CCPC Weihai 简要题解

2021 年 11 月 21 日

前言

- 预计难度：A $<$ DGJ $<$ EHM $<$ I $<$ FK $<$ C $<$ BL
- 封榜前难度：A $<$ DGJ $<$ EHM $<$ F $<$ IK $<$ BC $<$ L
- 基准时限设为 0.5s 是因为 PTA 太快了。

A. Goodbye, Ziyin!

- 题意：给出一颗无根树，询问有多少个点以它为根时是棵二叉树。
- 一棵有根树为二叉树当且仅当：
  - 根至多只有两个儿子 $\Rightarrow$ 根的度数 ≤ 2
  - 非根节点至多只有两个儿子 $\Rightarrow$ 非根节点的度数 ≤ 3（加上父亲）
- 先判断是否存在度数 > 3 的节点，然后统计有多少个度数 ≤ 2 的节点即可。

A. Goodbye, Ziyin!

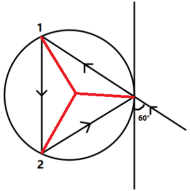
- 一个不算彩蛋的彩蛋：

  In short, given an unrooted tree, please calaculate how many nodes satisfying that if the tree is rooted by the node, the rooted tree is a rooted binary tree.

- 简短题面被故意安排到第二页，如果直接看网页版可能会一眼看出来。
- 愿有情人终成眷属。

J. Circular Billiard Table

- 题意：从圆的边缘以某个角度发射一颗小球，小球在球内部沿反射定律运动，问什么时候回到出发点。
- 注意到小球每两次反射之间的圆心角是相同的，不妨设为 α



- 若小球反射 n 次后回到原点，则 $n\alpha = k \cdot 360^\circ$（$k$ 为某个自然数）
- 换言之，最小的 n 即为最小的 n 使得 $360 | n\alpha$，利用 gcd 算一下即可

D. Period

- 题意：给定一个只包含小写字母的字符串，每次询问将某个位置修改为 #，问周期数量（修改独立）。
- 因为串长-周期长度= border 长度，所以周期数量等价于求 border 数量，又因为修改的字符是之前完全没出现过的，所以修改完之后的 border 只能原串的border。
- 先使用 kmp/hash 求出原串的 border，然后预处理/每次询问二分有多少个 border 不包含某个位置即可（注意到这等价于有多少个 border 小于等于某个长度）

G. Desserts

- 题意：有 n 种糖果，第 i 种糖果共有 a_i 个，问把所有糖果分给 k 支队伍，使得每支队伍不会拥有相同类型糖果的方案数。保证 $\sum a_i \le 10^5$。
- 考虑第 i 种糖果分给哪些队伍，可得 $ans_k = \prod_{i=1}^{n} \binom{k}{a_i}$
- 又因为 $\sum a_i \le 10^5$，所以本质上不同的 a_i 只有根号种，直接枚举每种不同的 a_i ，然后相同的 a_i 快速幂贡献在一起即可。
- 复杂度 $O(m\sqrt{\sum a}\log n)$

E. CHASE!

- 题意：有 n 个数，每次系统随机选出两个数 a_i, a_j，你可以接受或重选，但总共只有 k 次重选机会。问期望最大和。还有 q 个子情况需要回答最优决策。
- 如果 $k = 0$，那么没有重选的机会，期望得分 E_0 为任选两个数字的和的期望；
- 如果 $k > 0$，假设当前选出来的和为 s，那么 $s \geq E_{k-1}$ 时不需要重选，否则需要重选。
- dp，用 two pointers 算这两种贡献来转移。

M. 810975

- 题意：求长度为 n 的 01 串，有 m 个 1，最长 1 连续段长度恰好为 k 的方案数。
- 先容斥，转化成 $\leq k$ 的方案数。
- 如果两个连续的 0 视为中间有长度为 0 的 1 连续段，则相当于一共有 $n-m+1$ 个 1 连续段，每段长度 $\leq k$，长度和为 m 的方案数。
- 形式化地，设 x_i 为第 i 个 1 连续段的长度，即求：
  - $0 \leq x_i \leq k$
  - $\sum x_i = m$

  的方案数。
- 这是一个经典问题，容斥枚举有多少个 x_i 违反 $\leq k$ 的限制即可。
- 多项式快速幂也可以过，没有刻意去卡。（没过说明板子常数可能较大）

