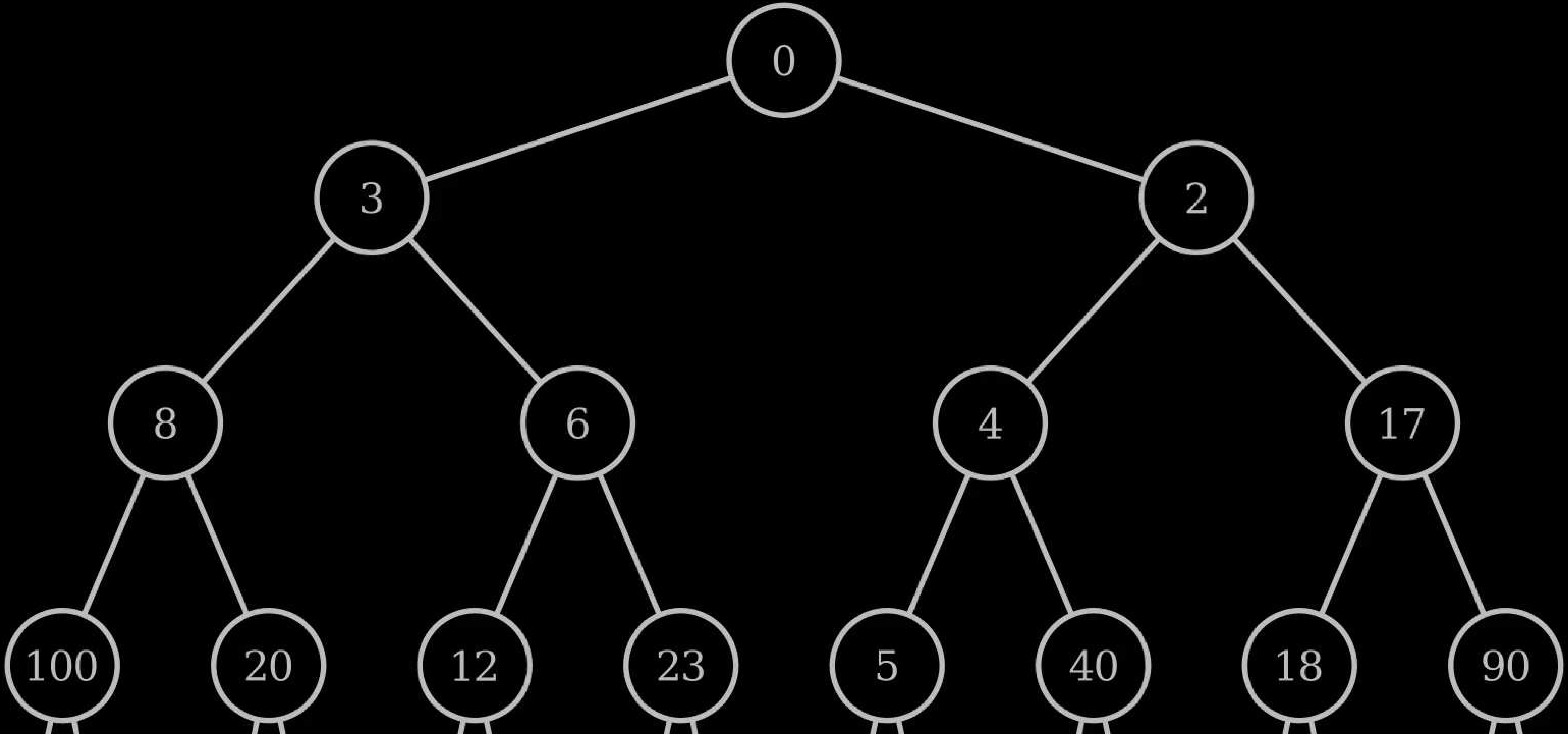
Selection in explorable heaps

Sander Borst (CWI), Daniel Dadush (CWI),
Sophie Huiberts (Columbia University), Danish Kashaev (CWI)

