The 13th Northeast Collegiate Programming Contest Editorial

By Claris & quailty

2019 年 5 月 22 日

Overview

	Easiest									Hardest
Idea	J	B	C	E	G	H	F	D	I	A
Coding	J	E	G	H	B	C	F	A	D	I
Summary	J	B	E	C	G	H	F	D	A	I

J. Time Limit

Author: Claris

Shortest Judge Solution: 319 Bytes

Description

给定一道题目的标程和若干验题代码的运行时间，按照规则设定时限。

- $n \leq 10$。
- $a_i \leq 10$。

Solution

- 按照题意模拟。

B. Balanced Diet

Author: Claris

Shortest Judge Solution: 948 Bytes

Description

n 颗糖果，每颗糖果价值为 a_i，种类为 b_i。

选择一些糖果，使得总价值除以同种类糖果数量的最大值最大。

如果第 i 类糖果选了，那么至少要选择 l_i 个。

- $n, m \leq 100000$。

Solution

- 设 f_i 表示每种糖果选择不超过 i 种的最大价值，则 $ans = \max(\frac{f_i}{i})$。
- 每种糖果按价值从大到小排序，对 f 的贡献是一个后缀。
- 贡献相同的部分用前缀和实现。
- 时间复杂度 $O(n \log n + m)$。

E. Minimum Spanning Tree

Author: quailty

Shortest Judge Solution: 548 Bytes

Description

给定一棵 n 个点的树 G，每条边有权值。

G 的线图 $L(G)$ 中连接 (u, v) 的边权为它们对应树上两条边的边权之和。

求 $L(G)$ 的最小生成树的边权和。

- $n \le 100000$。

Solution

- 考虑树上每个点 i 对答案的贡献。
- 如果 i 的度数小于 2，那么不会有贡献。
- 否则 i 连出去的这些边在线图中形成一个团。
- 一个团的最小生成树即为每个点向权值最小的点连边。
- 时间复杂度 $O(n)$。

C. Line-line Intersection

Author: quailty

Shortest Judge Solution: 1166 Bytes

Description

给定平面上 n 条直线，统计有多少对直线存在公共点。

- $n \leq 100000$。

Solution

- 两条直线有公共点当且仅当它们重合或者不平行。
- 重合的直线：求出直线表达式 $ax + by + c = 0$，约分后统计。
- 平行或重合的直线：将表达式中 c 置零，约分后统计。
- 答案为总方案数减去平行或重合的方案数加上重合的方案数。
- 时间复杂度 $O(n \log n)$。

G. Radar Scanner

Author: Claris

Shortest Judge Solution: 605 Bytes

Description

给定平面上 n 个边平行坐标轴的矩形，每个矩形覆盖若干方格。

用最少的操作次数上下左右移动这些矩形，使得存在一个方格被所有矩形覆盖。

- $n \leq 100000$。

Solution

- 两维独立，考虑一维的情况。
- 假设最终交点为 x，则线段 $[l, r]$ 移动到 x 的代价为 $\frac{|l-x|+|r-x|-(r-l)}{2}$。
- 除以 2 以及减去 $r-l$ 的部分与 x 无关，只需要最小化 $\sum(|l-x|+|r-x|)$。
- 经典问题，当 x 取所有 l, r 的中位数时最优。
- 时间复杂度 $O(n)$。

H. Skyscraper

Author: Claris

Shortest Judge Solution: 940 Bytes

Description

n 座摩天楼从左往右排成一排，第 i 座摩天楼的预计层数为 a_i。

m 次操作：

1. 将 a 的某个区间 $[l, r]$ 的所有数都加上 k。
2. 给定 $[l, r]$，问假如仅对 $[l, r]$ 的摩天楼从零开始施工，最少需要几个阶段。

