

CUGACM2024 入队选拔 Solution

CUGACM

2024 入队选拔



中国地质大学
CHINA UNIVERSITY OF GEOSCIENCES

过题情况（封榜前）

题目	定位	通过次数	未通过次数	题目	定位	通过次数	未通过次数
A	签到题	48	17	H	简单题	10	83
B	中档题	0	11	I	防 AK	0	0
C	简单题	23	32	J	中档题	0	6
D	困难题	1	25	K	中档题	1	8
E	简单题	5	7	L	签到题	22	23
F	中档题	0	3	M	困难题	0	8
G	签到题	44	67				

A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看

数组

题目大意

给定 n ，输出大小为 n 的数组的最后一个下标。

签到题。输出 $n - 1$ 即可。

A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看

借阅图书

题目大意

题目要求模拟图书馆的借书流程，处理预约借书、立即借书、归还图书、以及查询当前借阅状态的操作，维护读者与图书的借阅关系，并且保证每个操作按照时间顺序正确执行。

- 大模拟。考察选手编码能力、阅读理解能力 (标程代码总共 80 行)。
- 考虑分别维护五个函数：
`getReserve()`, `getBorrow()`, `getReturn()`, `getQuery()`, `dayInit()`，分别代表四个操作和每一天一开始的初始化，即处理预约在当天借书的读者。
- 箱子中的取书还书过程可以用栈进行模拟，预约在某一天的这些读者可以用数组或者队列存储，每个读者对应一个状态，可用数组/哈希表/红黑树存储。
- 这题数据范围设置过小，目的是允许选手可以用较为暴力的手段维护该过程，主要考察选手代码逻辑能力，而没有过多考察数据结构能力。

A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看

文神的序列 I

题目大意

给定一个长度为 n 的正整数序列，每次操作可以选择两个不同元素进行按位与运算并更新其中一个元素。判断是否能够通过至多 n 次操作将序列中的所有元素变为 0。

文神的序列 II

结论

若存在一组 i, j ($i \neq j; 1 \leq i, j \leq n$) 使得 $a_i \& a_j = 0$, 则经过至多 n 次操作后, 能将序列中的所有数字改为 0; 否则不能。

文神的序列 III

充分性证明.

假设找到 i, j ($i \neq j; 1 \leq i, j \leq n$) 使得 $a_i \& a_j = 0$. 因此, 我们可以将序列中其他所有元素依次与 a_i (现在为 0) 进行 $\&$ 操作, 从而将它们全部变为 0。由于每次操作可以选择两个不同的元素, 因此最多需要 $n-1$ 次操作 (除了 a_i 已经是 0 外, 其他 $n-1$ 个元素各需要一次操作)。



必要性证明.

反证法. 假设无法找到 i, j ($i \leq j; 1 \leq i, j \leq n$) 使得 $a_i \& a_j = 0$. 那么第一次操作必然无法构造出一个 a_i 使得 $a_i = 0$, 剩下的 $n-1$ 次操作必然无法使得序列中的 n 个元素全部变成 0。



A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看

低谷 I

题目大意

在给定长度为 n 的数组中，通过最多进行 k 次任意元素减一操作，求操作后数组中可能出现的最大低谷数。

对于数组 a ，位置 i 是低谷当且仅当 $2 \leq i \leq n-1$ 且满足 $a_i < a_{i-1}$ 和 $a_i < a_{i+1}$ 。

- 低谷之间不能相邻。
- 若 i 是 a 的一个低谷，且若要 i 成为 a 的一个低谷，则无需操作。
- 若 i 不是 a 的一个低谷，且若要 i 成为 a 的一个低谷，则需要将其变为相邻元素的最小值-1。

因此，若要让 i 成为 a 的一个低谷，则需要耗费

$$\max(0, a_i - \min(a_{i-1}, a_{i+1}) + 1) \quad (1)$$

次操作。

低谷 II

由此可以得到一个长度为 $n - 2$ 的数组 v , 其中第 i 个元素表示使 a 中第 $i + 1$ 个元素成为低谷需要的最少次数。由此可以转化为经典的 dp 问题, 即求相邻元素不能同时取, 且总和小于等于 k 的情况下, 能取的元素的最大值。



