CSCE 750
9/26/2023

① Simulating a fair coinflip with a coin of unknown bias
{say, $Pr[H] = p$, for some unknown $0 < p < 1$}

How do you do it, and how many times do you <u>expect</u> to flip the coin?

~~Se~~ Selection Problem:

Given an array $A[1..n]$ of distinct comparable elements and a number k, $1 \le k \le n$. Return is the k'th smallest element in $A[1..n]$:

unique element that has exactly $k-1$ smaller elements

$k = 1$: min element

$k = n$: max element

$k = \lfloor \frac{n}{2} \rfloor$: median

One solution:

$\Theta(n \lg n)$ — 1) Sort A into increasing order

$O(1)$ — 2) Return $A[k]$

Better solution? Yes

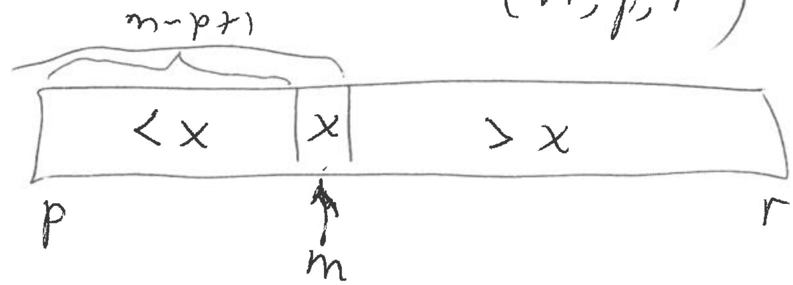
1) Randomized select in $\Theta(n)$ worst-case expected time
2) Deterministic select in $\Theta(n)$ worst-case time

RandomizedSelect(A, p, r, k)

// precondition: $1 \le k \le r-p+1$

// returns k^{th} smallest element of $A[p..r]$

$\Theta(n)$ [$m := \text{RandomizedPartition}(A, p, r)$



if $k == m-p+1$:

Return $A[m]$

if $k < m-p+1$:

Return RandomizedSelect$(A, p, m-1, k)$

else // $k > m-p+1$

Return RandomizedSelect$(A, m+1, r, k-m+p-1)$

["k^{th} order statistic" = k^{th} smallest]

Analysis: Worst-case Expected time to do selection on n elements: $E(n)$:

$$E(n) := \Theta(n) + \frac{1}{n} \sum_{\underbrace{q}_{\text{size of left list}}=1}^{n-1} \max\{E(q), E(n-q-1)\}$$

$$\cancel{= \Theta(n) + \frac{1}{n}\sum_{q=1}^{n-1} \max\{}$$

$$= \Theta(n) + \frac{2}{n}\sum_{q=\lfloor \frac{n}{2} \rfloor}^{n-1} E(q)$$

$\therefore$ $E(n) = \Theta(n)$ (subst. method)

Deterministic selection in worst-case time $\Theta(n)$

Lemma: Fix $\alpha, \beta \geq 0$ such that $\alpha + \beta < 1$.
Let $T(n)$ satisfy

$$T(n) = T(\alpha n) + T(\beta n) + n$$

Then $T(n) = \Theta(n)$.

Proof by subst. method: Assume $T(m) \leq cm$ $\forall m < n$.

$$\begin{aligned}
T(n) &= T(\alpha n) + T(\beta n) + n \\
&\leq c\alpha n + c\beta n + n \\
&= cn(\alpha + \beta) + n \\
&= n\underline{(1 + c(\alpha + \beta))} \\
&\leq cn
\end{aligned}$$

provided $1 + c(\alpha + \beta) \le c$

$\Leftrightarrow \quad 1 \le c(1 - (\alpha + \beta))$

$\Leftrightarrow \quad c \ge \dfrac{1}{1 - (\alpha + \beta)}$

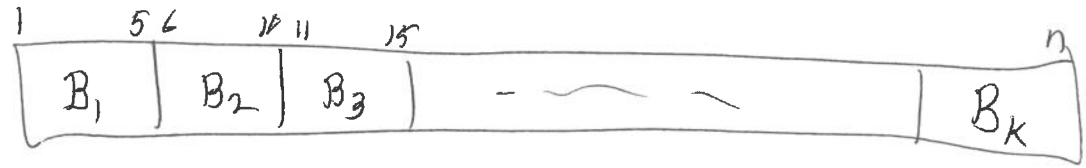
$\therefore \; T(n) = O(n)$

Obviously $T(n) = \Omega(n)$ ∎

DetSelect(A, p, r, k) // $n := r - p + 1$

1) Chop A into about $\frac{n}{5}$ many blocks of 5 elements each;

