构造型问题

任飞宇

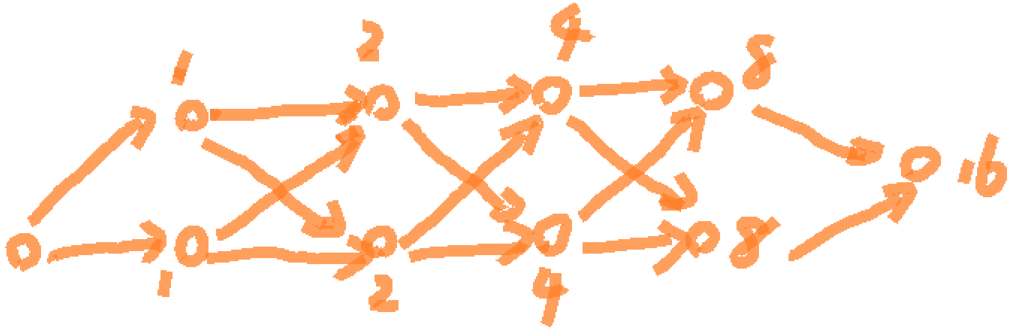
1. Fox and Minimal path

构造一张不超过1000个点的，边权均为1的无重边无自环有向图，使得点1到点k（k为你构造的图的点数）的最短路径条数恰好为n
数据范围：$1 <= n <= 10$^9

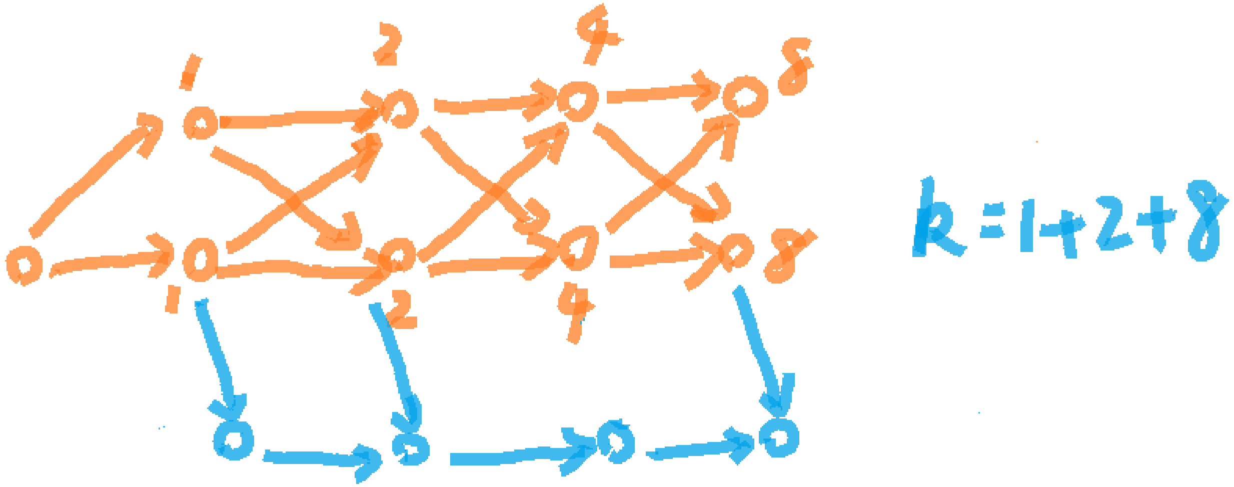
1. Fox and Minimal path

二进制是经常使用的思维。

不妨先构造一个有2^0条最短路的方案。



考虑再拉一条链，用来构造多个2的幂之和：



2. Kuroni and the Celebration

这是一道交互题。

给你一棵有 n 个节点的树（以无根树的形式）。这颗树有一个隐藏的根节点，你可以通过若干次询问来进行猜测。对于每次询问，提交两个点，评测机会返回这两个点的 LCA。

询问格式为 `? u v`，此时评测机会返回 u 和 v 的 LCA。

提交格式为 `! x`，表示你得出树根为点 x。

你可以最多询问 $\lfloor \frac{n}{2} \rfloor$ 次。

$2 \le n \le 1000$

2. Kuroni and the Celebration

这是一道交互题。

给你一棵有 n 个节点的树（以无根树的形式）。这颗树有一个隐藏的根节点，你可以通过若干次询问来进行猜测。对于每次询问，提交两个点，评测机会返回这两个点的 LCA。