Selection in explorable heaps

finding the
nth smallest element

binary tree
with increasing values

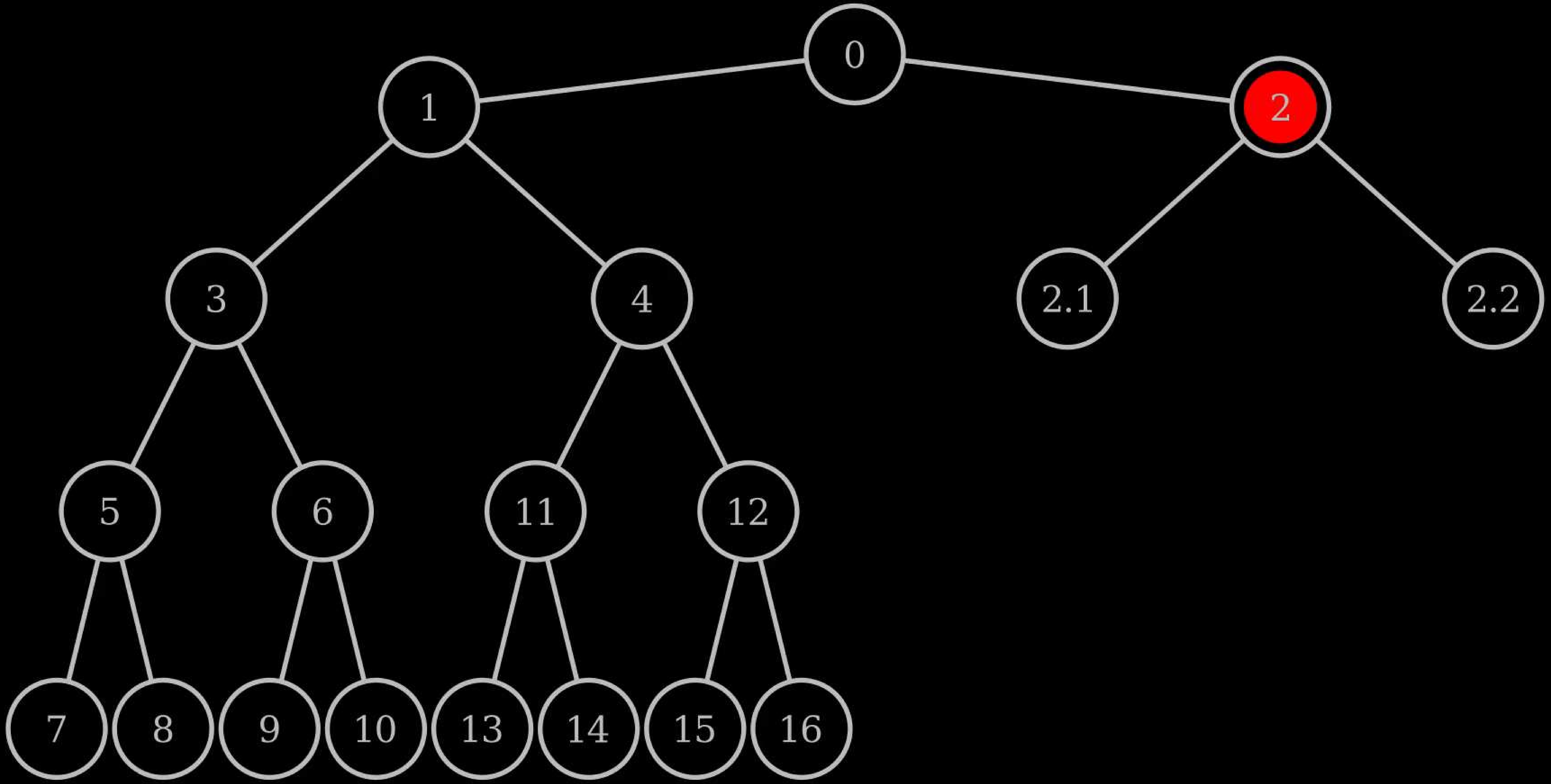


- Branch & Bound algorithm for solving MIP’s
- B&B-tree is a heap
- Node selection: Which subproblem to solve next?
- Travel is expensive

Best First Search

$O(n^2)$

Depth First Search



Prior work

- $O(n \cdot 2^{\sqrt{2\log(n)}})$ [KSW86]
- $O(n \cdot \log(n)^3)$ (our result)