低谷 III

转化后的问题

相邻元素不能同时取，且总和小于等于 k 的情况下，能取的元素的最大个数。

看到数据范围，可以估计这是一个 $O(n^2)$ 的 dp。设 dp 数组为 $dp[n][n]$ 。

- $dp[i][m]$ 表示在前 i 个位置中选择 m 个不相邻的低谷所需的最小总操作次数。
- 由于不能相邻，因此状态转移时考虑的是 $dp[i-2][m-1]$ 转移到 $dp[i][m]$ ，有：

$$dp[i][m] = \min(dp[i-2][m-1] + v_i, dp[i-1][m]) \quad (2)$$

- 最后在 dp 数组中找到最大的 m ，使得 $dp[n-1][m] \leq k$ ，这个 m 即为答案。
- 本题 dp 状态还可以有其它设计，本解法也可以有空间时间优化，比如考虑答案最多只有 $\lfloor \frac{n-1}{2} \rfloor$ 个，可以将 dp 第二维压缩更少。但是不优化也可以轻松通过本题。

低谷 IV

进阶

此题有困难版本： $n \leq 10^5$ 。

反悔贪心。显然我们可以预处理出来将每个位置变成低谷所需代价。
由于低谷的特性，显然我们选择的两个低谷不能相邻，那么题目就变成了：在序列中选择不相邻的若干个位置 S ，使得 $\sum_{i \in S} cost_i \leq k$ ，求最大的 k 。

看到相邻位置不能选择，考虑反悔贪心中的双向链表反悔自动机，可以参考例题：
P1484 种树 - 洛谷

低谷 V

- ① 当选择一个点 u 的时候，由于相邻节点无法选择，我们选择删除左右节点 L, R
- ② 如果点 u 不是左端点和右端点，我们需要将 val_u 重新赋值成 $val_L + val_R - val_u$ ，并塞入堆中，实现反悔贪心

由于每个点的进入必定伴随着其他点的删除，因此堆中最多进入 n 个点，时间复杂度 $O(n \log n)$

A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看

伤害最大化 I

题目大意

在 n 个时间刻内，通过合理安排 4 种技能的释放顺序，求对敌人造成的最大伤害。技能之间有冷却时间限制，并且每个时间刻必须释放一个技能，给定怒气积累值 a 和每个时间刻可以造成的不同伤害值 d_i ，需要计算出通过最优操作能造成的最大伤害值。

- 注意到 n 的取值范围很小，只有 18.
- 考虑暴力搜索，dfs+ 剪枝。

伤害最大化 II

- 通过递归进行暴力搜索，枚举每个时间刻释放哪个技能，考虑每个技能的冷却时间。用递归的深度表示时间刻，每个时间刻都有 4 种选择。
- 使用一个数组维护技能的冷却时间，若某个技能未冷却，则不能选择该技能。每次技能释放后，将相应技能的冷却时间设置为 2，并在后续时间刻中逐步减少冷却时间。
- 在递归过程中，维护当前累积伤害值，搜索所有可能的技能组合，最终记录下最大伤害值。

A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看

猫猫虫困境 II

题目大意

题目要求计算每只猫猫虫从它所在的位置到达矩阵出口 (位于坐标 $(g, 1)$) 的最少时间。猫猫虫可以通过向左、向右移动, 或者使用传送门来加速移动。需要找到从猫猫虫所在的位置到出口的最优路径, 输出每只猫猫虫的最短到达时间。

- 本题有多种解法, 此处主要讲两种解法: 第一种贪心解法, 第二种运用图论最短路解法。

猫猫虫困境 II II

解法一 (贪心)

先预处理每个传送门到出口的路径长度，然后再考虑每个猫猫虫左侧或者右侧最近的一个传送门的长度，以及该猫猫虫直接到出口的长度。最后

这个猫猫虫的最短到达时间 $= \min(\text{猫猫虫到达左侧传送门时间} + \text{左侧传送门到达出口时间}, \text{猫猫虫到达右侧传送门时间} + \text{右侧传送门到达出口时间}, \text{猫猫虫直接到达出口时间})$

找传送门的过程可以用一次扫描，也可以用二分查找，还可以用 c++ 中自带二分查找功能的 `lower_bound()` 函数查找，在此不作赘述。注意特判最边缘的猫猫虫。

猫猫虫困境 II III

解法二 (Dijkstra 最短路)

把每个猫猫虫, 传送门和困境出口均看作一个节点, 把每个相邻传送门之间用边连接, 每个猫猫虫与其最近的左右传送门之间用边连接, 相邻传送门间边权为 1, 其它边权均为两节点之间距离, 以困境出口为起点, 跑 Dijkstra 最短路。
该解法同样需要找到猫猫虫左右最近邻的两个传送门。