询问格式为 `? u v`，此时评测机会返回 u 和 v 的 LCA。

提交格式为 `! x`，表示你得出树根为点 x。

你可以最多询问 $\lfloor \frac{n}{2} \rfloor$ 次。

$$2 \le n \le 1000$$

不妨先找两个叶节点a，b，如果得到LCA就是a或b，那么根就是LCA，否则可以得出根一定不是a和b。

允许询问的次数为$\lfloor \frac{n}{2} \rfloor$，正好每次删去两个叶节点。

3. Linguistics

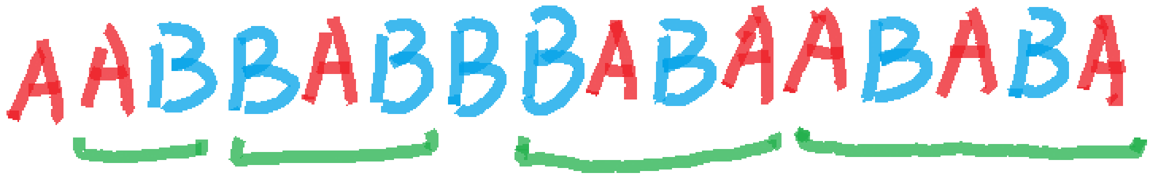
现有4种碎片A、B、AB、BA，数量分别为a、b、c、d，和一个字符串S，你可以以任意顺序拼接这些碎片，问能否拼成S？

$1 \le |s| \le 2 \cdot 10^5$

3. Linguistics

直接dp的复杂度过高。
考虑减少问题的变量。如果确定了AB和BA的位置，那么把剩下的A和B填进去就行了。

那么AB和BA的可选位置的特征是什么呢？



位于所有长度大于1的段之间，A与B交替的绿色区域，这些区域互不影响。

1. 长为2k+1的区域可以放下总量k的AB和BA；
2. 长为2k以A打头的区域如果全放AB，能放k个，如果AB和BA皆有，共能放k-1个
3. 长为2k以B打头的区域如果全放BA，能放k个，如果AB和BA皆有，共能放k-1个

贪心地想，肯定优先把AB放第二类，BA放第三类，剩下的应放尽放即可。

4. Inversion SwapSort

给定长度为 $n\ (1 \le n \le 1\,000)$ 的数列 $a_1, a_2, \ldots, a_n$，找到其所有逆序对的一个排列 (u_i, v_i)，使得依次交换 u_1 和 v_1 位置上的数、u_2 和 v_2 位置上的数、……最后得到的数列不减。

4. Inversion SwapSort

不妨先考虑数列是排列的情况。

由于需要交换的位置是定死的，不好操作。
考虑反转排列的下标和值，问题就变为交换所有逆序对的值(而不是位置)，使结果有序。

使用冒泡排序，每次交换相邻两个数，就可以保证每个逆序对都被交换恰好一次。

5. Kuroni and the Score Distribution

构造一个序列 a 满足如下:

1、长度为 n 。

2、$\forall i, 1 \le a_i \le 10^9$ 。

3、$\forall i > 1, a_i > a_{i-1}$。

4、满足 $i < j < k$ 且 $a_i + a_j = a_k$ 的三元组 (i, j, k) 数量恰好为 m。

$n \le 5000, m \le 10^9$ 。

5. Kuroni and the Score Distribution

如果m太大的话会导致无解。不妨先考虑什么情况可以得到最多的三元组数量。

当$a_i = i$时每个数都能在前面找到尽可能多的数对，此时三元组数量最多，达到

$$N = \left\lfloor \frac{0}{2} \right\rfloor + \left\lfloor \frac{1}{2} \right\rfloor + \cdots + \left\lfloor \frac{n-1}{2} \right\rfloor$$

考虑先令前k个数为$a_i = i$，使得三元组数量不超过m，此时已有$\left\lfloor \frac{0}{2} \right\rfloor + \left\lfloor \frac{1}{2} \right\rfloor + \cdots + \left\lfloor \frac{k-1}{2} \right\rfloor$数量的三元组，还需要不超过$\left\lfloor \frac{k}{2} \right\rfloor$个。

如果$a_{k+1} = k + 1$，能产生$\left\lfloor \frac{k}{2} \right\rfloor$个，为了让$a_{k+1}$产生的数量减少，我们可以增大$a_{k+1}$的值，使得$a_{k+1}$产生的三元组数量符合要求。

