Lecture 1
Median Finding

Overview

- Divide & Conquer Overview
- Master Theorem
- Median Finding Trivial Solution
- Median Finding (Blum, Floyd, Pratt, Rivest and Tarjan)

Divide & Conquer

Given a problem of size n divide it into subproblems of size $\frac{n}{b}$, $a \geq 1$, $b > 1$. Solve each subproblem recursively. Combine solutions of subproblems to get overall solution.

$$T(n) = aT(\frac{n}{b}) + [\text{work for merge}]$$

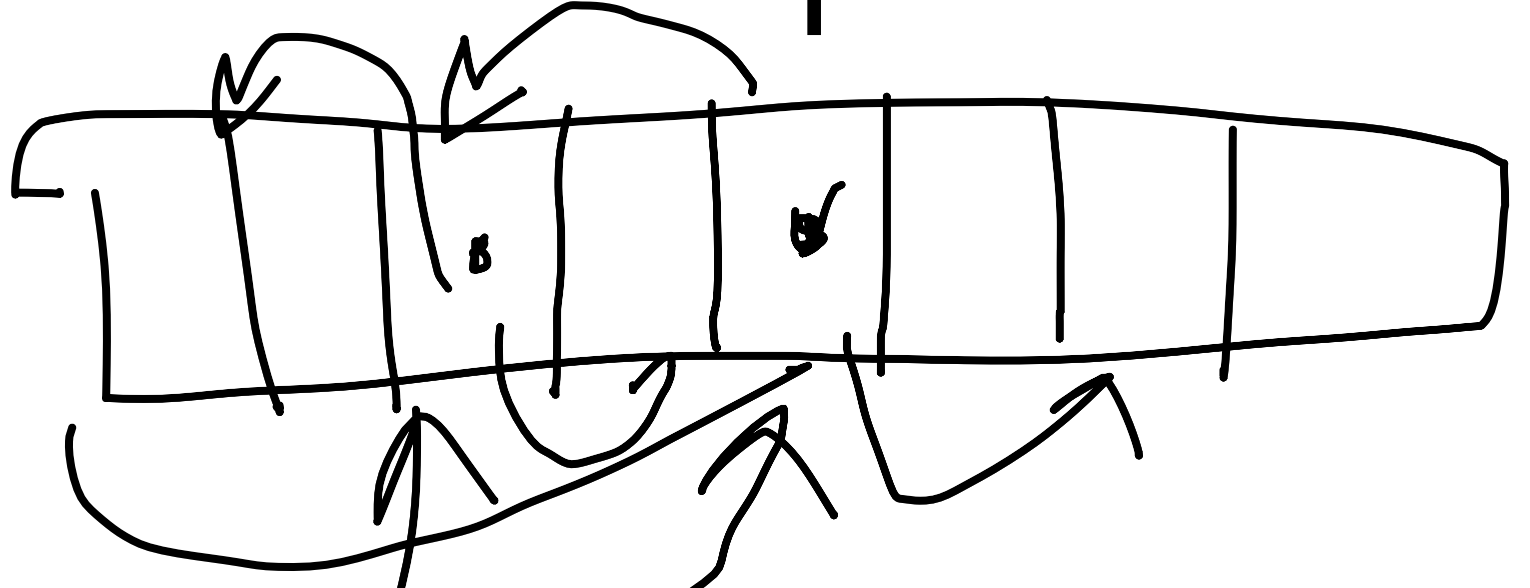
Master Theorem Definition

$$T(n) = \begin{cases} aT\left(\frac{n}{b}\right) + O(n^c), & n > 1 \\ O(1), & n = 1 \end{cases}$$

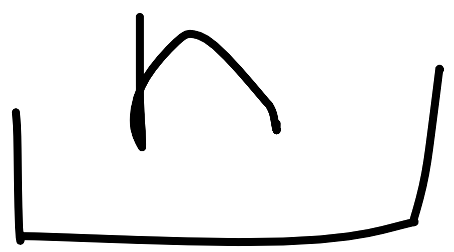
$a \in \mathbb{N}, \; b \in \mathbb{R}, \; c \in \mathbb{R}$

$$\begin{cases} 1) \; c > \log_b a, & T(n) = O(n^c) \\ 2) \; c = \log_b a, & T(n) = O(n^c \cdot \log n) \\ 3) \; c < \log_b a, & T(n) = O(n^{\log_b a}) \end{cases}$$

Master Theorem Examples



$$T(n) = aT\left(\frac{n}{b}\right) + O(n^c)$$



$$T(n) = 1 \cdot T\left(\frac{n}{2}\right) + O(1) \qquad n^c = 1$$

$c = ?$

$c = 0$

$a = 1$

$b = 2$

$c = 0$

$c \;?\; \log_b a$

$0 = \log_2 1 = 0$

$$\boxed{T(n) = O(n^0 \cdot \log n) = O(\log n)}$$

Median Finding Problem Definition

Problem: Given an array $A = [1, \ldots, n]$ and index i $(1 \leq i \leq n)$. Find $\underbrace{i\text{-th smallest element}}$ in A.