每个阶段可以选择一个区间，将该区间内所有摩天楼往上修建一层。

- $n, m \leq 100000$。

Solution

- 考虑静态的情况，令 $a_0 = 0, b_i = a_i - a_{i-1}$。
- 若 $b_i > 0$，则至少需要多花 b_i 个阶段才能完工。
- 若 $b_i \le 0$，则 $i-1$ 的施工可以连带着把 i 给修好。
- 令 $c_i = b_i[b_i > 0]$，对于区间 $[l, r]$，最少阶段数为 $a_l + c_{l+1} + c_{l+2} + \cdots + c_r$。
- 即 $(b_1 + b_2 + \cdots + b_l) + (c_{l+1} + c_{l+2} + \cdots + c_r)$。

Solution

- 考虑区间加操作，对 b 和 c 的影响为 l 和 $r+1$ 两处单点修改。
- 树状数组维护 b 和 c 的前缀和。
- 时间复杂度 $O((n+m)\log n)$。

F. Mini-game Before Contest

Author: Claris & quailty

Shortest Judge Solution: 1674 Bytes

Description

两支队伍在一张 n 个点 m 条边的有向图上玩游戏，6 个人轮流移动棋子行动一步，不能移动的人所属队伍输。

正常人希望赢，不能赢则希望平局；而演员希望输，不能输则希望平局。

对于每个点计算棋子放在该点开始游戏时最后的游戏结果。

- $n \leq 100000$。
- $m \leq 200000$。

Solution

- 状态：$f_{i,j}$ 表示棋子位于 i，目前轮到第 j 个人行动时最终的游戏结果。
- 定义 -1 表示 A 胜，0 表示平局，1 表示 B 胜，则每个状态的转移是 min 或者 max。
- 一开始假设所有状态的值都是 0，并维护出 cnt 数组表示每个状态后继状态中有几个 $-1, 0, 1$，用于 $O(1)$ 计算一个状态的值。
- 对于所有不能行动的状态，重新设定它的状态值，并加入队列。

Solution

- 用 SPFA 的方式依次松弛每个值发生改变的状态的前驱状态，并更新 cnt 数组。
- 每个状态的值只会在 $-1, 0, 1$ 之间切换常数次。
- 时间复杂度 $O(n+m)$。

D. Master of Data Structure

Author: Claris

Shortest Judge Solution: 2181 Bytes

Description

给定一棵 n 个点的树，每个点有权值 w_i，一开始每个点的权值都是 0。

需要支持 m 次操作，每次操作是给定的 7 种路径点权操作之一。

- $n \leq 500000$。
- $m \leq 2000$。

Solution

- 注意到 m 非常小。
- 定义每个操作的链的两端点 u 和 v 为关键点。
- 一共有 $O(m)$ 个关键点。
- 求出这些关键点的虚树，虚树点数也只有 $O(m)$，每条虚树边表示若干个被压缩的点。
- 对于每个操作，在虚树上暴力模拟即可，注意特判边长为 0 的情况。
- 时间复杂度 $O(n + m^2)$。

A. Apple Business

Author: Claris

Shortest Judge Solution: 1694 Bytes

Description

给定一棵 n 个点的完全二叉树，每个点有 a_i 个苹果。

给定 m 份订单，每份订单指定了一条 u 到 v 的路径，最多收购这些苹果树上 c 个苹果，每个苹果给 w 元钱。

请找到一种销售方式使得总收益最大。

- $n \leq 100000, m \leq 100000$。
- 保证 u 是 v 的祖先。

Solution

- 将所有订单按照 w 从大到小排序，依次考虑每份订单。
- 如果当前的订单能满足，那么满足它。
- 如果当前的订单不能满足，那么要满足它只能舍弃前面的订单，但是前面的订单比它付的钱更多，所以不划算。
- 只需要求出在不改变前面所有订单的销量下，当前订单最多能满足多少。

Solution

- 考虑暴力判定。
- 建立一张左边 m 个点右边 n 个点的二分图，左边每份订单向右边对应路径的苹果树连边。
- S 向左边每份订单连边，容量为实际销量。
- 右边每个苹果树向 T 连边，容量为苹果数量。
- 一个销售方案合法当且仅当 S 到每份订单的边都满流。

