水题选讲

彭博

北京大学

2024.7

https://qoj.ac/contest/1475/problem/8008

CF741C Arpa's overnight party and Mehrdad's silent entering

有 n 对情侣围成一圈坐在桌子边上，食物有两种，要求情侣不能吃同一种食物，并且桌子上相邻的三个人的食物必须有两个人是不同的。

构造一种可行分配方式。

$n \leq 10^5$

CF741C Arpa's overnight party and Mehrdad's silent entering

把限制加强到“$2i$ 和 $2i+1$ 的食物不能相同”。

那么两个匹配的并必然是二分图，所以直接染色即可。

CF1656G Cycle Palindrome

给定一个长度为 n 的数列 a ，你需要找到一个长度为 n 的置换 p ，使得 p 作用在 a 上的结果是回文串，且 p 只由一个环组成。或者判断无解。

$n \le 2 \times 10^5$

p 是一个环？这什么奇怪限制。

先不管它，先随便取一个 p 使得作用在 a 上是回文串。

现在 p 有很多小环，考虑怎么把它们合并成一个大环。

“回文串”的限制是很松的：i 和 $n-i+1$ 可以交换，并且 $i, j, n-i+1, n-j+1$ 这两对也可以同时交换。

随便换换就变成一个大环了。

https://codeforces.com/problemset/problem/1844/E

1st ucup stage 14 C LaLa and Lamp

有 $n(n+1)/2$ 个点排成一个正三角形，每个点是黑色或者白色。

你每次可以选择横着的一行，或者向左 $60°$ 的一行，或者向右 $60°$ 的一行，把这一行上的所有点的颜色取反。

判断能否把所有点的颜色都变成白色。

$n \leq 2000$

可能的操作数只有 $3n$ 种，但限制有 $O(n^2)$ 个，因此只需要很小一部分的限制就足以确定唯一解，然后再判断合法。

发现最下面两行非常不错：只要枚举两行是否操作，然后就有恰好 $2n-1$ 个限制和 $2n$ 个变量，每个限制恰好连接两个变量。

然后模拟就好了。

有一个 $n \times m$ 的地图，每个位置上有一种颜色。另外给了 k 种模型，第 i 种模型是一个 $n_i \times m_i$ 的矩形，同样是每个位置上有一种颜色。

你有两种操作。第一种操作是把地图上相邻两个位置的颜色交换，第二种操作是选择一个模型，把地图左上角 $n_i \times m_i$ 的子矩形的颜色变为模型对应位置的颜色。

构造一个从初始地图到目标地图的操作序列，或判断无解。操作序列有长度限制，轻微卡常。

$n, m, k \leq 20$

倒着做。从目标地图开始，当经过调整之后左上角 $n_i \times m_i$ 的子矩形的颜色与某个模型匹配时，就可以把这个子矩形的颜色全都变成通配符。

无脑贪心就完事了。

PKU-CPC 2023 C Empty up a Bottle

有三个容量无限大的瓶子，初始分别装了 A, B, C 的水。

你可以进行若干次操作，每次选择两个瓶子，从水较多的瓶子往水较少的瓶子里倒水，使得水较少的瓶子里的水量翻倍。

你的目标是把任意一个瓶子清空。

允许将操作合并：只要选择两个瓶子之后操作就唯一确定了（除了水量相等的情况，但此时就做完了，所以无所谓）。你可以选定两个瓶子，然后再选定一个正整数 k，对这两个瓶子连续进行 k 次操作。称这样为一组操作。你需要在 200 组操作内完成目标。

$A, B, C \le 3 \times 10^{13}$

对水量为 A, B 的两个瓶子操作，会得到 $(2A) \bmod (A+B), (2B) \bmod (A+B)$ 。因此操作 k 次就等价于模意义下乘 2^k 。

由于可以一次进行一组操作，所以只要 $A+B$ 是奇数，能乘 2^k 就也可以除 2^k 。

不妨假设 A, B 是偶数，C 是奇数。那就可以 $(A, B, C) \to (2A, B-A, C) \to (A, B-A, C+A)$ ，用三次操作实现 $B \to B-A$ ，而仍然保证 C 是奇数。这就是一个辗转相减求 gcd 的过程，只需要在 $B \gg A$ 时一次减掉 $2^k A$ ，就可以在 $O(\log(A+B))$ 轮内完成。

