

贪心

黄洛天

THU, IIIS

April 27, 2025

一些要讲的

- 反悔贪心
- 贪心杂题
- 构造杂题

现在有 n 个怪物。攻击一个怪物会先消耗 a_i 的血量，再回复 b_i 的血量。

你想知道你初始最少多少血，才能打完所有怪。

但是这个问题太经典了，于是现在有 m 天，每天第 i 个怪物的 a_i 会增加 a'_i ， b_i 会增加 b'_i 。你需要求出这 m 天，每天把所有怪打完的初始血量之和。

$$n \leq 100, m \leq 10^{18}。$$

GYM104651 I - Solution

首先考虑一天咋做，这是个经典问题。

首先打 $b_i > a_i$ 的怪，这些怪按照 a_i 从小到大排序，然后打 $b_i \leq a_i$ 的怪，这些怪按照 b_i 从大到小排序。

由于 n 很小，考虑处理出每种可能的排序方式。也就是把 m 天按照排序的结果分成若干段，显然最多有 $O(n^2)$ 段。

对于每一段，打完前 i 个怪物的血量可以写成 $kx + b$ 的形式，其中 x 是天数。

于是求个半平面交 / 凸包就可以得到这段时间的贡献了。

时间复杂度 $O(n^4)$ 。

loj 3988

<https://loj.ac/p/3988>

loj 4220

<https://loj.ac/p/4220>

小练习 1

有一个二分图，你需要找一组边的匹配。一对边是好的，当且仅当两条边共点，你要最大化好的匹配对数。

小练习 2

有一个二分图，你需要找一组边的匹配。一对边是好的，当且仅当两条边共点，你要最小化好的匹配对数。

P6631

<https://www.luogu.com.cn/problem/P6631>

uoj 455

<https://uoj.ac/contest/47/problem/455>

给一个无向图，要求给边定向成 dag，再给出两个拓扑序 P_i, Q_i ，最大化 $\sum_{i=1}^n [P_i \neq Q_i]$ 。 $n, m \leq 5 \times 10^5$

CF1477D - Solution

注意到一个点集 S 的导出子图如果不连通，那么可以无代价的删掉 S 。令 V_1, V_2 分别为 S 到导出子图的两个连通快，按照随意的方式定向，得到两个拓扑序 P_1, P_2 。 A 中放 P_1, P_2 ， B 中放 P_2, P_1 。然后把 S 到 $V - S$ 的边全部设置为 S 到 $V - S$ ，这样就把问题转化到了更小的规模。

接下来注意到每次选取的 V_1, V_2 ，一定可以满足 $\min(|V_1|, |V_2|) = 1$ ，因为选 $u \in V_1, v \in V_2$ ，变成 $(u, V_2 - \{v\}), (V_1 - \{u\}, v)$ 也行。

CF1477D - Solution

答案为 $n - \sum_{i=1}^n [deg_i == n - 1]$, 这个下届是好证明的, 考虑构造。

先把 $deg = n - 1$ 的点全部扔开头。

对于剩下的图我们唯一有的性质叫, 每个点都有至少一个点没连边。因此我们不妨设 P_i 为 i 任意一个与 i 没有边的点。

然后可以很自然的建在 i 到 P_i 连边, 于是会得到一个基环树。

下文称选了一组 V_1, V_2 表示对这些点染色, 不妨设 $|V_1| = 1$, 那么称 V_1 里的元素为根 (就是菊花的根, 很形象!)。

CF1477D - Solution

对于树的可以直接 dfs 从下往上构造，如果 u 的至少一个儿子还没被染色，那么以 u 为根染色。

考虑断开环上一个边。最后如果根剩下了单独的，那么把根归类的到任意一个儿子的 V_2 里即可。

时间复杂度 $O(n + m)$ 。

CF1290D

给定正整数 n, k 。有 n 个数 a_1, a_2, \dots, a_n 初始未知，有一队列 Q 初始为空。

每次查询你可以给出一个编号 x ，交互系统会依次进行如下操作：

- ① 告诉你当前 Q 中是否有与 a_x 相同的数
- ② 将 a_x 放入 Q 的尾部
- ③ 如果此时 Q 中元素多于 k 个，弹出队首元素

此外，你还可以重置队列，这将使得 Q 中所有元素被弹出。

现请你求出 a_1, a_2, \dots, a_n 中有多少个不同的数。要求你的查询次数不超过 $\frac{n^2}{k}$ ，重置次数不超过 30000，保证 n, k 是 2 的整数次幂。

CF1290D - Solution

首先很想把 $\frac{k+1}{2}$ 个东西分成一组，称组的大小为 S ，假设会被分成 m 组。

对于任意两组都要考虑他们之间的贡献，也就是判断有没有重复数字。

如果单独考虑第 i 组和第 j 组的贡献的化。方法是先把第 i 组的都插进去，再把第 j 组的都插进去，如果插进去的时候已经有重复了，那么这个点就没救了。

我们希望对于每个字符都只保留一个点有救。这意味着如果一个点已经没救了，那么他不能让别人也没救，否则会出现一个字符算死了的情况。

这样直接考虑 $\frac{m*(m-1)}{2}$ 对贡献，每个花费 k 的代价，总操作次数为 $\frac{2n^2}{k}$ 。

CF1290D - Solution

每次都从一个空栈开始是很浪费的。如果考虑 (i, j) 贡献后立刻考虑 (j, k) 贡献，那么第二次只需要 S 的代价。

于是考虑把需要计算的贡献看成一个完全图，把 m 个点的完全图拆成 $\frac{m}{2}$ 个哈密顿路，每个哈密顿路的代价是 mS ，总代价就是 $\frac{n^2}{k}$ 。

log 3805

<https://loj.ac/p/3835>