H. City Safety

- 题意：一棵树，加强第 i 个点有 w_i 的花费，而如果距离某个点 $\leq p$ 的所有点都加强了，则会有 v_p 的收益，求最大净收益。
- 转化为最小割。先把所有收益加到答案里。
  - 左边的点 $v_{i,p}$ 表示距离第 i 个点 $\leq p$ 的所有点全选
  - 右边的点 u_i 表示原图的点 i
  - 源点连向每个 $v_{i,p}$，容量为增量收益 $v_{i,p} - v_{i,p-1}$，割掉这条边表示放弃这个增量收益
  - 每个 $v_{i,p}$ 连向右边距离 i 为 p 的点，容量为无穷大
  - 每个 $v_{i,p}$ 连向 $v_{i,p-1}$，容量为无穷大，这样使得 $v_{i,p}$ 间接连向与 i 距离 $< p$ 的点，用来限制放弃收益必须按 p 从大到小放弃
  - 右边每个点连向汇点，容量为选这个点的代价，割掉这条边表示付出这个点的代价
- 答案减去最小割。

H. City Safety

- bonus，有个 $O(n^2)$ 的树形 dp 做法。

F. Stone

- 题意：轮流拿石头，每个人先确定一个上次选的数字的因数 s，然后再选择若干堆同时拿走 s 个石头，问先手有多少种第一步的方案数使得先手必胜。
- 考虑这样一种局面：存在某堆石头的数量是奇数。那不管是先手还是后手，选择 $s=1$(奇数) 先把所有堆变成偶数个，然后对手也只能 $s=1$(奇数)，模仿对面拿法即可。
- 换言之，如果所有堆都是偶数，则不管是谁都不能选 s 为奇数，否则变成前面一种情况，对手必胜。即 s 只能为偶数，此时堆数量 $/=2$，转化为相同的问题。
- 那么一直除以 2，总会在某个时候出现某个石头堆有奇数个，转换为前面的局面。
- 此时答案为 (最小奇数$+1)/2$。

I. Distance

- 题意：给出一张哈斯图，求两两最短路之和。
- 易知两个点 i, j 的距离为 $\sum_{p \in prime} p \times |e_i - e_j|$
- 枚举每个质数的贡献，可得答案为 $\sum_{p \in prime} p \times \sum_c \lfloor \frac{n}{p^c} \rfloor \left(n - \lfloor \frac{n}{p^c} \rfloor \right)$
- 小于 $\sqrt{n}$ 的质数可以直接枚举，大于 $\sqrt{n}$ 的质数只会枚举 $c = 1$，整除分块+min25 即可。

K. Tiny Stars

- 给出奇质数 n，决定每个 i 是连向 $\frac{i}{a}$ 还是 $\frac{a}{i}$（各有连边代价），然后每个连通块付出大小平方的代价。总代价 $\leq 12n$ 即可通过。评测时每个 n 会随机生成 20 个 a，任意一个 a 通过即可。
- 观察到 $i \to \frac{i}{a} \to (\frac{a}{\frac{i}{a}} = \frac{a^2}{i}) \to (\frac{\frac{a^2}{i}}{a} = \frac{a}{i}) \to (\frac{a}{\frac{a}{i}} = i)$，即只要能够实现交替连边，那么图就会变成全是四元环，连通块代价为 $4^2 \times \frac{n}{4} = 4n$。

K. Tiny Stars

- 实现交替连边即每个 i 连向的点，$oracle$ 必须与自己不同。这有至少两种方法可以实现：
  - 二次剩余，$oracle_i = 0$ 当且仅当 i 是 n 的二次剩余，这样当 a 不是二次剩余时（概率 50%）性质成立；
  - 指数，设原根为 g，$oracle_i = 0$ 当且仅当 $i = g^x$ 且 x 是奇数，这样当 a 的指数也是奇数时（概率 50%）性质成立。
- 这两种的连边代价都是 $\frac{n}{2} + \frac{n}{2} \times \log n \leq (7 + \frac{1}{2})n$，总代价 $\leq (11 + \frac{1}{2})n$。
- 随机生成 20 个 a 全部失败的概率 $\leq (\frac{1}{2})^{20}$，可忽略。

C. Assign or Multiply

- 问 $a_i \times \prod_{j\in S, S\in 2^b} b_j$ 在 $\bmod p$ 意义下有多少种可能。
- 先利用原根把乘法转化为加法（0 特判）。
- 有个暴力 $O(nq)$ 的做法：

```
for (int i=0;i<q;++i)
    for (int j=0;j<n;++j)
        new_dp[j]=dp[j]|dp[(j-b[i]+n)%n];
```