A



$k := \frac{n}{5}$

for $j := 1$ to k, let
$O(1)$ → m_j be the median of B_j — $\Theta(k)$

let $B[1..k]$ be the array of medians: $B[j] = m_j$ — $\Theta(k)$

$\Theta(n)$ time

2) ~~Det~~ Call DetSelect recursively on the array B ~~with~~ to find its median ($\frac{k}{2}$ order statistic) m — $T\left(\frac{n}{5}\right)$

[$T(n)$ = worst-case time of DetSelect on n elements]

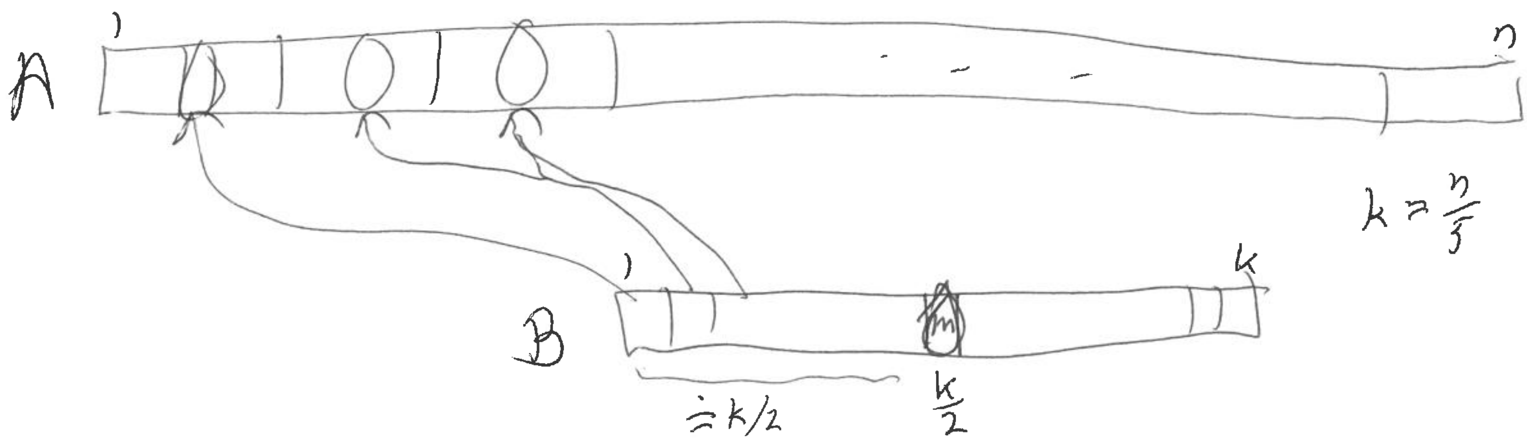
$\Theta(n)$

3) Partition $A[p..r]$ using pivot m.
let $A[l] = m$ for l
pivot is in $A[l]$ after the partition
$p \le l \le r$

$T(s)$

s = max size of a sublist

4) if $k == l-p+1$:
    Return $A[l]$
if $k < l-p+1$
    Return DetSelect$(A, p, l-1, k)$
else
    Return DetSelect$(A, l+1, r, k-l+p-1)$



- Elements of B that are $< m$: $\frac{k}{2}$
- For each of these there are 2 elements from its block less than it : $\ge \frac{3k}{2} = \frac{3n}{10}$

$\therefore$ There are $\le n - \frac{3n}{10} = \frac{7n}{10}$ elements $> m$

Symmetrically, there can be $\le \frac{7n}{10}$ elements $< m$

$\therefore$ Size of either sublist is $\le \frac{7n}{10} + 5$

So

$$T(n) = \underline{\Theta(n)} + T\left(\frac{n}{5}\right) + T\left(\frac{7n}{10}\right)$$

$$\alpha = \frac{1}{5} \qquad \beta = \frac{7}{10} \qquad \alpha + \beta = \frac{9}{10} < 1$$

$\therefore$ $T(n) = \Theta(n)$ by the lemma. ∎

Same idea: Can get a deterministic version of QuickSort running in worst-case time $\Theta(n \lg n)$

Key: Use $\overset{\Theta(n)}{\text{DetSelect}}$ to find the median, use that as the pivot.
