# 省选 2024 模拟赛 Day 1 (2.16) 题解

**By YeahPotato**

2024.2.16

# T1

## 算法 1

$f_{i,j}(i>j)$ 表示当前已经染好了前 $i$ 个，两类颜色的末端分别在 $i,j$ 的方案数。

暴力是 `2D/0D` 的，可通过前 $2$ 个子任务：

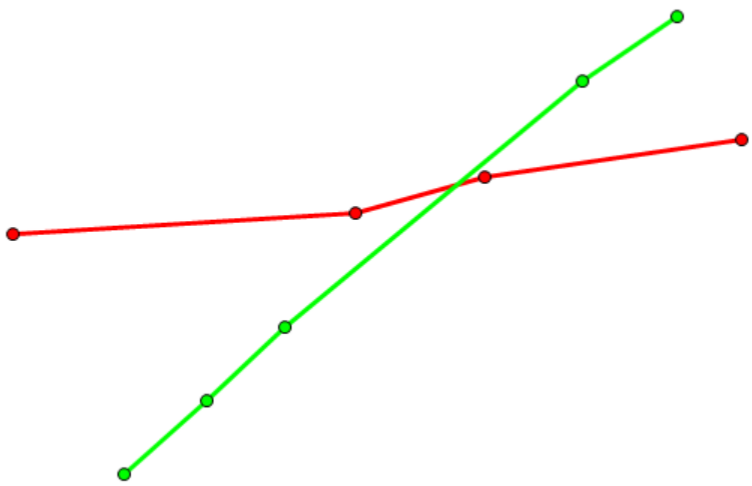
$$f_{i,j}=[j<i-1\wedge a_{i-1}<a_i]f_{i-1,j}+[j=i-1]\sum_{k=0}^{i-2}[a_k<a_i]f_{i-1,k}$$

可以按套路用权值线段树维护，可通过前 $3$ 个子任务。

## 算法 2

记 $a_i>a_{i+1}$ 的情况为一个"下降"。

考虑一个下降 $(a_i,a_{i+1})$，及其后一个下降 $(a_j,a_{j+1})$，将 $(i,a_i)$ 画在 `xOy` 上。



首先应当 $a_j\ge a_i,a_{j+1}\ge a_{i+1}$，$a_{i+2\cdots j-1}$ 中 $<a_i$ 的部分必须与 $a_{i+1}$ 同色，$>a_{j+1}$ 的部分必须与 $a_j$ 同色，其余两种颜色均可。

一种免去复杂细节讨论的实现见 `std`，其维护上一个下降 $(a_x,a_{x+1})$ 的 $x$ 以及上一个 $<a_x$ 的 $y$，两个下降的两种匹配方式在 `for` 循环里顺便乘上去了。时间复杂度 $\mathrm{O}(n)$，只需一个长为 $n$ 的 `int` 数组。

# T2

## 结论

题意可以视作一类无向图上点权均不同的移棋子博弈在步数足够大时的情况，这类问题的结论是：

先手必胜 $\Leftrightarrow$ 将无向图的边定向为从小到大，然后将得到的 `DAG` 视作博弈图，`SG` 不为 $0$。

定义极大状态，为不存在棋子能移动使新落点比原落点更大。

证：

$\Leftarrow$：在原图中使用与 `DAG` 相同的策略，如果未足够快地到达极大状态，说明后手进行了足够多的“向小的走”的操作，先手可以原路返回，这会直接使先手得分与后手拉开足够大的差距；否则，应当是先手移一步后到达极大状态，接下来先手就可以与后手拉开足够大的差距。

$\Rightarrow$：如果原图中先手存在一个必胜策略，那么考虑这样一个后手策略：如果先手往小的走，则后手原路返回；否则找到所有可以往大的走的棋子选一个往大的走；否则就是极大状态，后手随便走。这样可以得到一棵决策树，其中分叉为后手在第二种情况中选不同的棋子和边。这样就把先手的有效策略给“套出来”了。注意到这个决策树任意往下走都会足够快地出现先手移一步后到达极大状态这种情况。因为如果不足够快，那么说明先手进行了足够多的“向小的走”的操作，这会直接使后手得分与先手拉开足够大的差距；如果足够快但不是先手移一步后到达极大状态而是后手移一步后到达极大状态，那么接下来后手就可以与先手拉开足够大的差距。

