Median-of-Medians Selection: Deterministic Worst-Case $O(n)$

Problem

Given an array A of n distinct elements and an integer k with $1 \leq k \leq n$, find the k-th smallest element in A in deterministic worst-case linear time.

Algorithm (Median-of-Medians)

Algorithm 1 Select(A, k) — deterministic selection (median-of-medians)

Require: array A of size n, integer k with $1 \leq k \leq n$

if $n \leq 5$ **then**
    Sort A and **return** the k-th smallest element
end if
Partition A into $\lceil n/5 \rceil$ groups of size at most 5
for each group **do**
    find the median of the group (by sorting the group of size ≤ 5)
end for
Let M be the array of these medians (size $m = \lceil n/5 \rceil$)
$p \leftarrow$ Select$(M, \lceil m/2 \rceil)$ ▷ the median of medians
Partition A into $A_{<p}, \{p\}, A_{>p}$ (elements less than, equal to, and greater than p)
if $k \leq |A_{<p}|$ **then**
    return Select$(A_{<p}, k)$
else if $k = |A_{<p}| + 1$ **then**
    return p
else
    return Select$(A_{>p}, k - |A_{<p}| - 1)$
end if

Key Lemma (Quality of the pivot p)

Let n be the size of A. After grouping into groups of at most 5 and taking each group's median, and then taking p as the median of those medians, the pivot p satisfies:

$$\text{at least } \frac{3n}{10} \text{ elements of } A \text{ are } \leq p \quad \text{and} \quad \text{at least } \frac{3n}{10} \text{ elements of } A \text{ are } \geq p.$$

Thus, after partitioning around p, both sides have size at most $\frac{7n}{10}$.

Proof. Partition A into $\lfloor n/5 \rfloor$ full groups of 5 and possibly one smaller group. Consider only the full groups (ignore the smaller remainder group if any) — there are at least $\lfloor n/5 \rfloor$ such groups.

For each full group of 5, after sorting that group, its median is the 3rd smallest element in the group. If a group's median is $\leq p$, then in that group at least 3 elements are $\leq$ that median (hence $\leq p$). Similarly, if a group's median is $\geq p$, in that group at least 3 elements are $\geq$ that median (hence $\geq p$).

Since p is the median of the medians, at least half of the medians are $\leq p$ and at least half are $\geq p$. Consider the medians that are $\leq p$: there are at least $\lceil (\lfloor n/5 \rfloor)/2 \rceil$ such groups, and each contributes at least 3 elements that are $\leq p$. Therefore the number of elements $\leq p$ is at least

$$3 \cdot \left\lceil \frac{\lfloor n/5 \rfloor}{2} \right\rceil \geq 3 \cdot \left(\frac{n/5 - 1}{2} \right) = \frac{3n}{10} - \frac{3}{2}.$$

For sufficiently large n, this is at least $3n/10 - O(1)$. A symmetric argument gives the same lower bound for elements $\geq p$. Concretely, for $n \geq 5$ one can show that at least $\lfloor 3n/10 \rfloor$ elements lie on each side. Hence at most $n - \lfloor 3n/10 \rfloor - 1 \leq 7n/10$ elements can lie strictly on one side of p. Thus each partition side has size at most $\frac{7n}{10}$. ☐

Recurrence for the running time

Let $T(n)$ denote the worst-case time to select the k-th smallest element from n elements using this algorithm. Steps and their costs:

- Partition into groups of size at most 5 and find each group's median: each group of size ≤ 5 is sorted in $O(1)$ time, so total $O(n)$.
- Recursively select the median of medians from the $m = \lceil n/5 \rceil$ medians: cost $T(n/5)$ (up to constants).
- Partition around the pivot p: $O(n)$.
- Recurse on at most $7n/10$ elements (by the lemma).

Thus we obtain the recurrence (for sufficiently large n):

$$T(n) \leq T\left(\frac{n}{5}\right) + T\left(\frac{7n}{10}\right) + cn$$

for some constant $c > 0$.

Solving the recurrence

We prove $T(n) \leq Cn$ for some constant C and all n by induction. Choose C large enough so that base cases (small n) hold; we focus on the inductive step.

Assume the induction hypothesis holds for all smaller sizes. Then:

$$
\begin{aligned}
T(n) &\le C \cdot \frac{n}{5} + C \cdot \frac{7n}{10} + cn \\
&= C\left(\frac{1}{5} + \frac{7}{10}\right) n + cn \\
&= C\left(\frac{2}{10} + \frac{7}{10}\right) n + cn \\
&= C\left(\frac{9}{10}\right) n + cn.
\end{aligned}
$$

To make $T(n) \le Cn$, it suffices that

$$
C \cdot \frac{9}{10} n + cn \le Cn \quad \Longleftrightarrow \quad \left(1 - \frac{9}{10}\right) C \ge c \quad \Longleftrightarrow \quad \frac{1}{10} C \ge c \quad \Longleftrightarrow \quad C \ge 10c.
$$

So pick $C = \max\{10c, C_0\}$ where C_0 handles base cases. Then by induction $T(n) \le Cn$ for all n. Hence $T(n) = O(n)$.

(One may also solve the recurrence more formally using the recursion tree — the tree levels shrink geometrically and the sums of costs form a convergent geometric series bounded by a constant times n.)

Conclusion

The median-of-medians pivot guarantees that each recursive call reduces the problem size by a constant fraction in the worst case (no more than $7n/10$ remain on the larger side). The recurrence

$$
T(n) \le T(n/5) + T(7n/10) + O(n)
$$

resolves to $T(n) = O(n)$. Therefore the deterministic selection algorithm (median-of-medians) runs in linear time in the worst case.