

线段树与树状数组

wwwodddd

2019 年 4 月 9 日

目录

1 自我介绍	4
2 线段树	4
2.1 简介	4
2.2 打标记	4
2.3 矩形并	4
2.4 可持久化线段树	4
2.5 统计的力量	4
3 poj 1151	4
3.1 题目背景	4
3.2 题目大意	4
3.3 题目解法	5
4 bzoj 1798	5
4.1 题目背景	5
4.2 题目大意	5
4.3 题目分析	5
5 bzoj 2809	5
5.1 题目背景	5
5.2 题目大意	5
5.3 题目分析	5
6 bzoj 2957	5
6.1 题目背景	5
6.2 题目大意	5
6.3 题目分析	5

7	nowcoder 147H	6
7.1	题目背景	6
7.2	题目解法	6
7.3	我的代码	6
8	bzoj 4299	6
8.1	题目背景	6
8.2	题目大意	6
8.3	题目分析	6
9	Codeforces 280D	6
9.1	题目背景	6
9.2	题目大意	6
9.3	题目解法	6
10	ProjectEuler 663	6
10.1	题目背景	6
10.2	题目大意	6
10.3	题目分析	7
11	Luogu P4513	7
11.1	题目背景	7
11.2	题目大意	7
11.3	题目解法	7
12	hdu 5239	7
12.1	题目背景	7
12.2	题目大意	7
12.3	题目分析	7
13	spoj COT	7
13.1	题目背景	7
13.2	题目大意	8
13.3	题目解法	8
14	Codechef DGCD	8
14.1	题目背景	8
14.2	题目大意	8
14.3	题目解法	8
15	Atcoder Code Festival 2017 Exhibition B	8
15.1	题目背景	8
15.2	题目大意	8
15.3	题目解法	8

15.4 参考代码	9
16 树状数组	9
16.1 简介	9
16.2 一些推广	9
16.3 高维	9
16.4 第 k 大	10
16.5 差分	10
16.6 维护最大值	10
17 poj 2155	10
17.1 题目背景	10
17.2 题目大意	10
17.3 题目解法	10
18 poj 2828	10
18.1 题目背景	10
18.2 题目大意	10
18.3 题目解法	10
19 poj 3468	11
19.1 题目背景	11
19.2 题目大意	11
19.3 题目解法	11
20 poj 2559	11
20.1 题目背景	11
20.2 题目大意	11
20.3 题目解法	11

1 自我介绍

毕克

2010 ~ 2013, 石家庄二中。

NOI 2012 金牌。

2013 ~ 2017, 清华大学。

5 次区域赛/EC-FINAL 金牌。

没去过 IOI, 也没去过 ACM World Final。

三大爱好: 比赛, 出题, 讲课。

My vegetable exploded up.