该证明的直觉是“如果不到极大状态，那么其中一方就会往小的走足够多步，就输了”。

该模型改编自 `CF1658E`。值得注意的是，网上题解的证明基本都是不严谨的，因为先手往小的走，后手再走回来，这可以使先手耗费掉一步，这在步数不足够大时是有可能逆转局势的。

对该题的证明如下（`from linrui`）：最大值处肯定是先手必胜，所有可以一步到达最大值的一定是后手必胜，将这些点去除。剩下的点中最大值，后手不能移到已知后手必胜的点，否则先手就可以一步移到最大值从而必胜。因此同理这个点也是先手必胜，如此归纳，从而证明原图与 `DAG` 的等价性。注意这里的“必胜”指在足够多轮后一方比另一方多足够多分。这个证明也可以推广到本题，只需把先后手必胜互换即可。

注意，如果点权允许相同，那么先手可以通过反复移权值相同的相邻一度点，将问题归约到轮数不足够大的情况，此时我们只知道一个暴力 `dp` 做法。

## 算法 1

直接枚举删边，算 `SG` 值。注意到如果没有棋子可移，`SG` 必为 $0$，故不必担心必须有棋子可移的限制。

性质 `A` 的树形 `dp` 只用记胜负态。

## 算法 2

定根。

对于 $fa_u < u$ 的点，要求 $f_{u,i,j}$ 表示 $u$ 子树内任意删边，$sg_u = i$，$\bigoplus_{v\in \mathrm{subtree}_u}[2\nmid c_v]sg_v = j$ 的方案数。

对于 $fa_u > u$ 的点，要求 $f_{u,i,j}$ 表示 $u$ 子树内任意删边，$sg_{fa_u} = i$，$\bigoplus_{v\in \mathrm{subtree}_u}[2\nmid c_v]sg_v = j$ 的方案数。

所有转移合并中 $j$ 一维是 `xor` 卷积形式，直接维护 `DFT` 后的值，就可以直接点乘。如果 $2\nmid c_u$，改变即为所有 $2\nmid |sg_u\cap j|$ 的 $j$ 对应值乘 $-1$。

考虑所有 $>u$ 的儿子 $V_0$，它们的 $i$ 的 $\mathrm{mex}$ 决定了新的 $i$。然而这个看似是无法快速求的，因为必须记录一个状压表示每个 `SG` 值是否出现过，或者用对应的容斥（即 `or` 卷积）。

记 $m_u$ 为不考虑父亲时，$sg_u$ 的最大可能值。将 $V_0$ 按 $m$ 从小到大排序，再暴力状压，只转移可能达到的状态。这样的时间复杂度是对的。

证：首先，$m_u \ge i$ 的 $u$ 数量 $c_i \le n/2^{i-1}$。这是因为，一个 $m_u \ge i$ 的点至少要有一个 $m_v \ge 0$ 的儿子，另一个 $m_v \ge 1$ 的儿子，……，另一个 $m_v \ge i-1$ 的儿子，即 $c_i \le \sum_{j=0}^{i-1} c_j$，即 $c_i \le c_0/2^{i-1} = n/2^{i-1}$。

而点 $u$ 会贡献 $(m_u+1)2^{m_u+1}$ 的计算次数，共 $\sum_i c_i \left[(i+1)2^{i+1} - i2^i\right] = \mathrm{O}(n \log^2 n)$。

考虑所有 $< u$ 的儿子 $V_1$，只需在枚举 $u$ 的 $\mathrm{mex}$ 时将它们乘起来即可。精细实现的话这部分可以做到除了内层点乘以外 $\mathrm{O}(n)$。

若 $fa_u > u$，不需要同时枚举 $i$ 和 $sg_u$ 导致多一个 $\log$。对于一个状压状态 $s$，只有 $i = \mathrm{mex}(s)$ 时 $sg_u$ 会到下一个未出现的值，其余情况都相当于没事，因此 $i = \mathrm{mex}(s)$ 处先减一下即可，另外 $i$ 只需记到 $m_u + 1$ 即可，$i = m_u + 1$ 的情况顺便可视作是断边。总之这部分也不是瓶颈。