Solution

- Hall 定理：二分图左边存在完备匹配当且仅当对于左边任意一个 k 个点的子集 U，U 都至少与右边 k 个点相连。
- 在之前建立的图中，Hall 定理可以表述为：
- 满流当且仅当对于任意一个订单子集 U，假设它们的实际销量和为 A，它们对应的路径的并集为 V，则 V 中所有苹果树的苹果数量之和 B 不能小于 A。

Solution

- 若干条路径的并集在树上是多个连通块。
- 只需要考虑那些最紧的限制：路径的并集是一个连通块的情况。
- 即：对于树上任意一个连通块 S，S 的苹果数量之和不能小于两端点都在 S 中的订单的实际销量之和。

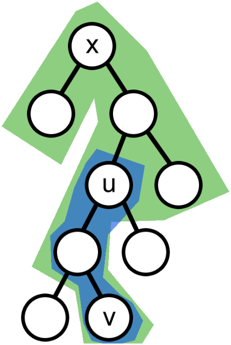
Solution

- 考虑如何计算当前订单 (u, v, c) 的实际销量 d。
- 设 f_x 表示考虑以 x 为根的子树，包含 x 点的连通块中，苹果数量之和减去两端点都在连通块中的订单的实际销量之和的最小值。
- 设 $g_{x,y}$ 表示为了计算 f_x，考虑以 y(x 是 y 的祖先) 为根的子树，包含 y 点的连通块中，苹果数量之和减去两端点都在连通块中的订单的实际销量之和的最小值。
- 完全二叉树的情况下总状态数为 $O(n \log n)$。

Solution

- $f_x = g_{x,x}$，考虑如何计算 $g_{x,y}$。
- 设 $b_{x,y}$ 表示为了计算 f_x，点 y 的权值，一开始 $b_{x,y} = y$ 点的苹果数量。

Solution

- 

- 对于之前的某份订单 (u, v)，实际销量为 d，那么当且仅当 x 是 u 的祖先且连通块包含 v 时，该订单两端点都被包含在连通块中。

Solution

- 枚举 u 的祖先 x，将对应的 $b_{x,v}$ 都减去 d。
- 则 $g_{x,y}$ 只需要寻找包含 y 的连通块的 b 的和的最小值，有 $g_{x,y} = b_{x,y} + \min(g_{x,2y}, 0) + \min(g_{x,2y+1}, 0)$。
- 在将 $b_{x,v}$ 减去 d 的同时，需要重新计算 v 到 x 路径上所有点 y 的 DP 值 $g_{x,y}$。
- 一共需要枚举 $O(\log n)$ 个 x，每个 x 内部要枚举 $O(\log n)$ 个 y。
- 加入每份订单的影响的时间复杂度为 $O(\log^2 n)$。

Solution

- 回到之前的问题，如何计算当前订单 (u, v, c) 的实际销量 d？
- 枚举 u 的祖先 x，用类似的 DP 计算最高点是 x 且同时包含 x 和 v 的连通块的 b 的和的最小值 tmp。
- 若 $tmp - d \geq 0$ 则合法，即 $d \leq tmp$。
- 故 $d = \min(c, tmp)$。
- 计算每份订单的实际销量的时间复杂度为 $O(\log^2 n)$。
- 总时间复杂度 $O(n \log n + m \log m + m \log^2 n)$。

I. Temperature Survey

Author: quailty

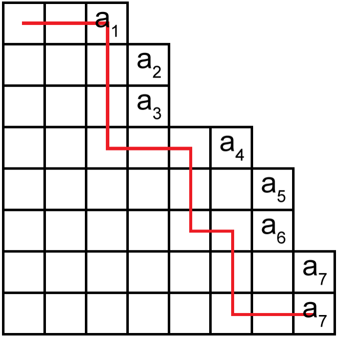
Shortest Judge Solution: 3368 Bytes

Description

给定一个不下降序列 $a_1, a_2, \ldots, a_n$ ，统计有多少不下降正整数序列 $b_1, b_2, \ldots, b_n$ 满足 $b_i \leq a_i$。