A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看

奥运会

题目大意

预测 L 年到 R 年间有几场奥运会。

- 不难发现年数为偶数的年份都会按计划都会开展一次奥运会。输出 L 到 R 中有多少个偶数即可。

A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看



文神的比赛

题目大意

题目要求在 n 个时间刻内，通过合理安排 4 种技能的释放顺序，求对敌人造成的最大伤害。技能之间有冷却时间限制，并且每个时间刻必须释放一个技能，给定怒气积累值 a 和每个时间刻可以造成的不同伤害值 d_i ，需要计算出通过最优操作能造成的最大伤害值。

- 首先预处理阶乘数组，统计每种长度的个数，由于不同长度之间的相对排名固定，因此相互独立。
- 根据乘法原理计算所有长度为 i 的字符串个数的阶乘的乘积 (i 从 1 到 26)，即为最终答案。
- 定义 c_i 为长度为 i 的字符串个数，则答案为 $\prod_{i=1}^{26} (c_i!)$ 。
- 注意计算过程中可能会超出 `int` 范围，需要进行取模运算防止爆 `int`。

A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看

智力博弈 I

题目大意

在双方玩家采取最优策略下，给定正整数 n 和 k ，计算所有由 1 到 n 每个数恰好出现两次组成的长度为 $2n$ 的数组中，经过 k 轮轮流删除最左或最右元素操作后，剩余数组中不存在任何只出现一次的元素的不同数组数量，并将结果对 $10^9 + 7$ 取模。

智力博弈 II

- 首先考虑后手必定要在博弈过程中制衡先手，来稳定局面，即：若先手选一侧，则后手选另一侧。
- 因此可以形成一个大概的局面：该局面分为两个部分：两侧的数，内部的数。每个部分内每个数都必须在该部分恰好出现 2 次。



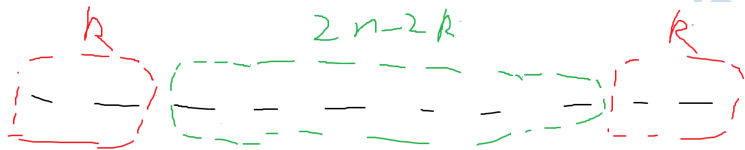
智力博弈 III

现在考虑具体情况。

情况一

在 k 轮里每轮先手选一侧，后手选另一侧，则 k 轮后必然会造成如图所示的局面 (只剩绿色部分)，其中红色区域代表两侧的数，绿色区域代表内部的数。该方案数为：

$$ans_1 = C_n^k \cdot \frac{(2k)!(2n-2k)!}{2^n} \quad (3)$$



智力博弈 IV

情况二

在 $k-1$ 轮里每轮仍然先手选一侧，后手选另一侧，但在第 k 轮中选与先手同侧的元素。则 k 轮后必然会造成如图所示的局面 (剩绿色部分，在蓝色部分和紫色部分中有一个部分剩余)，其中红色区域代表两侧的数，绿色区域代表内部的数，蓝色和紫色部分代表边缘的数。该方案数为：

$$ans_2 = A_n^2 \cdot C_{n-2}^{k-1} \cdot \frac{(2k-2)!(2n-2k-2)!}{2^{n-2}} \quad (4)$$



智力博弈 V

特殊情况

当 $k = 0$ 或 $k = n$ 时，该数组任意排列都满足题意。该方案数为：

$$ans^* = \frac{(2n)!}{2^n} \quad (5)$$

智力博弈 VI

则最后的答案为

$$ans = \begin{cases} ans_1 + ans_2, k \notin \{0, n\} \\ ans^*, otherwise \end{cases} \quad (6)$$

$$= \begin{cases} C_n^k \cdot \frac{(2k)!(2n-2k)!}{2^n} + A_n^2 \cdot C_{n-2}^{k-1} \cdot \frac{(2k-2)!(2n-2k-2)!}{2^{n-2}}, k \notin \{0, n\} \\ \frac{(2n)!}{2^n}, otherwise \end{cases} \quad (7)$$

计算该表达式即可。并且由于有多个测试用例，因此需要注意预处理 10^6 以内的阶乘。

A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看

星穹列车 I

题目大意

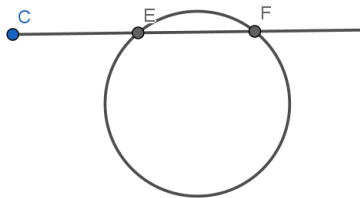
题目要求计算二维平面上一条射线经过若干个圆形区域的总长度，重叠区域不重复计算。

星穹列车 II

考虑单个圆形区域在于一条直线的交点。设圆心坐标 (x_d, y_d) , 半径 r_d , 直线为 $y = y_0$, 则两个交点坐标分别为

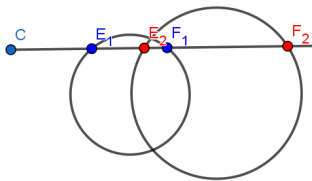
$$E \left(x_d - \sqrt{r_d^2 - (y_0 - y_d)^2}, y_0 \right) \quad \text{和} \quad F \left(x_d + \sqrt{r_d^2 - (y_0 - y_d)^2}, y_0 \right)$$