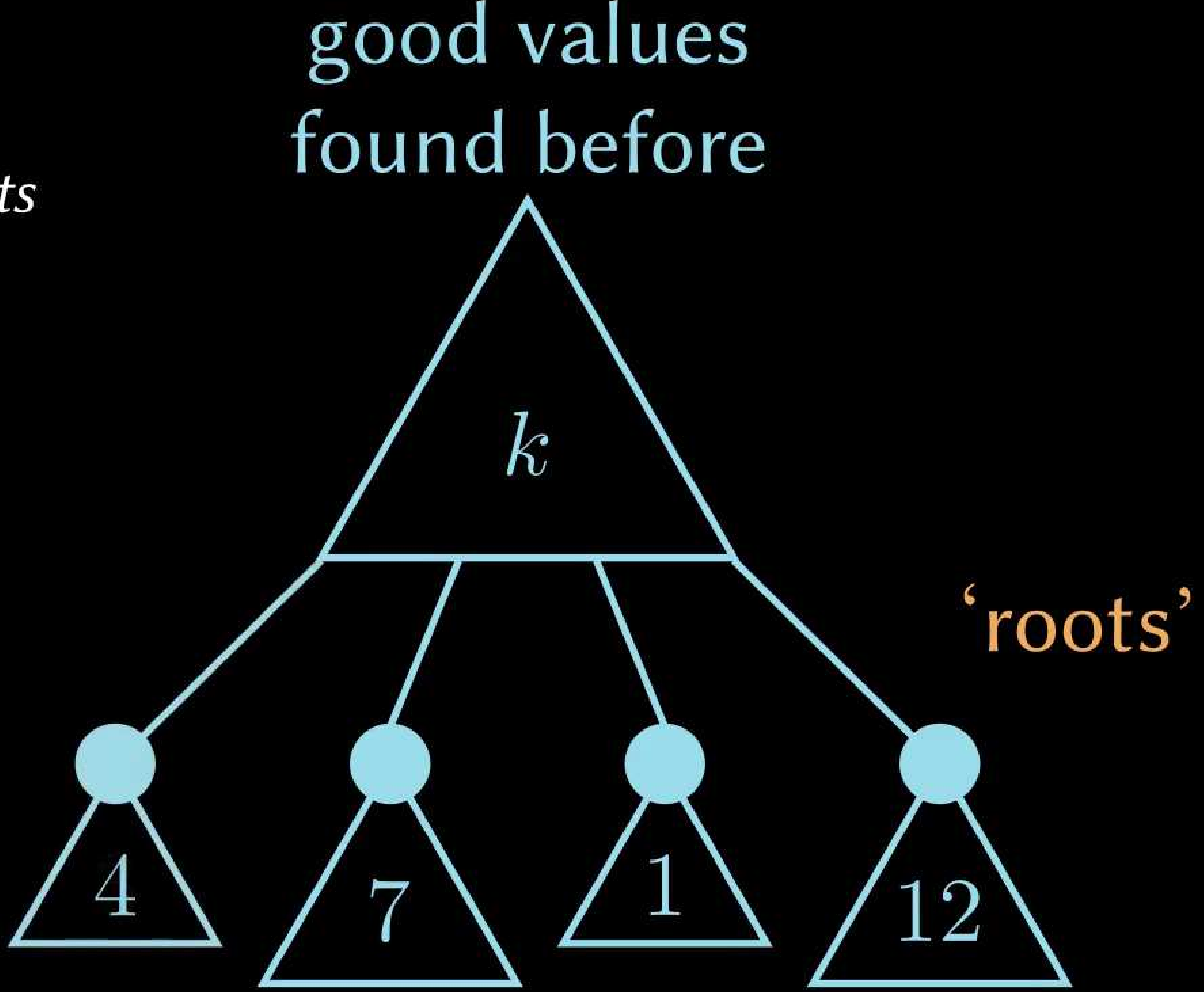
[KSW86] Richard M Karp, Michael E Saks, and Avi Wigderson. On a search problem related to branch-and-bound procedures. In *FOCS*, pages 19–28, 1986.

The algorithm

- Call the n smallest values the *good* values
- Can test if value is good in $O(n)$ time using DFS
- For set of m values: can test which are good using binary search in $O(n\log(m))$

The algorithm

- Find n smallest nodes, assuming that the $k \geq \frac{1}{2}n$ smallest are given
- Children of the k smallest nodes are called *roots*
- Take a random root:
  - Find n' smallest nodes under root by recursive call for $n' = 1, 2, 4, \ldots$
  - When a found node is not good, find highest good value under root



The algorithm

- Take a random root:
  - Find n' smallest nodes under root by recursive call for $n' = 1, 2, 4, \ldots$
  - When a found node is not good, find highest good value under root

$\log(n)$

$n \log(n)$

The algorithm

$\mathsf{ALG}(n, k)$

$\mathsf{ALG}(n_1^{(1)}, k_1^{(1)}), \ldots, \mathsf{ALG}(n_{r_1}^{(1)}, k_{r_1}^{(1)})$

$\mathsf{ALG}(n_1^{(2)}, k_1^{(2)}), \ldots, \mathsf{ALG}(n_{r_2}^{(2)}, k_{r_2}^{(2)})$

$\vdots \quad \vdots$

$$n \log(n)^2 \geq \sum_{i=1}^{r_1} n_i^{(1)} \log(n)^2 \geq \sum_{i=1}^{r_2} n_i^{(2)} \log(n)^2$$

Comparison

	Time	Space
[KSW86]	$n \cdot 2^{\sqrt{2\log(n))}}$	$\sqrt{\log(n)}$
New algorithm	$n\log(n)^3$	$\log(n)$

- Lower bound $\Omega(\frac{n\log(n)}{\log\log(n)})$ when in $O(\log(n))$ space
- In full space: only trivial lower bound known ($\Omega(n)$)

Open problems

- Closing the gap between $\Omega(n)$ and $n\log(n)^3$ in full space setting
- Is randomness needed?

Questions?