

IOI2020 中国国家集训队第一阶段作业 试题准备

江苏省常州高级中学 卞浏予

November 20, 2019

Contents

1	Codeforces 571D Campus	2
1.1	题目大意	2
1.2	数据范围与约定	2
1.3	解题过程	2
1.3.1	解法一	2
1.3.2	解法二	3
1.4	参考文献	3
2	Codeforces 708E Student's Camp	4
2.1	题目大意	4
2.2	数据范围与约定	4
2.3	解题过程	4
2.4	参考文献	5
3	arc093f Dark Horse	6
3.1	题目大意	6
3.2	数据范围与约定	6
3.3	解题过程	6
3.4	参考文献	7

1 Codeforces 571D Campus

题目来源: <https://codeforces.com/contest/571/problem/D>

1.1 题目大意

维护两类集合和一个数组, 两类集合初始时都有 N 个且第 i 个集合里只有元素 i , 第一类集合支持集合合并和集合中每个元素所对应的下标加上集合的大小, 第二类集合支持集合合并和集合中每个元素所对应的下标上数值清零, 要求实时查询数组中的某一个元素。

1.2 数据范围与约定

对于所有数据, 保证 $1 \leq n, m \leq 5 \cdot 10^5$ 。

时间限制: $2s$

空间限制: $256MB$

1.3 解题过程

首先, 我们假设对于每一组询问我们已经知道这个位置上一次清零的时间, 此时问题就会变得很简单, 我们就可以不管所有 M 和 Z 操作, 用询问时位置上的数减去上一次清零时位置上的数就是答案。

有了以上结论, 我们只要解决以下两个问题:

1. 用所有 M 和 Z 操作求出每次询问上一次清零的时刻。
2. 用所有 U 和 A 操作求出某些时刻某个位置上的数值。

接下来, 我们讨论如何解决这两个问题, 由于第二个问题需要用到第一个问题的信息, 所以需要离线处理信息。这里提供两种解法。

1.3.1 解法一

我们使用启发式合并来解决, 启发式合并就是把小的集合合并给大的。

对于第一个问题, 我们要维护一些信息, 对于每一个集合, 维护整个集合中的元素以及整个集合上一次被清空的时间, 对于每个元素, 维护这个元素属于哪一个集合, 加入集合的时间和加入集合前上一次被清空的时间。

合并集合时, 我们只需要把小的集合中的元素加入大的集合, 注意, 大的集合上一次被清空的时间不需要改变, 而加入的元素加入前上一次清空的时间就需要分类讨论一下。

对于第二个问题, 解法类似, 只是维护的清空的时间变成增加的值。

由于每个元素在被加入别的集合时，所在集合大小至少增加一倍，所以最多被加入 $\log(n)$ 次，总的时间复杂度为 $O(n\log(n) + m)$ ，空间复杂度为 $O(n + m)$ 。

1.3.2 解法二

对于第一个问题，由于合并的关系构成了一个森林，我们先离线把森林构建出来，并按照森林的 *dfs* 序建立线段树，此时集合清零就代表了线段树上的一个区间赋值，询问就代表单点查询，使用带 *tag* 的线段树即可完成。

对于第二个问题，与上面类似，只是区间赋值变成区间加。

复杂度与解法一类似，总的时间复杂度为 $O(n\log(n) + m)$ ，空间复杂度为 $O(n + m)$ 。

1.4 参考文献

<https://codeforces.com/blog/entry/19923>

2 Codeforces 708E Student's Camp

题目来源: <https://codeforces.com/contest/708/problem/E>

2.1 题目大意

有一个 $(n+2) * m$ 的长方形, 除了第一行和最后一行, 其他每一行每一天最左边和最右边的格子都有一确定概率被摧毁, 每行之间独立且左边和右边独立, 求 k 天之后最上面一行与最下面一行四联通的概率。

2.2 数据范围与约定

对于所有的数据, 保证 $1 \leq n, m \leq 1500, 1 \leq a, b \leq 10^9, 0 \leq k \leq 10^5$ 。

时间限制: 3s

空间限制: 256MB

2.3 解题过程

首先, 令 $dp[t][l][r]$ 表示前 t 行联通且第 t 行剩下的是区间 $(l, r]$ 的概率。(剩下的一定是一段连续的区间)。

那么我们就可以写一个十分简单的转移

$$dp[t][l][r] = p_{l,r} \sum_{lp,rp} dp[t-1][lp][rp]$$

条件是 $(l, r] \cap (lp, rp] \neq \emptyset$, 其中 $p_{l,r}$ 表示 k 天后当前行剩下的区间是 $(l, r]$ 的概率。其中 $p_{l,r} = p_l \times p_{(m-r)}$, 而 $p_l = \binom{k}{l} p^l (1-p)^{k-l}$ 。

