

分治

岳镒

2025 年 3 月 7 日

主元素

设 A 是包含 n 个元素的数组，如果元素 x 在 A 中出现的次数大于 $n/2$ ，则称 x 是 A 的主元素。

- ▶ 假设 A 中元素可排序，设计算法判断 A 中是否存在主元素
- ▶ 假设 A 中元素不可排序，设计算法判断 A 中是否存在主元素

主元素

设 A 是包含 n 个元素的数组，如果元素 x 在 A 中出现的次数大于 $n/2$ ，则称 x 是 A 的主元素。

- ▶ 假设 A 中元素可排序，设计算法判断 A 中是否存在主元素
- ▶ 假设 A 中元素不可排序，设计算法判断 A 中是否存在主元素

$A = \{a_1, a_2, \dots, a_n\}$
 $\log \text{ poly}(n) = \log n$
 $\log n$
 \boxed{m}
 $\boxed{\text{cnt} \leftarrow 0}$
 $m \leftarrow \text{NULL}$

$O(n)$

a_i :

If $a_i = m$, $\text{cnt} \leftarrow \text{cnt} + 1$

If $a_i \neq m$ $\left\{ \begin{array}{l} \text{If } \text{cnt} > 0 \text{ } \text{cnt} \leftarrow \text{cnt} - 1 \\ \text{If } \text{cnt} = 0 \text{ } m \leftarrow a_i \\ \text{cnt} \leftarrow 1 \end{array} \right.$

If $\exists \text{主} \rightarrow x$. $\boxed{m = x}$

a_i contributes to cnt_i
 ± 1 .

$\boxed{\text{主}}$ contribution $\left\{ \begin{array}{l} \boxed{w^+} = \text{次} + 1 \\ w^- = \text{次} - 1 \end{array} \right.$

其他 $\left\{ \begin{array}{l} v^+ = \text{次} + 1 \\ \boxed{v^-} = \text{次} - 1 \end{array} \right.$

$\boxed{w^+ > v^-}$

① $w^+ + w^- > v^+ + v^-$

② $\text{cnt} \geq 0$

\Updownarrow
 $w^+ + v^+ \geq w^- + v^-$

① + ②

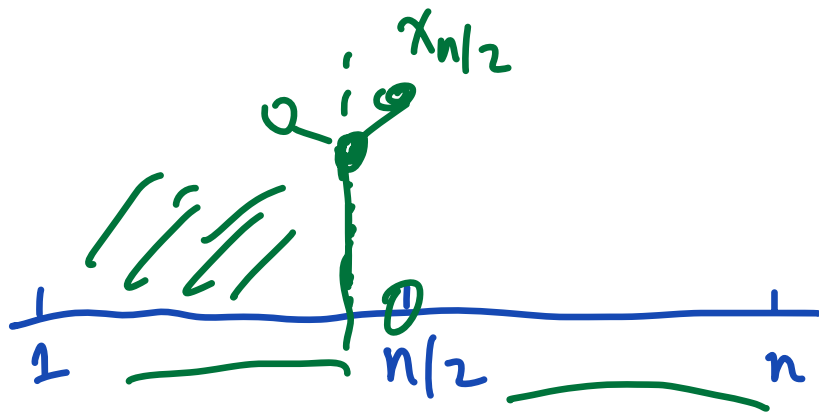
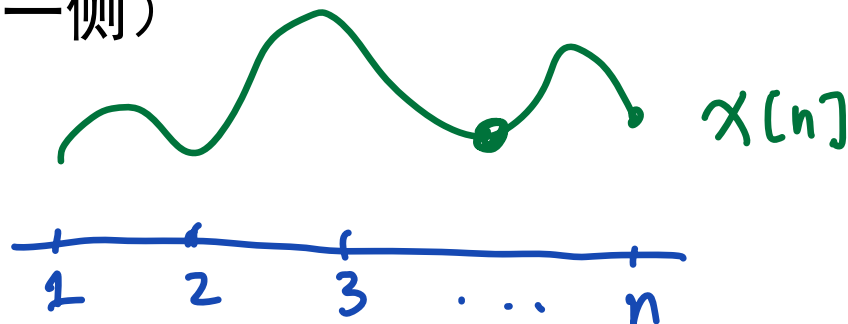
主元素

设 A 是包含 n 个元素的数组，设常数 $k \in \mathbb{Z}_+$ 。设计算法判断 A 中是否存在出现次数大于 n/k 的元素。

极小值点

$O(n) \checkmark$

设数组 $A = [x_1, x_2, \dots, x_n]$ 由 n 个互不相等的整数组成。设计一个比较次数尽可能少的算法，找到 A 中的一个极小值点（称下标 j 为极小值点，若 $x_j < \min\{x_{j-1}, x_{j+1}\}$ ；当 $j = 1, n$ 时只考虑一侧）



