Solution

顶峰远眺(mountain)

> powered by CJ

凑数签到题，单调栈求出每个点左右比他高的地方即可。稍微注意一下重复元素的处理即可。

游戏(game)

做法 1

暴力记下每个点的状态，上博弈图直接暴搜，时间复杂度为 $O(2^n n)$，期望得分 20 分。

做法 2

观察到如果存在连续 k 个相同颜色的区域，那么博弈图中一个点会连向他自己。也就是说这个点是一个不败点(如果离开这个状态到达的所有状态都是先手必胜，我可以在原地等着)。

而由于无论进行怎样的操作，一次操作之后都必然存在这样连续的 k 个颜色，因此，如果第一步 Alice 无法直接获胜，他就不可能获胜了。

首先判掉一步就能赢的情况，也即黑色/白色全部在 k 个内出现。

可以通过 Bob 不会获胜的部分分，复杂度 $O(n)$，另外获得 15 分。

做法 3

枚举 $Alice$ 第一次的操作，类似地，如果 Bob 第一次操作不能获胜，那 么 Bob 就永远获胜不了了，这种情况就是平局，复杂度 $O(n^2)$，配合做法 2 获得 80 分。

做法 4

事实上我们可以直接暴力优化做法 3，如果某一种颜色出现位置的 $\max - \min + 1 \leq k$ 就可以一步获胜，直接维护前后缀的最大最小值，枚举操作之后利用前后缀最小值判断两种颜色的 $\max - \min$ 即可。 复杂度 $O(n)$。

做法 5

本质上，做法四就是在判断是否两种颜色的 $\max - \min + 1 \leq k + 1$，如果是的话 $Alice$ 不能一步获胜的情况下必然会使另一种颜色变短，变成 $\leq k$，此时 Bob 获胜。

依次判断即可，复杂度仍为 $O(n)$

计算几何 (imp)

假设最终选的正方形左下角坐标为 (l,l)，右上角坐标分别为(r,r)。容易发现，一个点 (x,y) 能在正方形内，当且仅当 $l\le\min(x,y),\max(x,y)\le r$。

因此我们可以把题目变成：给定 n 个小区间，每个小区间区间为 $[\min(x_i,y_i),\max(x_i,y_i)]$，价值为 c_i，你需要选择一个大区间 $[l,r]$，使得完全包含的小区间的价值之和减去区间长度最大。

容易发现，除了 $l=r$ 的情况，l 一定和某个小区间的端点相同，不然微调一下显然更优。

对左端点扫描线，从大往小加，那么问题变成了：给定 l，你需要选择一个 r，使得右端点 $\le r$ 的小区间权值之和 $-(r-l)$ 最大。然而 l 固定，实际上就是右端点 $\le r$ 的小区间权值之和减 r 最大。

考虑线段树维护，位置 p 的点初始值为 $-p$，每次加一个小区间 (l_i,r_i,c_i) 相当于 在线段树上对后缀 $[r_i,n]$ 做一个区间加 c_i。选 r 的时候就变成了询问最大值。

注意特判 $l=r$ 的情况，此时答案为 0。需要离散化，时间复杂度为 $O(n\log n)$。

树上连通问题(tree)

连通块数量 = 点数 - 边数。

统计连通块数量，等价于统计点数和边数。

$k=1$

因为 $k=1$ ，所以点数和边数我们可以分开统计，点数直接求和，对于一条边 (u,v)，我们希望得到有多少个区间包含它。

假设 $u<v$，容易发现方案数就是 $u(n-v+1)$，可以直接算。 时间复杂度 $\mathcal{O}(n)$。

$k=2$

求连通块数量的平方和。

我们考虑枚举左端点，并且维护所有右端点的答案。（当然你从左往右也可以）

当左端点由 $i+1$ 移动到 i 时：

① 所有右端点对应区间的点数 $+1$，因此连通块数量均 $+1$。

② 对于一条边 $(i,j),i<j$，当右端点在 j 右面时，包含的边数 $+1$，因此连通块数量 -1。

我们发现，只需要支持区间加和整体求平方和。可以用线段树轻松实现。

时间复杂度 $\mathcal{O}(n\log n)$。

喀克托斯（cactus）

1. 树的拓扑序计数

树的拓扑序计数是简单的，只需要设 f_i 表示以 i 为根节点的子树的拓扑序数量，转移时乘上重标号的系数即可：

$$f'_u = f_u \times f_v \times \binom{siz_u + siz_v}{siz_v}$$

2. 点仙人掌的拓扑序计数

对于边仙人掌，先把点双缩点，然后把相邻（即有公共点）的点双之间连边，就可以得到一颗有根树。

但是次数环上的转移有限制，而环外的和树的情形相同。因此设 f_u 表示以 u 为根 **遍历 u 所在环外的点** 的拓扑序个数；g_u 表示以 u 为根遍历 u 子树的拓扑序个数，其中要求 u 是所在环的最高点。

f 可以从 g 转移：$f'_u = f_u \times g_v \times \binom{siz_u+siz_v}{siz_v}$。

g 的转移还需要一个 DP。u 所在的环可以分成左右两边，每边内部需要按顺序访问。可以设 $h_{i,j}$ 表示从左边第 i 个，右边第 j 可开始遍历的拓扑序个数。转移的时候从左边或者右边前面添加一个点，并钦点该点是第一个访问的，转移到 $h_{i-1,j}$ 和 $h_{i,j-1}$。最后 $g_u = h_{0,0}$。

答案即为 g_r。时间复杂度 $O(n^2)$。