CF1693F I Might Be Wrong

给定一个长度为 n 的 01 串 s ，你每次可以任意选取一个区间 $[l,r]$ ，付出 $|cnt_0 - cnt_1| + 1$ 的代价把这个区间排序，其中 cnt_0, cnt_1 分别表示区间中 0 和 1 的个数。

求出把整个 01 串排序所需的最小代价。

$n \le 2 \times 10^5$

CF1693F I Might Be Wrong

手造几个 1111100 的样例玩玩，发现好像总是可以使得每次操作都 01 个数相等。

这并不难证：不妨假设 $cnt_0 > cnt_1$ 且 $s_l = 1$ ，那么一定存在一个前缀 $[l, r']$ 使得 01 个数相等，那么操作 $[l, r]$ 的结果和代价与先操作 $[l, r']$ 再操作 $[l+1, r]$ 的结果和代价相同。继续调整即可。

假设整个 s 里 0 比 1 多。

考虑把最靠左的 1 往右推，发现每次贪心推就好了：从第一个 1 开始找一个尽量长的区间使得 01 相等，贪心操作，直到推到底为止。

可以发现贪心操作非常优秀：任何多余的操作只会使得往后推的过程更加困难，不如不做。

给定 n 个两两不同的球，第 i 个球上写了数字 a_i 。你要把它们放进 k 个两两不同的盒子里，使得每个盒子非空。最大化每个盒子的 AND 之和，并计算有多少种最大化的方案。

$n, k \le 10^5, a_i \le 10^9$

感性理解一下，把大数字单独放在一个盒子里，好像总不是太亏。

假设所有数字里最高位是 2^b ，且只有不超过 $k-1$ 个数字有 2^b 。可以发现把它们每个单独放一个盒子一定是严格最优解：假设 x_1, y 一个盒子，x_2 一个盒子，且 $x_1, x_2 < 2^b \le y$ 。由于 $x_1 + x_2 = x_1 \text{ OR } x_2 + x_1 \text{ AND } x_2$ ，把 x_1, x_2 放一起只会亏掉 $x_1 \text{ OR } x_2$ ，严格小于 y 。其他情况也可以用类似方式证明。

而如果超过 $k-1$ 个数字有 2^b ，同样可以证明 $< 2^b$ 的数字必然处于同一个盒子内。此时直接把它们合并成一个数字，那么无论怎么分配都会有 $k-1$ 个盒子能贡献 2^b ，于是 b 这一位就可以删掉了。

1st ucup stage 3 F Flower Garden

给定 n, m ，你需要构造一个长度为 $3n$ 的 01 串 s ，满足给出的 m 个限制。每个限制都形如：

- 给定 a_i, b_i, c_i, d_i ，你构造的串 s 需要满足以下两个条件中至少一个：
  - $s_{a_i} = s_{a_i+1} = \cdots = s_{b_i} = 0$
  - $s_{c_i} = s_{c_i+1} = \cdots = s_{d_i} = 1$

另外，你还需要保证 s 至少有 n 个 0 、至少有 n 个 1 。

构造任意一个合法解，或判断无解。

$\sum 3n, \sum m \leq 10^6$

1st ucup stage 3 F Flower Garden

限制是二选一，不难想到 2-sat 。但如果直接建图跑 2-sat ，那么会比较难处理至少 n 个 0 和 n 个 1 的要求。

注意到限制的形式非常统一，因此不需要拆点，可以直接用 $x \to y$ 表示 "如果 $s_x = 1$ ，那么 $s_y = 1$"，于是每个限制变成了一个区间向另一个区间连边。显然可以用线段树优化建图，然后缩点。

此时问题转化为，给定一个有点权的 DAG ，需要把它切成两半，使得每一半的权值和都在 $[n, 2n]$ 内。

如果存在一个大点，其点权至少是 n ，那么把它分到哪边，哪边的权值就一定够用。因此只需要枚举它在左边还是右边，然后让它和它的前驱/后继为一半，剩下的点为另一半即可。

否则，由于 $2n - n = n$ ，因此只需要按拓扑序一个个把点加入左边的集合，那么一定在某个时刻会落入 $[n, 2n]$ 的区间内，也就得到了一个合法方案。