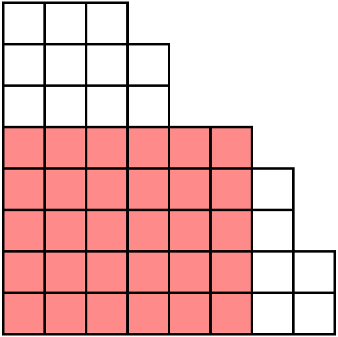
- $n \leq 200000$。
- $a_i \leq n$。

Solution

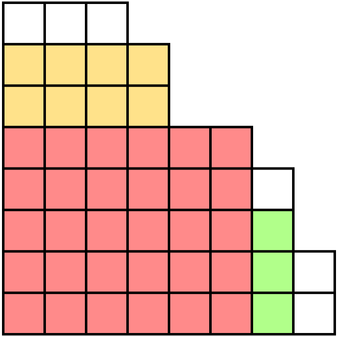
- 

- 令 $a_{n+1} = a_n$，方案数等同于上图中从左上角向右向下走到右下角的路径数。

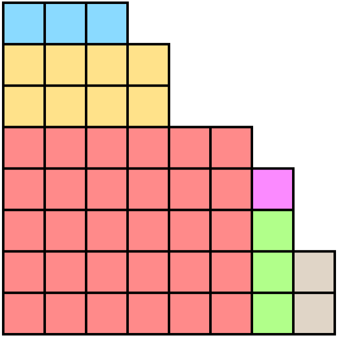
Solution

- 
- 不断取 $a_{\frac{l+r}{2}}$ 作为矩形的右上角，可以递归地划分出 $O(n)$ 个矩形。

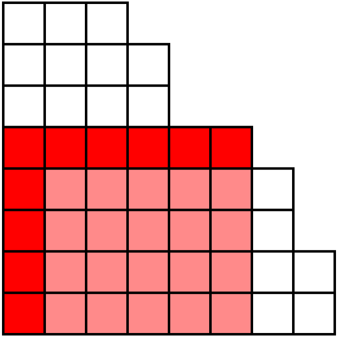
Solution

- 
- 不断取 $a_{\frac{l+r}{2}}$ 作为矩形的右上角，可以递归地划分出 $O(n)$ 个矩形。

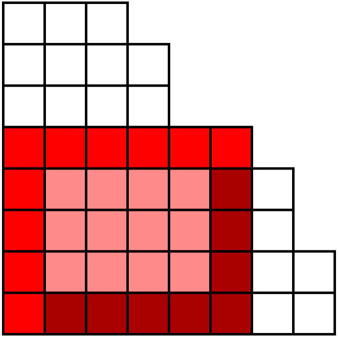
Solution

- 
- 不断取 $a_{\frac{l+r}{2}}$ 作为矩形的右上角，可以递归地划分出 $O(n)$ 个矩形。

Solution

- 
- 设 $f_{i,j}$ 表示从左上角走到 (i,j) 的方案数，每个矩形的左边界和上边界可以暴力转移求出：$f_{i,j}=f_{i-1,j}+f_{i,j-1}$。

Solution

- 
- 考虑左边界和上边界对右边界和下边界的贡献，是组合数卷积 DP 值的形式，可以 NTT 求出右边界和下边界的 DP 值。

Solution

- 一个边长为 $w \times h$ 的矩形需要支付 $O((w+h)\log(w+h))$ 的代价去 NTT。
- 结论：所有矩形的边长之和为 $O(n\log n)$，总时间复杂度 $O(n\log^2 n)$。
- 证明：划分矩形的递归过程只有 $O(\log n)$ 层，同一层的矩形在两个维度上都不相交，每层边长和为 $O(n)$，故总边长和为 $O(n\log n)$。

Thank you!