2020 Wuhan University Collegiate Programming Contest (Final Contest)

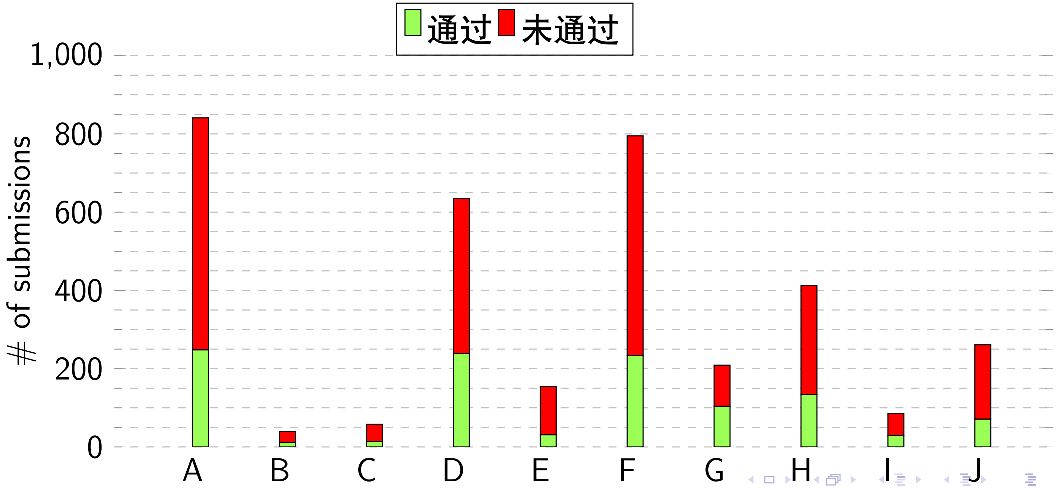
Solutions

Apr. 19th, 2020

比赛小结

- 本次比赛共收到 3656 份提交代码。
- 303 名参赛选手有提交记录。
- 279 名参赛选手至少通过一题。

各题通过情况



A. A Simple Problem about Election

题意

有 n 个人参加竞选，每个人必须投票给 m 个不相同的人，现在最后一个人来投票，他知道目前所有人的得票，问他投完之后最好名次可以是多少。

A. A Simple Problem about Election

题意

有 n 个人参加竞选，每个人必须投票给 m 个不相同的人，现在最后一个人来投票，他知道目前所有人的得票，问他投完之后最好名次可以是多少。

首先达成一个共识：第一票投给自己。

A. A Simple Problem about Election

题意

有 n 个人参加竞选，每个人必须投票给 m 个不相同的人，现在最后一个人来投票，他知道目前所有人的得票，问他投完之后最好名次可以是多少。

首先达成一个共识：第一票投给自己。

投完自己之后再来思考，哪些人得票之后会影响我的排名呢？结果发现只有刚好和自己差一票（也就是之前跟自己票数相等）的人会影响。

A. A Simple Problem about Election

题意

有 n 个人参加竞选，每个人必须投票给 m 个不相同的人，现在最后一个人来投票，他知道目前所有人的得票，问他投完之后最好名次可以是多少。

首先达成一个共识：第一票投给自己。

投完自己之后再来思考，哪些人得票之后会影响我的排名呢？结果发现只有刚好和自己差一票（也就是之前跟自己票数相等）的人会影响。

所以本题的做法就是：先投自己，然后投所有和自己之前票数不相等的人，如果还有余票再去投这些人。投完之后只需看大于等于自己票数有多少的人有多少。复杂度为 $O(n)$。

A. A Simple Problem about Election

可能存在的盲点

- “优先投比自己票低的不就好了”
- 直接每次 memset 原数组可能会 TLE

花絮

本题根据真实经历改编。

F. Figure out the Sequence

题意

给定两个字符串 s，t。先假设 $F_1 = s, F_2 = t$，然后对于任意的 $i > 2$，都有 $F_i = F_{i-2} + F_{i-1}$。求最后的串中各字符出现次数。

F. Figure out the Sequence

题意

给定两个字符串 s，t。先假设 $F_1 = s, F_2 = t$，然后对于任意的 $i > 2$，都有 $F_i = F_{i-2} + F_{i-1}$。求最后的串中各字符出现次数。