综上，时间复杂度 $\mathrm{O}(n \log^3 n)$。只要瓶颈部分不炸，其他地方实现差一点都是不影响的。

空间复杂度 $\mathrm{O}(n \log n)$，因为有效 $(u, i)$ 对只有 $\mathrm{O}(n)$ 个（也可以从总儿子数 $= n-1$ 来理解）。

## 前景

出题人怀疑可以通过缩小部分点 `DFT` 数组范围做到两个 $\log$，但限于水平并没有想到具体的方法。

如果您了解这类博弈问题的推广结论请赐教。

# T3

~~这题其实是 `CSP` 之前想到的，然后出题人当时也不会做，就去查了一下 OEIS。~~

## 结论

首先可以消连续的两个或三个相同的等价于可以消连续任意 $>1$ 个相同的。

本题的模型叫做 `same game`。结论详细证明见论文，这里简述如下：

1. 将串首尾相接，不改变可消除性。这是因为，如果一个串 $s$ 可消除，那么将其末尾一位放到开头得到的 $s'$ 也可消除。这是因为，显然可以将 $s$ 的消除过程中消除头尾的步骤放到最后。那么在 $s'$ 中同样做 $s$ 的消除过程中除头尾部分的操作，然后剩下先消 $s'_2$ 所在连续段即可。因此如果某个首尾相接的串可消除，那么取出其最后一步之前的任意一个剩余位置后的空隙断开即可。
2. 将 $s$ 接成环后拿出相邻异或和 $t$ 也组成一个环，设其中 $1$ 的个数为 $c$（显然为偶数），最长连续 $1$ 段长 $m$，那么 $s$ 可消除当且仅当 $m \le c/2$。这是因为可以把单步操作视作将连续一段 $0$ 删去，然后将两侧两个 $1$ 并成一个 $0$。于是考虑所有极长连续 $1$ 段的长度 $l_1, \cdots, l_k$ 形成环，那么每次操作相当于把相邻两个 $l$ 同时减去 $1$，减到 $0$ 就扔掉。这样结论就显然了。

## 算法 1

状压 `dp` 可通过子任务 $1, 2$。

暴力模拟结论是 $\mathrm{O}(n^3)$ 的，可通过前 $4$ 个子任务。

枚举左端点，扫描右端点，维护当前最长连续交替段，可做到 $\mathrm{O}(n^2)$，可通过前 $6$ 个子任务。

## 算法 2

子任务 $7, 8$ 留给不同常数的 $\mathrm{O}(n \log n)$ 做法。

现在讲一个严格线性的做法。分类讨论不可消除子串的情况。令 $a_i$ 为第 $i$ 位，$d_i = a_i \oplus a_{i+1}$，$s_i = \sum_{j=1}^{i} d_i$，$l_i, r_i$ 分别为 $d_i$ 向左/右连续 $1$ 的数量，区间为 $a_{x \cdots y}$。

1. 其绝对众数用上了头尾相异。要求 $2(r_x + l_{y-1} + 1) > s_{y-1} - s_{x-1} + 1$ 且 $a_x \ne a_y$。这是一个二维偏序，其中一维是 $x \le y$。但 $x > y$ 时只要 $a_x \ne a_y$ 就一定会被计入，故可转化为一维偏序。
2. 其绝对众数对应的 $d$ 的 $1$ 是一段前缀或后缀，但没用上头尾相异。以前缀为例，枚举 $x$ 满足 $d_x = 1$，要求 $2r_x > s_{y-1} - s_{x-1}$ 且 $s_{y-1} \ge s_{x-1}$ 且 $a_x = a_y$，这是一个一维偏序。别忘了补回减了两次的全 $1$ 情况。
3. 其绝对众数对应的 $d$ 的 $1$ 是一段中间的部分。这时这段在 $d$ 中一定是个极长连续 $1$ 段，枚举它设为 $a_{x' \cdots y'}$，分头尾异同讨论汇总得要求 $s_{y-1} - s_{x-1} \le 2(y' - x') - 2$ 且 $x < x'$ 及 $y > y'$。这是一个卷积形式，但由于 $y' - x'$ 总长度有限制，故可以枚举 $s_{x-1}$。注意如果枚举 $x$ 是可以卡到带 $\log n$ 的。