剩下的数不能产生三元组，考虑把它们全部放到后面，且间隔足够大即可。

6. Prefix Product Sequence

构造一个长度为n的，1~n的排列，使得这个排列的前i项积为n的一个完全剩余系，或者说明无解　　1<=n<=100000，

6. Prefix Product Sequence

首先考虑什么时候有解，答案是当n为4或者是质数的时候，之所以4有解是因为，4只能被分解成$4=2\cdot 2$,其他的合数都可以被分解成两个比他小的因数，那么这样在mod n意义下就是0了，所以不可能。

1不能放在除了第一个地方以外的其他地方，因为放在其他的地方就会出现前缀积相等的情况，n只能放在最后，因为前缀积加入n以后在mod n意义下就是0了。结合威尔逊定理也可以得出无解的情况。

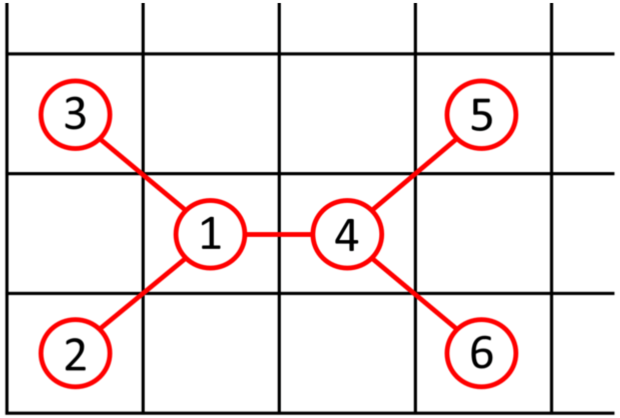
中间的数放$\frac{2}{1},\frac{3}{2},\frac{4}{3},\dots$，这些数是两两不同的，因为它们减一后是两两不同的。

另一个中规中矩的想法：凡质数必有原根。把1~n-1表示为$g^0,g^1,\dots,g^{n-2}$，问题变为排放0~n-2使得前缀和模n-1两两不同。容易想到从0出发向前1步，向后2步，向前3步。。。

	0	1	4	3	2	5
s	0	1	5	2	4	3

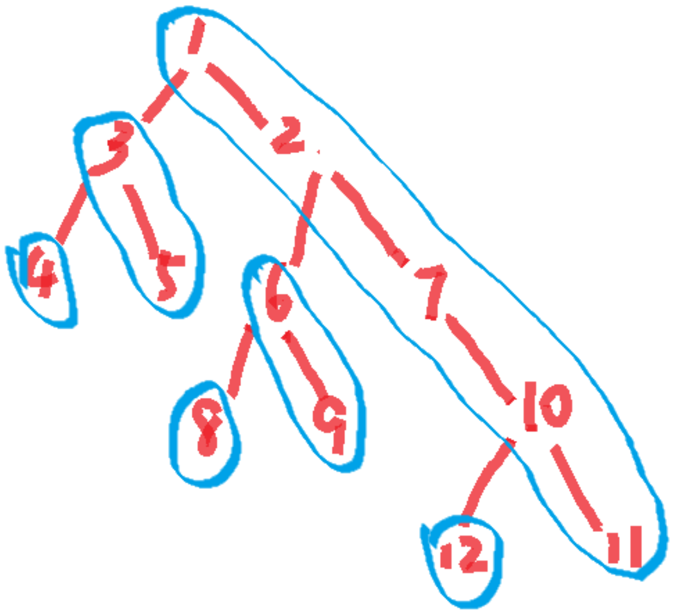
7. Carpet

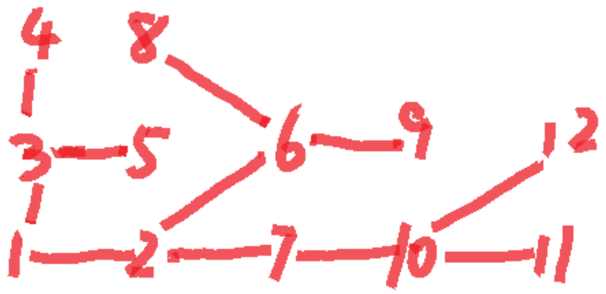
给定一个n(1<=n<=1e5)个结点的树，现在需要你把这棵树放在一个x*y的矩形中(1<=x<=1e6,1<=y<=20)，结点要放在整数坐标上，且没有边交叉。输出树上的每个结点在矩形中的坐标。