- 可以使用数字分组+ bitset 加速做到 $O(n^2/w)$，但这不是 expected 的做法，但某支队伍好像加了点特判跑过了（复杂度未知）。

C. Assign or Multiply

- 注意到 bitset 只执行 or，如果每次可以快速找到哪些 0 会被变成 1 会变快点？
- 好像比较难，那快速找到 i 和 (i+x)%p 不同的点然后看是否为 bitset[i]=1，bitset[(i+x)%p]=0？
- 假设新加入的数字为 x，则可以看成 i→(i+x)%p 连一条边，那么每个点恰好一个入度和一个出度——这是若干个环！
- 那么如果存在某条边 i→(i+x)%p 使得原本 bitset[i]=1，bitset[(i+x)%p]=0（简称为 $0 \to 1$），则一定有一条边是 $1 \to 0$ 的，即 $0 \to 1$ 和 $1 \to 0$ 的数量是相同的。
- 由于 $0 \to 1$ 只有 n 条，故复杂度为 $O(n \cdot \text{寻找不同点对})$ 的复杂度。
- 利用二分+树状数组可以做到 $O(n \log^2 n)$.

L. Shake Hands

- 题意：n 个人站在一排，每次相邻两个人握一次手并交换位置，问最大团。
- 假设 $a < b < c$ 并且 a 和 b 没有握手，b 和 c 没有握手，则 a 和 c 也一定没有握手。即没有握手的关系形成了一个闭包！
- 故原图最大团 $\rightarrow$ 补图最大独立集 $\rightarrow$ 补图 DAG 最长反链 $\rightarrow$ DAG 最小路径覆盖 $\rightarrow$ 二分图最大匹配。所以答案即为 $n -$ 补图的最大匹配。
- 注意到补图为竞赛图去掉 m 条边，可以类似于牛客多校 10C 的做法根据度数分成小于 sqrt 贪心+大于 sqrt 暴力匹配即可。
- 复杂度 $O(n\sqrt{m})$

B. Subset

- 题意：从 $[0, n]$ 中选 K 个数，使得异或和有 B 位 1 的方案数。
- 假设 $n = 2^m - 1$，考虑这样一个 dp：
- $dp[i][x]$ 表示选了 i 个不同的数 $x_1, x_2, \cdots, x_i$，异或起来恰好为 x 这个数的方案数。

B. Subset

- $dp_{i,x}$ 表示选了 i 个不同的数 $x_1, x_2, \cdots, x_i$，异或起来恰好为 x 这个数的方案数。
- 考虑容斥，则先选 $i-1$ 个数 $x_1, x_2, \cdots, x_{i-1}$，那么 $x_i = x \oplus x_1 \oplus x_2 \oplus \cdots \oplus x_{i-1}$，因为 $n = 2^m - 1$，所以 x_i 一定满足值域要求，不一定满足 x_i 互不相同的要求。
- 所以要扣掉 x_i 等于 $x_1, x_2, \cdots, x_{i-1}$ 中某个数的情况：
- $dp_{i,x} = (i-1)! \times C_n^{i-1} - \underbrace{dp_{i-2,x}}_{\text{扣掉2个还剩i-2个}} \times \underbrace{(n-i+2)}_{\text{剩下 n-i+2 个值可以选}} \times \underbrace{(i-1)}_{\text{选择其中一个位置插进去}}$

B. Subset

- $dp_{i,x} = (i-1)! \times C_n^{i-1} - dp_{i-2,x} \times (n-i+2) \times (i-1)$
- 注意到 $dp_{i,x}$ 只和 $dp_{i-2,x}$ 有关，并且边界为：
- $dp_{0,x} = [x == 0]$，$dp_{1,x} = 1$。
- 故 $dp_{i,x}$ 的值只与 x 是否为 0 而不同。

B. Subset

- 将 $[0, n]$ 拆成若干个区间，每个区间都有这个性质，区间与区间之间可以暴力 dp 合并，这样复杂度为 $O(k^2 \log^2)$。
- 但因为每个区间只有 0 的 dp 值不一样，所以转移的时候可以分成 0 和非 0 进行转移，类似于第一维是卷积，第二维是区间平移。
- 又因为第二维每次区间平移最多产生 $O(1)$ 个值域段，故 dp 数组至多 $O(\log)$ 个值域段。
- 对每个值域段维护一个 $O(K)$ 的 dp 数组（第一维），那么 dp 转移可以简化为值域段内的卷积，复杂度为 $O(K \log K \log^2 n)$。
- 在赛场上有队伍使用了复杂度为 $O(K^2 + K \log n)$ 的做法，也可以通过。