有 n 个玩家，从 1 到 n 编号，按顺序形成一个环。

现在有 n^2 张牌，每张上有一个整数，范围在 $[1, n]$ ，其中值为 i 的牌有 n 张。

每一次操作，令每个玩家选择一张牌给自己的下家。所有这些牌的移动都是同时执行的。

玩家 i 的目标是得到 n 张值为 i 的牌。请找出一种方案：在不超过 $(n^2 - n)$ 次操作使每个玩家达成目标。

你不需要使操作数最小化。

CF1646F Playing Around the Table

目标状态很难：值为 i 的牌都要到玩家 i 处，所以每张牌要去的位置都被固定了。就算某个人已经拿到了所有自己想要的牌，再过一轮他就会被迫丢失一张，这很难受。

考虑一个更简单的题目：每个人都要把值为 $1 \sim n$ 的牌各拿一张。在这个设定下，每张牌要去的位置就不是固定的了，有很多操作的空间。

不难发现，各拿一张要到拿同一张（反过来也一样），只需要 $(n^2 - n)/2$ 次操作。所以问题可以转化为用 $(n^2 - n)/2$ 次操作使得每个人都有值为 $1 \sim n$ 的牌各一张。

（可惜这不是等价转换，所以你只能希望它能做。）

CF1646F Playing Around the Table

现在就有一个非常简单的贪心：在每一次操作中，每个人选一张自己不需要的牌往前传。因为后面会来一张牌，所以每个人做决策时其实有 $n+1$ 张牌，所以必然会有一张是不需要的。

写一发，发现过了，为什么呢？

只考虑值为 1 的牌需要被往前送多少次。把牌看作 1 ，人看作 -1 ，那么必然存在一个循环移位使得前缀和非负。于是不难发现，在这个循环移位下，每张牌只会往前走，不会从第 n 个人回到第 1 个人。

因此走的次数一共最多是 $n(n-1)/2$ 。

有 n 种值，但是一次操作会移动 n 张牌，所以操作次数也最多是 $n(n-1)/2$ 。

给定一个 n 个点的完全图，每条边都被染了 m 种颜色中的一种。

你需要判断能否再加入 $m-n+1$ 个点，补成一个 $m+1$ 个点的完全图，并且每条边仍然被染了 m 种颜色中的一种，使得每个点都满足相邻的边的颜色恰好取遍这 m 种。

$n, m \leq 200$

首先判掉原图就不合法的情况，以及 $2 \nmid (m+1)$ 的情况。

在最终的图里面，每种颜色都会有恰好 $(m+1)/2$ 个匹配，或者说 $(m+1)/2$ 个 set 。

那么现在就可以先把 $1, \cdots, n$ 加入到这些 set 里。按照 set 的大小分为 0-set, 1-set, 2-set 。

显然，如果一种颜色现在的 1-set 和 2-set 个数之和已经超过了 $(m+1)/2$ 那么也无解。

否则，通过构造的方式证明一定有解。

只需要每次加入一个点，并仍然保证每种颜色的 set 个数不超过 $(m+1)/2$ 即可。

不难想到用二分图匹配来构造：左边 m 个点表示 m 种颜色，右边 $n+1$ 个点，前 n 个点表示图中的点，最后一个点表示不连边。左边到源和右边到汇的容量都是 1 ，只有第 $n+1$ 个点到汇的容量是 $m-n$ 。

如果图中第 j 个点旁边没有第 i 种颜色，就从 i 到 j 连一条容量为 1 的边。i,j 匹配的意义就是第 j 个点要向新点连颜色为 i 的边。

另外，每种颜色还要向右边第 $n+1$ 个点连两倍 0-set 个数条边，每条边的容量也都是 1 。(下面会说为什么要连那么多条边而不是只是 0 或 1 条边。)

跑一个二分图匹配。显然，只要满流，就成功地从 n 扩展到了 $n+1$ 。

我们惊奇地发现，左边的每个点的出度，和右边的每个点的入度(除去最后一个点)，都是 $m+1-n$ ！

因此，只要把右边最后一个点拆成 m 个，就得到了一个二分正则图，所以一定存在完美匹配。