这两个交点 EF 坐标之间的距离即为有效长度。



星穹列车 III

现在考虑多个圆形区域。由于每个圆形区域可能有重叠，故每个圆形区域对答案造成的贡献不能简单相加。同时，注意到每个圆形区域对答案的贡献实际上可以看作一条线段，而对于多个圆形区域而言，每条线段重叠部分不额外计算。因此问题可以转化成：给定若干个区间 $[l_i, r_i]$ ，求它们合并之后的长度。



星穹列车 IV

转化后的问题

给定若干个区间 $[l_i, r_i]$ ，求它们合并之后的长度。

考虑把这些区间以 l_i 为第一关键字， r_i 为第二关键字排序，排序后从左至右遍历这些区间，与前一个区间作比较，并考虑存在的三种情况：相离，相交，包含。

- 对于相离而言，这两个区间是独立的，直接分开算贡献即可；
- 对于相交而言，相当于扩展前一个区间的右侧，因此将前一个区间右端点移到当前区间的右端点，从而将贡献进行合并；
- 对应包含而言，相当于这个区间被吞掉了，不对答案造成贡献。



星穹列车 V

现在再考虑把直线变成射线。由于射线左侧的部分不对答案造成贡献，因此一开始将每个区间的左右端点横坐标与射线的左端点横坐标取 \max 即可。



A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看

冰红茶 I

题目大意

在 $n \times m$ 的网格中放置 k 瓶冰红茶，使得 lzt 从每一行中随机选择一瓶冰红茶时，所支付的钱的期望最大。需要根据冰红茶的价格，将冰红茶合理地放置在网格中，构造出最终的网格。

- 本题为构造题，考察选手贪心思维和贪心直觉，以及构造技巧。
- 构造方案：让价格最贵的冰红茶尽可能放置在冰红茶最少的对应行里。

冰红茶 II

证明.

对于当前布局，其中两列，一列有 a 个，一列有 b 个 ($a < b$). 首先从贡献法的角度，每个数 x 对答案造成的贡献是 $\frac{x}{a}$ 或者 $\frac{x}{b}$. 那么肯定 a 个数是这 $a+b$ 个数中最大的 a 个， b 个的是最小的。

下面进行调整，因为要保证原先 a 那列最小值也要大于 b 那列的最大值，所以依次将大列的最小值移到小列，这保证大列的平均值也上升了，小列的平均值也上升了。 □

A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看

数组交换

题目大意

对一个奇数长度的数组进行多轮操作，每轮操作中选择相邻的两个元素交换，并删除数组开头的两个元素，直到数组只剩下一个元素。要求设计一种策略使得最后剩下的元素尽可能大，求出这个最大值。

- 由于进行了 $\frac{n-1}{2}$ 轮，因此从后往前数的 $\frac{n+1}{2}$ 个元素都可以成为最后一个元素。输出这些元素的最大值即可。

A 数组

B 借阅图书

C 文神的序列

D 低谷

E 伤害最大化

F 猫猫虫困境 II

G 奥运会

H 文神的比赛

I 智力博弈

J 星穹列车

K 冰红茶

L 数组交换

M 猜猜看

猜猜看

题目大意

在 1 到 10^9 范围内，通过最多 70 次操作，包括询问一个数是否能被给定的 a 整除、对当前数减去 b ，以及直接猜测初始数 x_0 的交互操作，猜出 lzt 心中所想的正整数 x_0 。

- 交互题。看到 70 次操作，可以想到本题操作次数是 $2\log n$ 级别的。
- 考虑 2,4,8,16,... 一直往上问，如果发现出现 No，即不能被当前询问的值整除，设当前询问的值为 x ，则说明减去 $\frac{x}{2}$ 之后肯定能被 x 整除。数学归纳法易证。
- 注意特判大于等于 2^{29} 的数，来确定最后一步是剩下 0 还是剩下 2^{29} ，如额外询问是否能被 3 整除，如果不行，则说明最后的数是 2^{29} ，否则说明最后的数是 0。