“睿琪杯”浙江省第 21 届大学生程序设计竞赛

题解

浙江大学

4.13.2024

构造一张图，使得贪心染色的颜色数大于等于图的色数 $+k$。

考虑构造一个二分图，使得贪心染色的色数为 $k+2$。
具体构造方案是左部和右部各 $k+2$ 个点，每部的第 i 个点，向另一部的 $j(j<i)$ 号点连边。这样贪心染色时每边的第 i 的点颜色为 i，总颜色数为 $k+2$，总点数 $n=2k+4$。

A. Bingo
Description

给定 n, m，求最大的 $x > n$，使得 x 是 m 的倍数，或者 x 十进制表达中包含 m 作为子串。

令 $p = x - n$，则 $p \leq m$。$n + p$ 是 m 的倍数的情况很好计算，现在考虑子串的情况：

注意到 $n + p$ 只会影响 n 的一个后缀，不妨枚举 m 在 $n + p$ 的哪一位上出现，以及在这一位开始上和 m 和 n 的最长公共长度。具体来说，枚举 $0 \leq k \leq len(m)$，并且令 $n = \overline{\dots m_1 m_2 \dots m_k \dots}$，并尝试计算出 p 的值，使得补齐 m 缺失的这一部分。

这部分贡献由两部分构成：首先假设 m 多出的部分是 $\overline{m_k m_{k+1} \dots m_{len(m)}}$，且 n 中对应的数位为 $\overline{n_k n_{k+1} \dots n_{len(m)}}$(这里将 n,m 下标对齐，方便理解)。则需要让 n 中 $n_{len(m)}$ 加一 (这部分需要的值可以通过记录后缀和与 10^i 的差计算)。然后将 $\overline{n_k n_{k+1} \dots n_{len(m)}}$ 中对应数位变为 $\overline{m_k m_{k+1} \dots m_{len(m)}}$(这部分贡献是若干个 10^i 的形式)。

枚举所有情况，找到最小的 p 即可。复杂度 $O(10 len(n))$.

需要注意一些细节的讨论。

给定一个未知的的排列，每次询问区间次大值，求全局最大值。要求询问次数不超过 $\lceil 1.5 \log n \rceil$，询问总长度 $3n$。

考虑如何在长度限制下做到 $2\log n$。假设已经知道了 $[l, r]$ 的次大值，令 $mid = \frac{l+r}{2}$ 只需要询问 $[l, mid]$ 的次大值。如果次大值位置和 $[l, r]$ 相同，则说明最大值在 $[l, mid]$，否则在 $[mid + 1, r]$。

这个做法询问次数是 $2\log n$ 的，是因为每次如果最大值落在 $[mid + 1, r]$，需要重新询问区间次大值。

但是可以注意到，两边的询问是不均等的，这启发我们做不均匀的划分。具体来说，假设目前长度为 n，我们询问一个长度为 $c \cdot n$ 的子区间，如果次大值位置和当前区间相同，则最大值在这个区间，否则最大值在另一半区间。

此时，询问次数是 $T(n) = \max(T(c \cdot n) + 1, T((1-c) \cdot n) + 2)$，可以通过假设 $T(n) = \log_x n$ 来解出 c 和 x。最终最优的 $c = \frac{\sqrt{5}-1}{2}$，$x = \frac{\sqrt{5}+3}{2}$。此时询问总长度也为 $\frac{\sqrt{5}+3}{2} n$。

另一种做法是通过 dp 算出每次最优的 c，也可以通过。

给定一个 DAG，满足 $\sum_{i=1}^{n} |in_i - out_i| \leq 120$，求每个点出发可以抵达的点的个数。

首先这个限制可以推出，整张图可以拆成不超过 60 条边不交的链的并，具体做法有很多，其中一种是每次选择一个 $out_i > in_i$ 的点，从它出发找到一个能抵达的 $in_i > out_i$ 的点。可以证明每次操作之后 $sum_{i=1}^{n}|in_i - out_i|$ 减少 2。

现在我们说明，可以在 $O(n+m)$ 时间内，求出一条链上的所有点对应的答案。因为对于链上任意一个点，它所能抵达的点一定被它前面的点能到的点包含。这样每次从链的最末端到最顶端的每个点开始 bfs，并且在 bfs 的过程中，碰到之前访问过的点（在这条链上的后继点访问过的点）就可以直接返回，每次统计多访问了几个点即可。
时间复杂度 $O(60(n+m))$。

