Day5讲题+CS Academy选讲

高睿泉

南京外国语学校

2018 年 7 月 5 日

T1-算法1、2

最简单的暴力为枚举不考虑无法通过的边的情况下的路径，再枚举其中一条边去掉后的最短路的最大值。这样可以通过子任务1。

对于子任务4，可以发现只需考虑环上的情况：最坏情况一定是去掉T连的一条边，讨论顺逆时针行走的情况即可。

总期望得分20分。

T1-算法3

设h_x表示到达x后知道相邻的一条边无法通过时后的最短路的最大值。

如果计算出了所有的h_x，可以通过二分答案lim，计算从S到每个点的合法路径的最小值f_i（合法路径表示路径上不存在$f_j + h_j > lim$）。

可以暴力计算所有h_x，复杂度$O(nm + m\log^2 n)$，期望得分50分。

T1-算法4

考虑优化h_x的计算。首先去掉的边一定是x在最短路树上连向父亲的那条边，那么只需计算非树边对最短路的影响: 非树边(u, v, w)，可以产生影响的位置为u到v路径上的点，并且可以发现其对于h_x的影响为$dep_u + dep_v + w - dep_x$，用可并堆维护除去$dep_x$部分的最小即可，对于已经不合法的边，可以不用删除，只当其出现在堆顶的时候判掉即可。

这一部分复杂度为$O(m \log m)$，总复杂度为$O(m \log^2 n)$。

T1-算法5

可以发现算法4的瓶颈在于最后一部分的计算。设g_x为从x开始走到T的答案，有:

$$g_x = \max\{h_x, \min\{g_y + w_{x,y}\}\}$$

可以用计算最短路的方法处理。复杂度$O(m \log m)$，期望得分100分。

T2-算法1、2

可以用线性筛或者众所周知的积性函数求和理论得到27分或68分。

T2-算法3

观察f的性质：$f(n)=(p_1^{a_1}+1)(p_2^{a_2}+1)...(p_k^{a_k}+1)$，可以看成为$n$的$2^k$个质因子幂子集的乘积的和，即将$n$分成两个互质数字的积对第一个数字求和。

那么问题变成求$\sum\limits_{ab\leq n} a[gcd(a,b)=1]$。可以枚举$a,b$的较小值，再通过枚举约数容斥计算。复杂度$O(\sqrt{n}\log n)$，期望得分100分。

T3-算法1、2

直接枚举所有全排列即可。复杂度$O(n^2 \times (n \cdot (n-1)/2)!)$。期望得分10分。

对所有边进行状压dp，$dp[S]$表示选了用了S集合中边，且当前最小生成树合法的方案数。复杂度$O(2^{n\cdot(n-1)/2} \times n^2)$，期望得分30分。

T3-算法3

对于$a_i \le n$的部分，只要考虑这个图的前n条边即可。假设不在最小生成树内的边为m，枚举其所在环的大小s，这样就变成了一个新图的生成树计数问题，可以用matrix tree定理或者prufer编码解决。复杂度$O(n^4)$或$O(n)$，并上算法1、2,期望得分50分。

T3-算法4

仍然考虑状压，为每加完一条边后连通块大小的集合对应的方案数。这个集合的种类数也就是n的划分数$p(n)$，当$n = 40$时$p(n) = 37338$。

按权值从小到大决策，设$dp_{i,S}$表示决策完第i条边后连通块大小集合为S，如果$i+1$在最小生成树中，那么转移时连接两个连通块，否则乘上S算出当前块内没有出现的边数即可。注意到每个S以连接方式转移只会有1次;同时S中不同的数字只会有$\sqrt{n}$种，每次连接时只要枚举连接的两个连通块的大小即可。这样总的复杂度为$O(p(n) \times n^2)$，期望得分100分。

Xor Transform

题目大意

给定$n \times m$的矩阵A，定义矩阵变换$t(B)$:

$$t(B)_{i,j} = B_{i,j} \oplus B_{i+1,j} \oplus B_{i,j+1} \oplus B_{i+1,j+1}$$
$$t^K(B) = t(t^{K-1}(B))$$

其中矩阵外的元素均视为0。Q组询问：求$t^K(A)_{0,0}$。

数据范围

$n \times m \leq 5 \cdot 10^6, Q \leq 5 \cdot 10^7$

Xor Transform

可以发现$A_{i,j}$对$t^K(A)_{0,0}$有贡献当且仅当$K \land i = i, K \land j = j$。

因为(i,j)到达$(0,0)$的方案数为$\binom{K}{i}\binom{K}{j}$，由lucas定理即可推出上述结论。

最后做一遍莫比乌斯变换即可得到答案。复杂度$O(\max\{n,m\}\log\max\{n,m\}+Q)$。

Count BST

题目大意

给定n和序列$a_1, a_2, ..., a_k$，求有多少棵二叉排序树满足存在路径$a_1 \to a_2 \to ... \to a_k$。

数据范围

$n \le 3 \cdot 10^6, 0 \le k \le 2 \cdot 10^5$

Count BST

首先考虑$k = 0, 1$时的情况，$f_n = \sum_{0 \le i < n} f_i f_{n-i-1}$，可以发现$f_n$即为卡特兰数。

考虑$k > 1$的情况：设$L = \min\{a_i\}, R = \max\{a_i\}$。可以发现$[L, R]$内所有数值一定在路径lca 的子树内，其对答案贡献为若干f的乘积；而在固定路径形态后，其子树内$< L$和$> R$ 的点所属的子树是固定的，故可以把$[L, R]$ 的部分看成一个点。