QQ:751723392, Email:wwwodddd@gmail.com

2 线段树

2.1 简介

应用广泛的一个数据结构。

要求支持区间加法。

2.2 打标记

要求标记可以合并。

2.3 矩形并

因为有复杂度大概是 $4n^2$ 的方法, 所以很多题并不需要线段树。

2.4 可持久化线段树

可以部分替代划分树, 归并树, 等数据结构。

2.5 统计的力量

zkw 线段树。

然而感觉意义不大。

3 poj 1151

3.1 题目背景

poj 1151 Atlantis

3.2 题目大意

传说中经典的矩形面积并。

3.3 题目解法

但是因为 n 范围很小，直接离散化，暴力修改就可以通过。

4 bzoj 1798

4.1 题目背景

[Ahoi2009]Seq 维护序列 seq

4.2 题目大意

4.3 题目分析

区间加法，区间乘法，标记用 $kx + b$ 即可。

5 bzoj 2809

5.1 题目背景

[Apio2012]dispatching

5.2 题目大意

5.3 题目分析

确定了管理者后，一定选择子树内薪水最小的几个忍者。

因为对于所有管理者 M 相同，所以可以用可并堆。

也可以用 DFS 序转化成序列，然后通过可持久化线段树直接询问。

6 bzoj 2957

6.1 题目背景

楼房重建

6.2 题目大意

6.3 题目分析

线段树或者分块。

线段树的关键在于，每次合并两个节点，需要在右侧的树中进行一次询问。

分块的话不要自作多情使用分数。

7 nowcoder 147H

7.1 题目背景

Prefix Sum

7.2 题目解法

7.3 我的代码

我的代码

8 bzoj 4299

8.1 题目背景

bzoj 4299 FRBSUM

8.2 题目大意

8.3 题目分析

9 Codeforces 280D

9.1 题目背景

D. k-Maximum Subsequence Sum

9.2 题目大意

9.3 题目解法

可以维护 $3k$ 个变量，记录左最大，右最大，全局最大。

也可以同时维护最大，最小，和对应的方案。

每次询问 k 段时，贪心选取，最大，翻转正负，重复 k 。

10 ProjectEuler 663

10.1 题目背景

Sums of subarrays

10.2 题目大意

求（全局）最大子段和，带修改。

10.3 题目分析

直接线段树维护即可。

ProjectEuler 作为一个数学向的网站，难得出数据结构题。

11 Luogu P4513

11.1 题目背景

P4513 小白逛公园

11.2 题目大意

11.3 题目解法

经典问题，维护区间和，左最大，右最大，全局最大。

12 hdu 5239

12.1 题目背景

Doom

12.2 题目大意

输入一个数组，支持区间平方，和区间求和两个操作。

求和输出答案对 9223372034707292160 取模

12.3 题目分析

大概本机暴力下你会发现一个数一直平方之后会不变。

然后做法就类似于spoj QTREE4。

9223372034707292160 非常接近 long long，当然用 unsigned long long 配合快速幂是可以解决的。

还有另一个做法，注意到 $9223372034707292160 = 2^{31} \times (2^{32} - 1)$ 。

可以先分解成两部分，最后中国剩余定理合并。

13 spoj COT

13.1 题目背景

Count on a tree

13.2 题目大意

13.3 题目解法

经典区间第 k 大问题，用可持久化线段树即可。

14 Codechef DGCD

14.1 题目背景

Dynamic GCD

14.2 题目大意

一棵树，有点权。

要求：一条链加上一个值，区间询问 GCD。

14.3 题目解法

直接树链剖分。

区间修改，区间 GCD 是可以线段树解决的。

15 Atcoder Code Festival 2017 Exhibition B

15.1 题目背景

Atcoder Code Festival 2017 Exhibition B

15.2 题目大意

输入数列 $\{a_i\}$ 支持两个操作。

交换两个元素。

使一个元素加一。

问把输入的数组变成单调的，至少需要多少次操作。

15.3 题目解法

考虑数字一个一个加入，时刻维护单调不下降。

如果这个数字最终被改成了 v ，那么花费是 v + 比 v 大的数字个数 $-a_i$ 。

并且会造成影响，对于所有小于 v 的数字，花费都将增加 1（因为比他们大的数字多了一个）

我们需要用线段树，支持两个操作，区间加一，和区间求最小值。

一些实现技巧：注意到没出现在输入的序列里的数字，也不会出现在输出序列里，可以离散化建线段树。

造成的影响，如果实现成了所有小于 a_i 的数字，花费都将增加 1。

答案依旧是对的 (?)

我会证明了。

对于贪心做法来说，实际最优解大于询问的最小值 $w' > w$ 是没有意义的，因为这样不仅会多花费更大，也会让更多的位置 +1。

如果实际最优解小于 $w' < w$ ，那么相当于是通过花费一些代价，让 $w' \leq i < w$ 的位置 i 没有 +1。

但是事实上这是没有用的，一旦 w 的花费比 $w' \leq i < w$ 更好，因为修改花费都是前缀， w 将永远更好。

同样的，这也就是，为什么你实现成后者，答案正确的原因。

15.4 参考代码

我赛后（讲课前）补的

16 树状数组

16.1 简介

维护一个数组，支持修改一个点，区间求和。

如果直接暴力的话。

- 修改。 $O(1)$
- 询问。 $O(n)$

当然也可以暴力前缀和

- 修改。 $O(n)$
- 询问。 $O(1)$

树状数组把这两种做法综合了起来，使得复杂度变成了

- 修改。 $O(\log n)$
- 询问。 $O(\log n)$

设 $(\text{lowbit}(x))$ 是能整除 x 最大的 2 的次幂。

设 a_i 是原数组。

设 c_x 是满足 $x - \text{lowbit}(x) < i \leq x$ 的所有 a_i 的和。

16.2 一些推广

树状数组的操作必须支持减法，常见的有加法，异或，部分乘法（乘法逆元）

16.3 高维

树状数组相对于线段树的优势，就是可以很轻松的推广到高维的情况。

16.4 第 k 大

设 a_i 表示 i 出现的次数，我们可以通过二分，求和的方式询问第 k 大。

16.5 差分

对于修改区间，询问单点的可以用差分解决。

对于修改区间，询问区间的，可以用两个树状数组解决。比如 POJ3468

16.6 维护最大值

如果修改的时候，只会把一个位置改的更大。

询问的时候，只询问前缀。

也可以用树状数组。

17 poj 2155

17.1 题目背景

poj 2155 Matrix

二维树状数组的修改与查询。

把加法改为异或。

二维差分。

17.2 题目大意

17.3 题目解法

修改只需要修改四个角。

询问一个矩形内所有修改的异或。

18 poj 2828

18.1 题目背景

poj 2828 Buy Tickets

树状数组基本的修改，求第 k 个 1。

18.2 题目大意

18.3 题目解法

倒着考虑，一个人一个人的安排。

可以用树状数组或者线段树。

如果用树状数组，大概需要实现类似于找第 k 个 1 的操作。

19 poj 3468

19.1 题目背景

A Simple Problem with Integers

19.2 题目大意

一个数组 a_i ，支持区间加一个数字，区间求和。

19.3 题目解法

设 $b_i = a_i - a_{i-1}$ 。

如果求 a_i 的前 n 项和。

$$\sum_{i=1}^n a_i = \sum_{i=1}^n \sum_{j=1}^i b_j = \sum_{j=1}^n (n-j+1)b_j = (n+1)\left(\sum_{j=1}^n b_j\right) - \left(\sum_{j=1}^n j b_j\right)$$

用两个树状数组，分别维护 b_j 和 $j b_j$ 的前缀和。

20 poj 2559

20.1 题目背景

Largest Rectangle in a Histogram

20.2 题目大意

输入一个数组 h_i ，表示第 i 个板子的宽度是 1 高度是 h_i 。

问最大子矩形。

20.3 题目解法

单调栈求最大矩形。