Theta(n) \]

\[ = an \log n - \left( \frac{an}{4} - \Theta(n) \right) \]

\[ \leq an \log n \]

if \( a \) is chosen large enough so that \( an/4 \) dominates the \( \Theta(n) \).
Quicksort in practice

• Quicksort is a great general-purpose sorting algorithm.

• Quicksort is typically over twice as fast as merge sort.

• Quicksort can benefit substantially from code tuning.

• Quicksort behaves well even with caching and virtual memory.
Order statistics

Select the $i$th smallest of $n$ elements (the element with rank $i$).

- $i = 1$: minimum;
- $i = n$: maximum;
- $i = \lfloor (n+1)/2 \rfloor$ or $\lceil (n+1)/2 \rceil$: median.

**Naive algorithm**: Sort and index $i$th element.

Worst-case running time $= \Theta(n \lg n) + \Theta(1)$

$= \Theta(n \lg n)$,

using merge sort (not quicksort...).
Background: Properties of the Expectation

Here are some useful properties of the expectation:

• For any random variables $A$, $B$, and constants $c,d$:
  \[ E[cA+dB] = c \ E[A] + d \ E[B] \]

• For any two independent random variables
  \[ E[AB] = E[A] \ E[B] \]

• If $A$ only takes the values 0 and 1 then
  \[ E[A] = \text{Prob}[ A=1 ] \]
Background: Geometric Random Variables

• Suppose we flip a coin many times independently, and each time the probability that it comes up “heads” is $p$
  – If $X$ is the number of times we flip the coin before getting heads, then $X$ is a geometric random variable with parameter $p$

• E.g.
  – “How many times do we roll a die until we see a 1”? (Geometric with $p = 1/6$.)
  – “How many times do we flip a fair (50:50) coin before getting heads?” (Geometric with $p = 1/2$.)
  – “How many pivots do we choose before seeing an OK split?” (Geometric with $p = 1/2$.)

• For geometric r.v. with parameter $p$, $E(X) = 1/p$