时间复杂度$O(n + k \log k)$。

Number Elimination

题目大意

给定序列$a_1, a_2, ..., a_n$，每次操作(i, j)：把a_i, a_j中较小的数字删去，如果相同删去编号小的，操作的代价为$\max\{a_i, a_j\}$。操作到只剩一个数字为止。求共有多少种总代价最小的操作方案。

数据范围

$n \leq 10^5$

Number Elimination

显然相同的x，一定是相互操作删去的。那么x以下的只会有一个数参与x的操作。

问题变成：求m个x相互操作、一次x单独操作（其之前还有k次等价的特殊操作）的方案数。

枚举单独操作在相互操作中的位置计算组合数即可，复杂度$O(n)$。

Matrix Coloring

题目大意

最初有一个空白$n \times m$的网格纸，每次染色可以选一行或一列并将对应格子全部染成红色或蓝色。给定最终得到的网格纸（只有红色、蓝色），求最少多少次染色可以得到最终状态。

数据范围

$n, m \leq 3000$

Matrix Coloring

考虑如何判定是否有解：反向考虑整个过程，每次去掉一行颜色全部相同的行或列，看能否去完。

可以发现要么行全部去完，要么列全部去完。所以可以枚举哪一行不去，将那一行看做列操作的颜色。检验这样一组方案的方法是：先看每行去掉相同的列后是否为同色，然后对行和列建点，根据最后的颜色连有向边，看是否有环。

一个显然的观察为如果优解，不会出现$\frac{br}{rb}$这样的情况。那么同色一定满足，行的点一定连向列的点，故方案一定合法。答案即为$n+m$减去最多的相同行/列。

复杂度$O(nm)$。

An Unstable Graph

题目大意

给定n个点m条边的有向图，其中第i条边在每秒中出现的概率为p_i。现要求每秒内必须通过一条边走向另一个点，否则视为死亡。求在最优策略下从1活着走到n的概率。

数据范围

$n \le 50$，没有重边自环，只需绝对误差不超过10^{-6}。

An Unstable Graph

设f_i为i从到n的概率。容易发现f_i的计算与相邻的f_j的大小顺序有关。

考虑迭代：按照当前f的大小顺序来列f'的方程，做一遍高斯消元即可。

Cats

题目大意

给定一个字符集为abc的字符串s，每次可以将任意一个字符移到任意一个位置。求最少移动多少次，使得不存在字符a和b相邻。

数据范围

$|s| \leq 5000$

Cats

如果a的全部都动了，那么可以讨论剩余b的位置得到答案。下面计算a和b都有不动的方案数。

考虑去掉要动的a和b之后的字符串，可以发现所有要动一定能接在相同字母旁边，且答案为剩余段数$-$ab中间夹着c的数量$+$动的字符数量（剩余段数关系到是否合法）。可以想到维护$f_{i,j,x,y}$ 表示处理前i 个字符后有j 段，上一段是a 还是b，当前最后一个字符是否为c的最小值。时间复杂度$O(n^2)$。

Rooms

题目大意

给定一个$n \times m$的网格纸，每个格子均有颜色$a_{i,j}$，定义四连通且颜色相同的格子为一个地板块。

Q组询问:求左上角为(x_1, y_1)右下角为(x_2, y_2)的子矩阵中有多少种地板块中的格子。

数据范围

$n, m \leq 2000, Q \leq 5000$

Rooms

可以将地板块分成两种类型：全部在子矩阵中的和部分在子矩阵中的。

对于全部在的情况，可以给每个块一个关键点，统计子矩阵的关键点数量。

对于不全在的，可以发现一定有一个在子矩阵边上的格子，只需扫描这些格子即可。

复杂度$O(nm+Q(n+m))$。

Pitmutation

题目大意

已知有两个全排列a, b，对于每个i要么知道a_i的值，要么知道b_i的值。求一共有多少对这样的全排列满足恰好有k个位置满足$a_k > b_k$。

数据范围

$n \leq 300$

Pitmutation

设s_a为a知道的数值，s_b为b不知道的数值。

将这些数值排序后，设$dp_{i,j,k}$表示决策了前i个数值后，还有j个s_a中的未找到匹配，现在已有k个$a > b$的位置，转移直接决策是否与之前的匹配即可。

时间复杂度$O(n^3)$。

Popcorn

题目大意

给定n包爆米花，第i包的加热时间要求在区间$[a_i, b_i]$内，美味度为c_i。现有m个可加热的包裹，每个包裹在加热一定时间后里面的爆米花必须全部符合时间要求。问最大的美味度的和是多少。