K. Sweet Sugar 3

Description

给定 A 个 1，B 个 2，C 个 3，按顺序做摩尔投票。假设途中有 m 次变成 0，且最后为 0，则贡献 m^x。求所有排列的贡献和。

考虑从一个 0 到下一个 0 的过程，这过程中牌堆一直非空，并且其中的卡牌一定是同一种。称这种卡牌为这个过程中的主元。令 $n = A + B + C$。

考虑分成两部分计数，一部分是选择 $\frac{n}{2}$ 个主元，和 $\frac{n}{2}$ 个空格（我们并不关心空格中填入了什么数），排成一列的方案数。另一部分是将剩余的非主元的数字填入空格中的方案。

先预处理 $f_{i,j}$ 代表长度为 $2i$ 的序列，中间恰好经过了 j 次 0 的方案数。这部分通过卡特兰数进行转移即可。
枚举 A 中有 a 个数，B 中有 b 个数，C 中有 $c = \frac{n}{2} - a - b$ 个数成为主元。令生成函数 $F_i(x) = \sum \frac{f_{i,j} x^j}{j!}$，则 $[x^i]F_a(x)F_b(x)F_b(x)$ 是这些主元恰好组成了一个经过 i 次 0 的序列的方案数。
第二部分的计数比较简单，限制要求剩余的 $A-a, B-b, C-c$ 个数不能填入自家的空格中，枚举剩余的 $A-a$ 有几个填入 B 的空格就能解出所有填入空格的情况，组合数计算即可。

K. Sweet Sugar 3

Solution

上面的做法直接做是 $n^3 \log n$ 的，而且需要 fft 进行计算。不过一个经典的技巧是，可以直接使用点值乘积代替多项式系数做卷积，这样卷积复杂度降为 $O(n)$ 单组。并且我们注意到，每一轮过程中，我们并不关心系数具体是多少，只关心所有轮次之后系数和是多少。将每一轮计算出来的点值加起来，最后统一插值算出答案即可。时间复杂度 $O(n^3)$，此处 $n = 500$。

B. Simulated Universe

Description

给定一个包含 B 和 C 的序列，每个 B 至多匹配一个 C，每个 C 要么选择匹配不超过 a_i 个它左侧的 B，要么选择匹配不超过 b_i 个它右侧的 B。求最多产生多少对匹配。

B. Simulated Universe

Solution

一个很粗暴的 n^3dp 是，令 $f_{i,j,k}=0/1$，表示前 i 个字符，还剩下 j 个 B 没有匹配，且还能匹配后面的 k 个 B，这种情况是否可以取到。转移枚举当前的 C 是向左匹配还是向右匹配。
可以将其中一维压入 dp 值，令 $f_{i,j}$，表示前 i 个字符，还剩下 j 个 B 没有匹配，在这种情况下最大的 k 能取到多少。转移同样枚举 C 的匹配方向。计算答案时，找到 $f_{n,j}$ 合法的最小的 j 即可。
时间复杂度 $O(n^2)$。

F. Stage: Agausscrab

Description

给出一个比赛命名规则，求比赛名称。

模拟即可。可以通过枚举或排序求出每个出题人的题数排名，并由此得到最终结果。
时间复杂度 $O(n^2)$ 或 $O(n\log n)$，取决于使用枚举还是排序求排名。

给定一个边长为 n 的三角形网格，删掉一条长为 1 的边，求剩余几个三角形。

M. Triangles
Solution

第一步：统计在删除边之前一共有多少三角形
做法：枚举三角形下边界（或倒三角形的上边界）所在边，通过组合数算出有多少满足条件的三角形，然后求和。

第二步：统计删掉的边在多少三角形上
做法：枚举所在三角形最左侧的点，根据右边界和下边界的限制求出有多少满足条件的三角形，然后求和。

最后，将第一步的结果减去第二步的结果即可得到答案。
时间复杂度 $O(n)$。可以通过数学推导做到 $O(1)$。

I. Piggy Sort

Description

直线上有 n 个匀速、同向运动的动点，初始位置互不相同。给定 $m > n$ 个互不相同的未知时刻下的照片（不知道动点在多张图片上的对应关系），求动点的速度排序。