数据都不大，无需维护字符串本身，只需要维护 F_i 中各字母出现次数即可。这样只需要维护一个长度为 52 的数组，岂不美哉。

F. Figure out the Sequence

可能存在的盲点

- 还是题意理解出现了偏差。
- 看上去很恐怖，但这题对于 C/C++ 真的 int 范围内存得下答案。

花絮

本题原本是一个一不小心出多了的签到题，但经过一系列机缘巧合，它又复活了。

H. Hinnjaku

JO 厨狂喜

题意

建议阅读题目原文

H. Hinnjaku

JO 厨狂喜

题意

建议阅读题目原文

这题直接字符串模拟即可，注意一些判定的顺序。

H. Hinnjaku

可能存在的盲点

- 迷惑数据大赏

花絮

吸取预选赛 E 题的教训，这次本人亲自重新完善了一遍题面，并加强了数据。

验题人在加强数据的过程中，想到了一个可能很容易出错的数据，于是把它加入到了测试集中。果不其然，它卡掉了出题人的原标程。

题目当中很多关于 JOJO 的背景资料补充说明都是本人而并非出题人加的。

D. Deploy the Medical Team

题意

有 n 个人，现需要选出一些人组成队伍，并且队伍中必须有且仅有一名队长，这些人中只有 m 个人能够担任队长，问能组建多少种不同的队伍。

D. Deploy the Medical Team

题解

原题转化为：有 m 个人，你需要至少从中选取 1 人并指定一位队长，然后剩下的 $n-m$ 人里任选搭配。

更进一步，就变成了：从 m 个人里选一位队长，然后剩下的人任选！

所以说答案就是 $m*2^{n-1}$。

数据范围说明你需要知道一个叫快速幂的算法，这样总体复杂度为 $O(T\log n)$

I. Interesting Matrix Problem

题意

给定一个 N 行 M 列的矩阵，其中第 i 行第 j 列的数字是 $i \times j$，现在有多次询问，每次询问矩阵中第 k 大的数字在哪里。

I. Interesting Matrix Problem

k 的范围并不大，矩阵内的很多元素是用不上的。显然 $1 \sim k$ 的每个数都会出现在矩阵中，对于一个数 x，只需要知道 x 在矩阵中出现了多少次即可，如果我们能快速算出 $1 \sim x$ 在矩阵中个数，通过二分即可快速求出答案。

I. Interesting Matrix Problem

k 的范围并不大，矩阵内的很多元素是用不上的。显然 $1 \sim k$ 的每个数都会出现在矩阵中，对于一个数 x，只需要知道 x 在矩阵中出现了多少次即可，如果我们能快速算出 $1 \sim x$ 在矩阵中个数，通过二分即可快速求出答案。

考虑一个数 x，它在矩阵中出现的次数，即是有多少个坐标 (i,j) 满足 $i \times j = x$，这类似于求 x 的约数个数，但要注意好边界：

$$\sum_{d|x, d \leqslant N} [\frac{x}{d} \leqslant M]$$

显然交换矩阵的 N 与 M 不会影响最终的答案，因此我们总是设 $N \leqslant M$，这样，再注意到上式中 $x \leqslant M$，因此可简化为

$$\sum_{d|x, d \leqslant N} 1$$

I. Interesting Matrix Problem

这只是求单个数出现的次数，对于 $1 \sim x$，需要求前缀和，即

$$\sum_{i=1}^{i \leqslant x} \sum_{d|i, d \leqslant N} 1$$

接下来就是常用技巧了，改变求和顺序，考虑一个 d 会有多少个 i 与之出现在上面的两重和式中，显然是 $\lfloor \frac{x}{d} \rfloor$ 个，即上式等于

$$\sum_{d=1}^{\min(N,x)} \lfloor \frac{x}{d} \rfloor$$

接下来用分块求和来把计算该式的时间复杂度降低到 $O(\sqrt{\min\ N, x)})$ 即可。

J. Jogging along the Yangtze River

题意

你需要在坐标系上从 $(0,0)$ 移动到 $(2n,0)$，有三种走步：

1. (x,y) -> $(x+2,y)$
2. (x,y) -> $(x+1,y+1)$
3. (x,y) -> $(x+1,y-1)$

并且你不能移动到 x 轴下方的位置，求方案数目。

J. Jogging along the Yangtze River

题意

你需要在坐标系上从 $(0,0)$ 移动到 $(2n,0)$，有三种走步：

1. (x,y) -> $(x+2,y)$
2. (x,y) -> $(x+1,y+1)$
3. (x,y) -> $(x+1,y-1)$

并且你不能移动到 x 轴下方的位置，求方案数目。

显然这个东西和卡特兰数比较像（这个数列也叫做超级卡特兰数或者大施罗德数），我们给两种做法。

J. Jogging along the Yangtze River

法一

我们考虑把它转化为卡特兰数。我们可以枚举路径中 (1,1) 的数量，可以得到这样的式子

$$F_n = \sum_{i=0}^{n} \binom{n+i}{2i} C_i$$

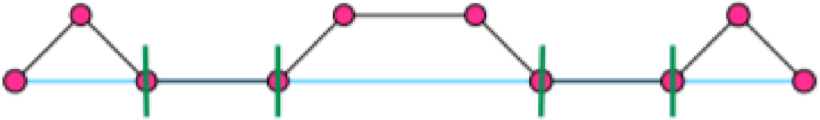
其中 Ci 为卡特兰数。也就是我们先枚举斜着走的路径数，然后剩下的 n-i 步都是 (2,0)，我们把它插入进去就可以了，一共有 2*i+n-i 步，然后里面插 n-i 个 (2,0)，然后把卡特兰数的公式

$$C_i = \frac{(2i)!}{i!(i+1)!}$$

代入即可。

法二

我们换一个方法考虑，我们考虑把一条路径按照它到达 x 轴的次数分成多段，例如：



我们可以把它分成这样 5 个段，每个段除了开头和结尾，其他点全部不在 x 轴上。每个长度为 2k 的小段或者是只有一步 (2,0)，或者由 (1,1),(1,-1) 和它们之间的一条长度为 2(k-1) 的路径组成。