现在, 我们定义

$$dp_r[t][r] = \sum_{l < r} dp[t][l][r]$$

和

$$sum_dp_r[t][r] = \sum_{a \leq r} dp_r[t][a]$$

那么

$$dp[t][l][r] = p_{l,r} (sum_dp_r[t-1][m] - sum_dp_r[t-1][l] - sum_dp_l[t-1][r+1])$$

这里我们减去了所有在线段左边或右边的情况。

现在我们不再计算 $dp[t][l][r]$ 而计算 $dp_r[t][r]$ 。

$$dp_r[t][r] = \sum_{l < r} p_{l,r} (sum_dp_r[t-1][m] - sum_dp_r[t-1][l] - sum_dp_l[t-1][r+1])$$

这里我们可以 $O(m)$ 来计算其中一个值, 显然, 如果我们预处理一些值, 用 $O(m)$ 的时间计算所有也是可行的, 例如我们需要预处理 p_l 的前缀和以及 $p_l \times dp_r[t-1][l]$ 的前缀和等等。

而 sum_dp_r 数组是 dp_r 数组的前缀和, 预处理即可。

综上, 我们在 $O(nm)$ 的时间内完成了这道题。

2.4 参考文献

<https://codeforces.com/blog/entry/46763>

3 arc093f Dark Horse

题目来源: https://atcoder.jp/contests/arc093/tasks/arc093_d

3.1 题目大意

有 2^n 名选手, 编号为 1 至 2^n 。现在这 2^n 名选手将进行 n 轮淘汰赛, 决出胜者。若 $x < y$, 则 x 能够战胜 y 。但有 m 个例外, 1 号选手会输给这 m 个选手。问有多少中排列方式使得 1 号选手取得胜利。

3.2 数据范围与约定

对于所有数据, 保证 $1 \leq n \leq 16, 0 \leq m \leq 16, 2 \leq a_i \leq 2^n (1 \leq i \leq m), a_i < a_{i+1} (1 \leq i < m)$ 。

时间限制: $2s$

空间限制: $256MB$

3.3 解题过程

首先假设 $p_1 = 1$, 最后把答案乘以 2^n 即可。

玩家 1 可以获胜当且仅当他可以战胜以下几位选手:

$$p_2, \min\{p_3, p_4\}, \dots, \min\{p_{2^{(n-1)}+1}, \dots, p_{2^n}\}$$

那么我们只需要统计以上 n 个元素不在集合 A 中的排列的数量。

考虑 n 个集合组成的集合 G : $G_1 = \{p_2\}, G_2 = \{p_3, p_4\}, \dots, G_n = \{p_{2^{(n-1)}+1}, \dots, p_{2^n}\}$, 那么 G 中每个集合中最小的元素不在 A 集合中。这里我们要用容斥原理, 对于集合 G 的每个子集 S 我们需要计算 $f(S)$, 表示满足以下条件的排列数:

- 对于 S 中的每个集合 g , 满足 g 中最小的元素在集合 A 中。

为了计算这些值, 我们要使用以下的一个 dp 。假设 $A_1 > A_2 > A_3 > \dots > A_m$, 对于每个 $i = 1, 2, \dots, m$, 按顺序做以下两种转移之一:

1. 对于某个 k , 选 $2^k - 1$ 个标号比 A_i 大的选手, 把他们和 A_i 分配给 G_k , 这样, G_k 中最小的元素就在集合 A 中间了。
2. 什么都不选。

对于每一个 $0 \leq i \leq m$ 和集合 $S \in \{G_1, \dots, G_n\}$, 定义 $dp_{i,S}$ 为我们已经决定了玩家 $A_1, A_2, A_3, \dots, A_m$, 且集合 S 已经被填满的方案数。(其他集合是空的) 那么, 我们就可以用 $dp_{m,S}$ 乘上一些系数来得到 $f(S)$, 最后的答案就是 $\sum f(S)(-1)^{|S|}$ 。

时间复杂度: $O(2^n * n * m)$ 。

3.4 参考文献

<https://img.atcoder.jp/arc093/editorial.pdf>