搜索即可。

特殊情况：所有速度均为 0。

性质 1：除了特殊情况外，对于每张照片，所有点坐标之和与时间的关系是一次函数，因此可以算出每张照片的“拍摄时间”。

性质 2：如果发现了一个点在 m 张照片对应时间的位置都能出现（存在一条位移-时间直线通过这些点），那么一定存在这个动点。

证明：根据抽屉原理，至少有一个点在两张照片中出现。

由于速度恒定，在至少两张照片中出现的点一定经过全部 m 张照片上对应的点。

搜索出一个动点后，将这个动点在所有照片上的位置删去，剩余情况与原题相同，因此可以递归解决。

时间复杂度 $O(n^2m)$。

D. Puzzle: Easy as Scrabble

Description

给定一个网格，要求在其中部分格子中填入字母，边界上有一些方向第一个非空格所填字母的线索，内部有一些强制是空格的位置。求一个合法填充方案。

D. Puzzle: Easy as Scrabble

Solution

对于一个“第一个非空格所填字母”限制，这个限制会加在一个特定的格子上。
对于一个格子，如果它有若干个方向的限制不同，那么它一定是一个空格。
反复使用以上两个性质，直到没有矛盾，或存在一个线索没有可填的格子为止。具体实现方式类似 BFS，使用队列维护矛盾格子即可。
时间复杂度 $O(nm)$。

J. Even or Odd Spanning Tree

Description

给定一张无向图，分别求边权和为奇数和偶数的最小生成树。

将所有边按边权分为奇数边和偶数边。
令 $f(k)$ 表示恰好选择 k 条奇数边的最小生成树。
求出整个图的最小生成树 T，假设它包含 k 条奇数边，显然 $f(k)$ 是 f 函数的最小值。
又因为 f 函数是凸函数，因此另一个答案只能是 $f(k-1)$ 或 $f(k+1)$。
根据拟阵的性质，往 T 中替换恰好一条非树边一定能得到 $f(k-1)$ 和 $f(k+1)$。
使用树上数据结构维护路径奇偶边的最大值即可。
时间复杂度 $O(m\log n)$。

n 个学生分组进行小组作业，第 i 个学生要求他所在小组的人数在 $[l_i, r_i]$ 之间。
如果一组包含 k 个人，那么得分为 w_k。找到总得分最大的方案以满足所有学生的要求。

假设确定了每个人数分别有多少组，考虑如何判定可行性。
按人数从小到大依次考虑每个组，假设当前这组需要包含 k 个人，显然可以贪心选择 r 最小的 k 个人。
利用位运算实现 $O(1)$ 取最小值、插入和删除，判定一次的时间复杂度为 $O(n)$。
注意到 60 的整数拆分方案只有 966467 个，枚举所有拆分方案然后判定可行性即可。
时间复杂度 $O(n \cdot P_n)$。

给定 n 个点的树，有 m 只乌龟在根节点。控制这些乌龟进行移动，使得第 i 个点至少被 c_i 只不同的乌龟爬过，使得所有乌龟爬行总路程之和最小。
乌龟很重，一条边最多允许被爬过 k_i 次。
需要对 $m = 1, 2, \ldots, M$ 输出答案。

令 x_i 表示 i 点从父亲往下有多少只乌龟下去，y_i 表示此时有多少只乌龟在子树中不再回来。那么需要满足：

- $x_i \geq c_i$，
- $x_i \geq y_i$，
- $2x_i - y_i \leq k_i$，
- $x_i \geq$ 它儿子的 x 值的最大值，
- $y_i \geq$ 它儿子的 y 值之和。

于是有树形 DP：设 $f_{i,x,y}$ 表示考虑 i 的子树及对应 x, y 的最优解。
暴力枚举儿子状态转移，时间复杂度 $O(nm^4)$，不能接受。

注意到固定 y_i 之后，x_i 的最优取值一定是 $\max(y_i, 子树\ c\ 的最大值)$，于是无需记录 x，状态优化为 $f_{i,y}$。
通过归纳可知 $f_{i,y}$ 是关于 y 的凸函数，因此转移可以用闵可夫斯基和做到 $O(m)$，时间复杂度 $O(nm)$。
更进一步地，如果使用 Splay 维护凸函数，那么通过启发式合并可以做到 $O(n \log n)$。
特别注意当 m 小于最大的 c 时一定是无解。

Thanks!