

bzoj-2259 新型计算机

标签: [bzoj](#) [DP](#) [单调栈](#) [并查集](#)

2015-10-27 08:32

256人阅读

[评论\(0\)](#)

[收藏](#)

[举报](#)

分类: [数据结构 \(86\)](#) [bzoj \(139\)](#) [动态规划 \(30\)](#)

题意:

给出一个长度为 n 的非负序列, 将一个元素 a 修改为 A 的代价是 $|a-A|$;

求最小的代价使序列合法 (合法的概念参照原题);

$1 \leq n \leq 1000000$;

题解:

这道题据说要卡 $O(n \log n)$, 然而我依然选择用 $O(n^2)$ 的算法AC了此题[滑稽];

我们可以很容易的得到一个 $O(n^2)$ 的算法;

设 $f[i]$ 为从 i 开始到序列末尾使序列合法所花费的最小代价, $A[i]$ 为 $i+a[i]+1$;

转移即为 $f[i] = \min(f[j] + \text{abs}(j - A[i]))$;

暴力转移是 $O(n^2)$, 这里我们也可以用线段树优化成 $O(n \log n)$;

但是线段树的常数太大了, 哪怕是ZKW也会比 $O(n)$ 慢不知道哪里去了(虽说BZ已经可A了);

我们考虑将转移方程中的 abs 分情况讨论, 那么就是以 $A[i]$ 为分界的两段方程;

而这两个方程一个可以用单调栈优化, 另一个其实可以 $O(1)$ 出解 (我写的比较凌乱所以都单调栈了==);

单调栈上可以利用二分找到分界点, 比线段树常数要小;

除此以外, 实际上我们考虑每一个元素在单调栈中是被谁干掉的, 维护这样一个并查集;

这个并查集指向的元素其实就是我们要找的决策点了;

时间复杂度似乎是玄学? 不过真的比二分快那么一点。。。