J. Jogging along the Yangtze River

我们构造一个指数型生成函数 R(x)

$$R(x) = \sum_{i=0}^{\infty} f_i x^i$$

其中 fi 为答案。根据上面的关系，我们可以得到 R(x) 的一个表达式

$$R(x) = \sum_{k=0}^{\infty} (x + xR(x))^k = \frac{1}{1 - x - R(x)}$$

这个式子展开化简之后可以得到

$$R(x) = \frac{1 - x - \sqrt{x^2 - 6x + 1}}{2x}$$

J. Jogging along the Yangtze River

通过解它的系数我们可以得到 f 的递推式

$$(n+1)f_{n+1} = 3(2n-1)f_i - (n-2)f_{i-1}$$

代入 f0=1,f1=2 递推即可。

E. Engage the Medical Workers

题意

给定一个二维方阵，你需要构造一个新的方阵，给定的大小关系与原方阵保持一致。

E. Engage the Medical Workers

题意

给定一个二维方阵，你需要构造一个新的方阵，给定的大小关系与原方阵保持一致。

将矩阵中所有数字排序，然后从小到大处理。

每一次将原矩阵中相等的数字填入新矩阵时，将这些数字所在新矩阵的行和列中已经填的最大的数字找到，则当前处理的相等数字的值就等于找出来的最大数字 $+1$.

E. Engage the Medical Workers

花絮

本次比赛的第二次撞题发生在这里。而且是比赛内撞题。
有两位出题人出了跟这题题意几乎一致的题，最后只能去掉一个人的题，并把前面一不小心多出的 F 题拿过来用。

G. Game Strategy

题意

三个人玩游戏，每个人每一轮从自己的数组中删去一个数，三个人分别希望最后三人剩下的数之和最大，最小，绝对值最小，求三人均按照最优策略后最终的和是多少。

G. Game Strategy

题解

首先注意到，游戏进行 $n-1$ 轮，每一轮 Alice，Bob，Cindy 轮流移除自己的一个数，直到大家都只剩下一个数。这等价于进行一轮游戏，每个人选择一个数留下，其它数都扔掉。
接下来再考虑每个人会留什么数，我们建立一个 $n \times n \times n$ 的三维立方体，立方体中的位于 (i, j, k) 的数是 $a[i] + b[j] + c[k]$。则可以进一步抽象：Alice 首先选择立方体的一个面，Bob 再从这个面中选择一列，最后 Cindy 从这一列中选择一个元素。

G. Game Strategy

题解

无论之前 Alice，Bob 是怎么选的，最后 Cindy 一定会在一列中选择绝对值最小的元素（根据题意，有多个时选择正的那个），Bob 知道 Cindy 的这个策略，那么他的最优策略一定是在一面中选择这样一列：绝对值最小的那个元素最小的那一列。同样，对于 Alice 来说，她可以计算她选了一个面之后，Bob 所选择的列，Cindy 所选择的元素，即可以计算出最终的答案，Alice 只需要枚举每一个面，并选择能使得答案最大的那一面即可。根据上面的算法，复杂度是 $O(n^3)$

B. Build the Huoshenshan Hospital

题意

输出给定脚手架需要多少钢筋构成，大模拟。

B. Build the Huoshenshan Hospital

题意

输出给定脚手架需要多少钢筋构成，大模拟。

重叠的钢筋只有两种情况：

1. 面对面相接，这种情况需要判端各类钢筋接触面的相同钢筋数目。
2. 有一条棱相接，在某些相邻的面没有其它立方体的情况下，必须得被挖掉。

建议采用从左到右从前到后从下到上的顺序，每次添加一个框架，然后删掉这个框架与当前整体重复了的钢筋的做法。

B. Build the Huoshenshan Hospital

花絮

本题正是之前网络赛提到的那个因为难度过大被撤下来的那个题，它并没有完全去世，而是在这里复活了。
新的验题人写了几天表示怎么对拍都不对，最后检查发现真的是标程写错了。但经过修改的标程仍然仅 90 行。

C. Calculate the Sanity Value

题意

定义了一个字符串的 SAN 值为这个字符串可拆分的连续的相同字串的最大数目，如 ababab 的 SAN 值为 3，abcabc 的 SAN 值为 2，你需要对于一个给定串 S 求出它的所有子串里最大的 SAN 值。

C. Calculate the Sanity Value

解法

本题就是求重复数最多的字串，如果重复数为 1，那么做法显然，然后只考虑重复数大于 1 的情况。
从小到大枚举长度 len，对于每个关键点 $x = i \times len$，有且仅有一个长度为 len 的串经过它。
算出 x 与 $x + len$ 的最长公共前缀 A 和最长公共后缀 B 后，贡献为 $\lfloor \frac{(A+B-1)}{len} \rfloor + 1$。
时间复杂度 $O(n \log n^2)$