① $x_1 < x_{n/2}$

② $x_n < x_{n/2}$

③ $\begin{cases} x_1 > x_{n/2} \\ x_n > x_{n/2} \end{cases}$

$[1, \boxed{n/2-1}]$

$T(n) = T(n/2) + O(1) \quad O(\log n)$

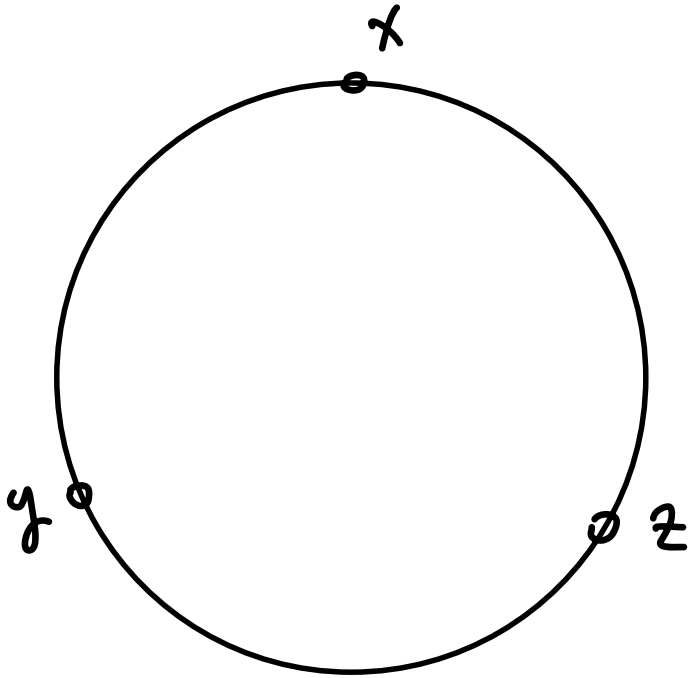
极小值点

设数组 $A = [x_1, x_2, \dots, x_n]$ 由 n 个互不相等的整数组成。设计一个比较次数尽可能少的算法，找到 A 中的一个极小值点（称下标 j 为极小值点，若 $x_j < \min\{x_{j-1}, x_{j+1}\}$ ；当 $j = 1, n$ 时只考虑一侧）

极小值点

$O(n)$

n 个互不相等的整数排列在圆环上。设计一个比较次数尽可能少的算法，找到圆环上的一个极小值点



$$x = \min \Rightarrow \widehat{y x z}$$

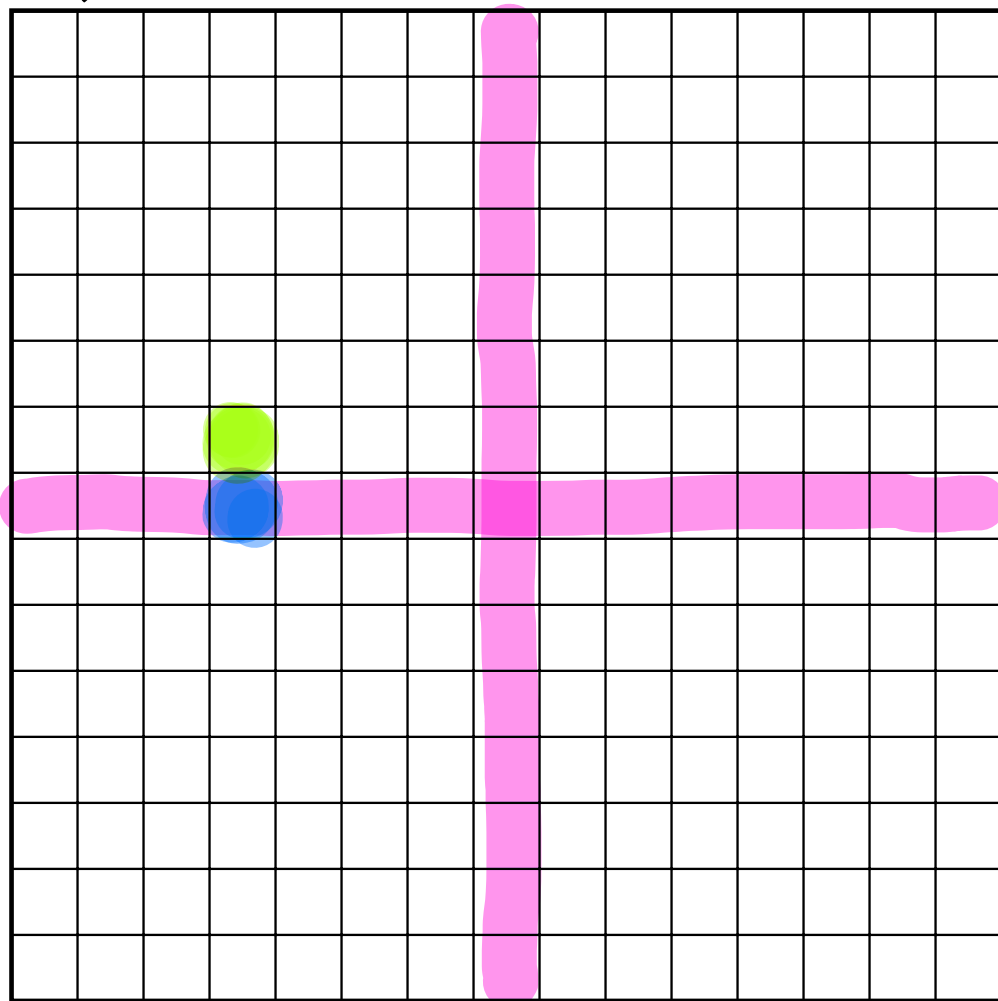
$$y = \min \Rightarrow \widehat{x y z}$$

$$z = \min \Rightarrow \widehat{x z y}$$

$$T(n) = T\left(\frac{2}{3}n\right) + O(1)$$

极小值点

在带权网格图中， mn 个顶点排列为 m 行 n 列，每个顶点的权值互不相等。如果一个顶点的权值比其相邻的 4 个顶点的权值都小，则称该顶点为一个极小值点。请设计一个尽可能高效的算法，在带权网格图中找出一个极小值点



提示：先找到网格中心“十字”上的最小值点，然后根据其两侧方格的大小决定进入哪一个子网格