7. Carpet

利用树剖的性质，根节点到每个节点经过的轻链总数不超过log（n），因此可以把重链放在同一行上。





8. Ehab and the Big Finale

这是一道交互题。

你有一棵n个节点的有根树，1号点是根节点。

这棵树中有一个隐藏的节点x，你需要通过询问把x找出来。

你可以进行如下两种询问：

1、$d\ u\ (1 \le u \le n)$。交互库会返回节点u和x的距离。

2、$s\ u\ (1 \le u \le n)$。交互库会返回从u到x的路径上第二个点的标号。注意，你询问的u必须是x的祖先，否则会Wrong Ans。

你需要在不超过36次询问之内找出x。x是预先设定好的，不会随着询问而改变。

$2 \le n \le 2 \cdot 10^5$

8. Ehab and the Big Finale

利用2操作可以知道x在祖先u的哪个子树下面。
但是一步步爬太慢了，需要利用1操作加速。

利用1操作可以知道x到某点的距离，不过单个信息似乎得不到什么结果。

考虑1操作得到x到u和v的距离，我们可以据此得到x在u-v链的哪个点的子树上。

令u为根，v为根所在重链的叶节点，就可以得到x和v的LCA。

容易判断x是否在当前重链上(即x=LCA)，如果不在，就在某个轻儿子的子树里。

LCA可能有多个轻儿子，可以用2操作找是哪个轻儿子，然后进入下一条重链，
v换成下一条重链的底部叶节点，继续找。

9. Tokitsukaze and Permutations

有一个长度为n的排列p，将执行k次操作

操作过程：枚举i从1到n，当$p_i > p_{i+1}$，则交换p_i，p_{i+1}

经过k次操作之后，得到了一个新数组a，再定义数组v表示在1~$i-1$中比a_i大的个数

现在给定v，但是有可能其中的值为-1，这表示它的值并不确定

求有多少种p满足在k次操作后得到的v和给定确定值一致，结果取模998244353

第一行两个正整数$n(1 \le n \le 10^6), k(0 \le k \le n-1)$，表示长度，操作次数

第二行包含n个整数$v_i(-1 \le v_i \le i-1)$，当$v_i = -1$时表示该值不确定

9. Tokitsukaze and Permutations

排列p与序列v存在双射关系，于是可以考虑对v计数。

分析一次冒泡对v的影响。

如果这趟冒泡没有交换位置i和i+1，说明冒泡前$\max\limits_{1\le j\le i} p_j < p_{i+1}$，$v_{i+1} = 0$，

冒泡会把$\max\limits_{1\le j\le i} p_j$提到位置i，所以冒泡后$v_i' = 0$。

如果这趟冒泡有数字从位置i去往i+1了，说明冒泡前$\max\limits_{1\le j\le i} p_j > p_{i+1}$，$v_{i+1} > 0$，

那么冒泡后$v_i' = v_{i+1} - 1$。

所以一次冒泡相当于$v_i' = \max(v_{i+1} - 1, 0)$，说明每个位置的v都是独立的，

根据每个位置的值域，不难统计方案数。

10. Non-coprime DAG

$N(\leq 10^6)$个点的有向图，每个点编号$1\sim N$。边$i -> j$存在当且仅当

1. $i < j$; 2. $\gcd(i,j) > 1$,

称a到b可达，如果点a能直接或间接走到b。

每个点有一个值$A_i(\leq 10^9)$。希望找一个点集S，使得A值之和最大，且满足：

对任意$x, y \in S, x < y$，x不可达y（在原图中）。

10. Non-coprime DAG

$N(\leq 10^6)$个点的有向图，每个点编号$1 \sim N$。边$i -> j$存在当且仅当
1. $i < j$;　　2. $\gcd(i,j) > 1$,
称a到b可达，如果点a能直接或间接走到b。
每个点有一个值$A_i(\leq 10^9)$。希望找一个点集S，使得A值之和最大，且满足：
对任意$x, y \in S, x < y$，x不可达y（在原图中）。

切入点在于根据奇偶性进行分析。
显然偶数从小到大都是有边的，考虑某个奇数x的连边情况，$x -> y$等价于$\gcd(x, y-x) > 1$
那么$y-x$最小能取x的最小质因子$f(x)$，最小的y为$x+f(x)$，且一定是偶数；
同理最大的入边w为$x-f(x)$，且一定是偶数。x在$[x-f(x)+1, x+f(x)-1]$范围内没有边，跟范围外的偶数都有边。

从x到y的策略为，x和y走最近的边到偶数上，然后走偶数之间的边。

定义偶数i的$L_i = i, R_i = i$，奇数i的$L_i = i - f(i) + 1, R_i = i + f(i) - 1$,
x不可达y等价于它们的区间相交。

S里的点两两不可达等价于它们的区间有交。
把点的A值加到区间里的每个位置，所有位置里的最大值就是答案。