数据范围

$n, m \le 2 \cdot 10^5$。

Popcorn

考虑最后m个加热时间，代价为完全出现在两个时间中的点的区间美味度之和，问题变成求最小代价。

设f_i为$m = i$时的最小代价，显然有$f_i - f_{i+1} \leq f_{i-1} - f_i$，可以用经典的凸优化来解决。二分答案之后，根据求最小代价可以想到用线段树进行优化。时间复杂度$O(n \log^2 n)$。

Product Replace

题目大意

给定序列$a_1, ..., a_n$，每次操作可以将两个数都变成他们的乘积。求一种总操作次数不超过5000的方案，使得最后所有数相同。

数据范围

$n \leq 100$，n为偶数

Product Replace

可以考虑归纳构造，假设当前已经满足$a_1 = ... = a_{2k} = b$，$a_{2k+1} = a_{2k+2} = c$：

$$
\begin{aligned}
&(b, b, b, ...b, c, c) \\
\rightarrow&(b, b, bc, b^2c, ..., b^{2k-2}c, b^{2k-2}c, c) \\
\rightarrow&(b, b, bc, b^2c, ..., b^{2k-2}c, b^{2k-2}c^2, b^{2k-2}c^2) \\
\rightarrow&(b^{2k-1}c^2, b^{2k-1}c^2, ..., b^{2k-1}c^2)
\end{aligned}
$$

每轮共$3k+1$次操作，那么总共不超过$\frac{3}{8}n^2$次操作。

Hamming Distances

题目大意

给定长度为n的序列$a(a_i < 2^m)$，要求对每组(i,j)，求a_i与$a_1 \sim a_{i-1}$中多少个数二进制表示恰好有j位不同。

数据范围

$n \le 10^5, m \le 16$

Hamming Distances

对位数分两种考虑，$f_{i,j,k}$表示前B位为i，后$m-B$位与j异或有k位为1的数字个数。

这样计算的复杂度为$O(2^B \times m)$，维护的复杂度为$O(2^{m-B})$。合起来为$O(n\sqrt{m2^m})$。

Cyclic Shifts

题目大意

给定n,m，试构造$n \times m$的矩阵x,y，满足$(x_{i,j} \bmod n, y_{i,j} \bmod m)$两两不同，且$((i+x_{i,j}) \bmod n, (j+y_{i,j}) \bmod m)$两两不同（先判断是否有解）。

数据范围

$n,m \le 300$。

Cyclic Shifts

n, m均是奇数,构造$(x,y)_{i,j} = (i,j)$即可。

n, m一奇一偶，根据同余性质，可以证明无解。

$n=2, m=2$有一组显然的构造：

$$\begin{bmatrix} (0,0) & (1,0) \\ (1,1) & (0,1) \end{bmatrix}$$

那么，如果$n' = 2n, m' = 2m$和$n' = (2p+1)n, m' = (2q+1)m$都可以类似上面地通过统一加上一个向量的方式构造。

Cyclic Shifts

下面考虑$n=2, m=2^k$的构造方式。

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0	0,0	1,0	0,1	1,1	0,2	1,2	0,3	1,3
1	1,4	0,4	1,5	0,5	1,6	0,6	1,7	0,7

可以发现对于偶数位置$2i$的y为i和$i+2^{k-1}(i \in [0, 2^{k-1}))$，由于不存在$3(i-j) \equiv 2^{k-1} (\bmod 2^k)$，故构造合法。

Divisible Matching

题目大意

给定左右点集大小均为n带权二分图，试判断是否存在完备匹配满足其权值和为s的倍数。

数据范围

$n, s \leq 100$

Divisible Matching

可以把完备匹配看成全排列，判断是否存在该全排列可以计算行列式$a_{i,j} = x_{i,j}$是否为0，其中如有边(i,j)，$x_{i,j}$为随机整数，否则为0。

对于和为s的倍数的要求，可以找到模s为1的质数p，及其满足$\omega^s = 1$的原根ω_s。

这样可以对上面的行列式进行变形$a_{i,j} = x_{i,j} \cdot \omega_s^{w_{i,j} \times c}$，对$c = 0 \sim s-1$求和，如果不为0就存在这样的完备匹配。时间复杂度$O(sn^3)$。

Find the Tree(Interactive)

题目大意

已知系统随机生成了一个树（给每个点随机一个标号小于他的父亲，然后打乱标号）。你每次可以询问3个点A, B, C，系统会返回$AD+BD+CD$最小的点D。试用不超过25000次操作回答出这棵树的所有边。

数据范围

$n \leq 2000$

Find the Tree(Interactive)

考虑用一条链对整棵树进行分治：每次随机两点a, b，把剩余所有点与这两个点询问一遍，可以得到所有路径上的点及对应子树的点集，并通过询问a, x, y即可将这些点排序，算出这些边。由于树是随机生成，直径期望为$O(\log n)$，即期望次数为$O(n \log n)$。