$i = 1$ - min

$i = n$ - max

$i = [n+1]/2$

→ order statistic

$1, n, \frac{n+1}{2}$ - median

$\lfloor x \leq y \rfloor$, $\left[\frac{x+y}{2}\right]$

Median Finding Trivial Solution

$A = [1, \ldots, n]$

Sort $O(n \cdot \log n) \rightarrow$

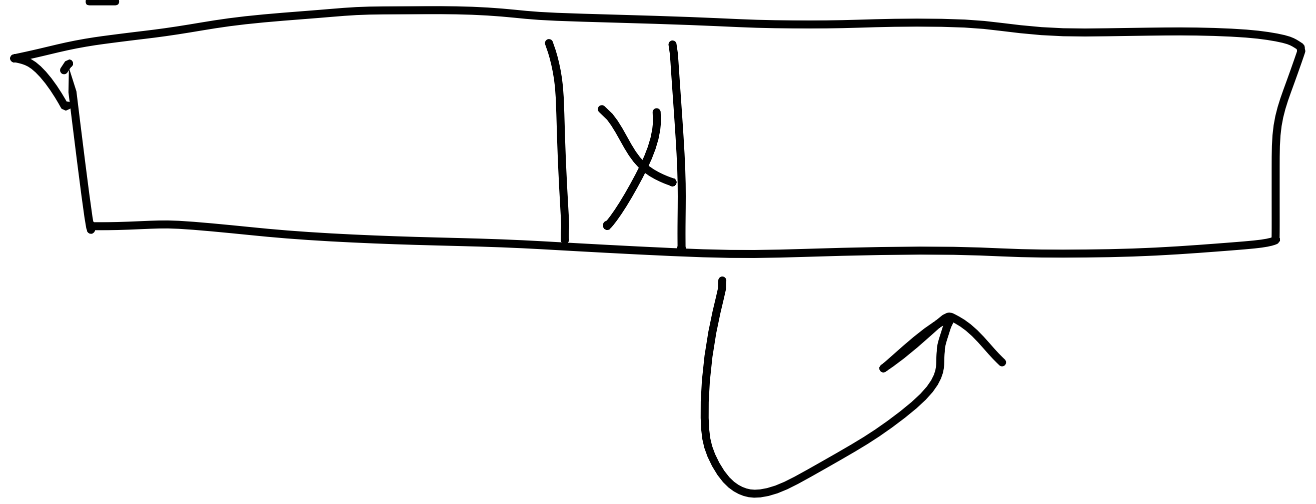
$+$ Find Median $O(1)$

$O(n \cdot \log n)$

Median Finding O(n) in average case by Toni Hoar

S – an array
i – the

```
Select(S, i)
  1.  Pick x ∈ S // choose a pivot element. We can do it even
randomly
  2.  B = {y ∈ S|y<x} // Set of values less than x
  3.  C = {y ∈ S|y>x} // Set of values greater than x
  4.  D = {y ∈ S|y==x} // Set of values equal to x
  5.  Compute k = rank(x) // We know how many elements less than x,so
we know the rank!
  6.  if k = i:
  7.    return x
  8.  else if k>i:
  9.    return Select(B, i)
  10. else if k<i:
  11.   return Select(C, i - k)
```



Median Finding O(n) in average case by Toni Hoar

```
Select(S, i)
  1.  Pick x ∈ S // choose a pivot element. We can do it even randomly
  2.  B = {y ∈ S|y<x} // Set of values less than x
  3.  C = {y ∈ S|y>x} // Set of values greater than x
  4.  D = {y ∈ S|y==x} // Set of values equal to x
  5.  Compute k = rank(x) // We know how many elements less than x,so we know the rank!
  6.  if k = i:
  7.    return x
  8.  else if k>i:
  9.    return Select(B, i)
  10. else if k<i:
  11.   return Select(C, i - k)
```

Median Finding O(n) in average case
Example of work

$\boxed{i = 6}$

$l = [9, 1, 0, \boxed{2}, 3, 4, 6, 8, 7, 10, 5]$

$l = 11$

$x = 2$

$[1, 0, 2]$ $[9, 3, 4, 6, 8, 7, 10, 5]$

rank(2) = 3 $6 - 3 = \boxed{3}$

rank(2) < 6

$[9, 3, \boxed{4}, 6, 8, 7, 10, 5]$ $i = 3$ — 3rd smallest

$\boxed{x = 4}$

$\boxed{5}$

Median Finding O(n) in average case
Time Analysis

Median Finding Fast Solution (Blum, Floyd, Pratt, Rivest and Tarjan)

Median Finding Fast Solution Time Analysis