

# 《逛街》解题报告

周雨扬

北京大学

2021 年 5 月 16 日

# 题目大意

给定长度为  $n$  的序列  $a_i$ ,  $Q$  次操作。

- 1  $l\ r$ :  $\forall l \leq i < r, a_i = \max\{a'_i, a'_{i+1}\}$ , 其中  $a'_i$  为未修改前  $a_i$  的权值。
  - 2  $l\ r$ : 求出  $\sum_{x \in S} a_x$ , 其中  $S = \{x | l \leq x \leq r, \forall l \leq i < x, a_i < a_x\}$ 。
- $1 \leq n, Q \leq 3 \times 10^5, 1 \leq a_i \leq 10^9, a_i$  互不相同。

# 备注

这里我们称元素  $a_x$  为一个严格前缀最大值当且仅当对于所有比  $x$  小的  $i$ , 都有  $a_i < a_x$ 。

原问题询问操作等价于询问子序列的所有严格前缀最大值的和。

# 算法 1

先来看看 subtask2。

不妨假设进行了  $m$  次的操作 1，则答案就是如下序列的所有严格前缀最大值的和：

$$\max\{a_l, a_{l+1}, \dots, a_{l+m}\}, a_{l+m+1}, \dots, a_{r+m}。$$

这里我们认为数组越界时的  $a_i$  的值恰好为 0。这是一个经典算法题，采用两个线段树维护即可。

# 算法 1

具体的，我们对于每个区间  $[l, r]$ ，求出所有严格前缀最大值的和，以及这个区间最大值。

询问初始时刻值为  $x$  的所有严格前缀最大值时由于取到的  $[l, r]$  的严格前缀最大值是一个后缀，因而可以在线段树上二分。

每一次 `pushup` 会调用一次二分，因而时间复杂度为  $O(n \log^2 n + Q \log^2 n)$ 。

## 算法 2

一般的，我们假设  $b_i$  表示在进行了一次操作 1 后，新的序列第  $i$  个元素为  $\max\{a_i, a_{i+1}, \dots, a_{b_i}\}$ 。

## 算法 2

一般的，我们假设  $b_i$  表示在进行了一次操作 1 后，新的序列第  $i$  个元素为  $\max\{a_i, a_{i+1}, \dots, a_{b_i}\}$ 。

每一次操作 1 后，不难发现我们对  $b_i$  序列进行了如下的修改。

- 将第  $r$  元素的一个拷贝放置在原先序列的第  $r+1$  个位置。
- 删除原先序列的第  $l$  个元素。

## 算法 2

将问题做一个变换:

假设存在序列  $c_i = \max\{a_{b_{i-1}+1}, a_{b_{i-1}+2}, \dots, a_{b_i}\}$ , 则不难发现原问题与下面的问题等价:

设  $x = \max\{a_l, a_{l+1}, \dots, a_{b_i}\}$ , 在初始值为  $x$  的时候对序列  $\{c_l, c_{l+1}, \dots, c_r\}$  做严格前缀最大值问题的答案。

正确性可以根据严格前缀最大值的性质证明, 这里略去。

## 算法 2

序列  $c_i$  为  $a_i$  的子序列，因而我们仍然可以按算法 1 中的做法维护类似的数据结构。

每一次修改只会删除  $c_i$  中的至多 1 个元素，并且插入在这个问题中可以忽略的 0。因而我们只需要不断地在序列  $a_i$  中删元素即可。

利用一颗额外的平衡树维护  $c_i$  下标的信息即可。时间复杂度  $O((n + Q) \log^2 n)$

## 算法 2.5

欸只有删元素能不能不用这个数据结构？

## 算法 2.5

欸只有删元素能不能不用这个数据结构?  
能!

## 算法 2.5

考虑这么一个树上结构：

- 假设  $p_i$  是比  $i$  大最小的坐标，使得  $a_i < a_{p_i}$ 。如果不存在规定  $p_i = n + 1$ 。
- 建立一棵树，使得  $i$  的父节点为  $p_i$ 。

则严格前缀最大值之和恰好为树链上元素之和。

## 算法 2.5

删元素会不会改变树的结构?

## 算法 2.5

删元素会不会改变树的结构?

不会!

如果将  $c_i$  合并到  $c_j$ , 则所有位于  $i+1 \sim j-1$  的已经被合并, 不可能成为父亲。且已知  $c_j$  一定是合法的父亲节点。

因而树结构不会改变, 此时是个脸蛋的树上单点修改, 树链求和。

时间复杂度  $O((n+Q)\log n)$ 。