

2026 年牛客寒假算法基础训练营第二场题解

牛客竞赛
Bingbong

2026 年 2 月 5 日

写在前面

这场本来还有 very hard，但是内测反馈过难，所以删除了一个，从榜单上来看整体符合预期（第二场的高手好多喵）。

对于个别原了的题，我们深感抱歉，内测都没检查出来，但是由于数据或者个别原因，对榜单产生影响基本可以忽略。

对于 I 题超时的原因是因为输出了 endl，评价是大赛被卡，不如寒假营被卡。

比较妙的是，每个人对于每个题目的难度体感差距都很大。

感谢页

致谢

平台管理: 王清楚

协调员: WIDA、emofunx

验题人员

7_divided_by_3、BlackCat__、bandiaoz、dremlory、
LFS__wait、Silencer76、TRfirst、Tobo、xhytom、zankizero、
超高校级的飞舞 acmer、撤云、牛客 655900812 号、
三角粑粑、神崎兰子、嚶嚶世界第一可爱

感谢所有人员为题面修正和数据加强所做出的宝贵贡献，以及参与正式比赛的各位选手。

难度预估

难度仅供参考，同等难度下，难度从左到右递增。

签到：A 比赛安排 I 01 回文

简单：F $x?y?n!$ B NCPC H 权值计算

中等：E 01 矩阵 J 终于再见

中等偏难：C 炮火轰炸 D 数字积木

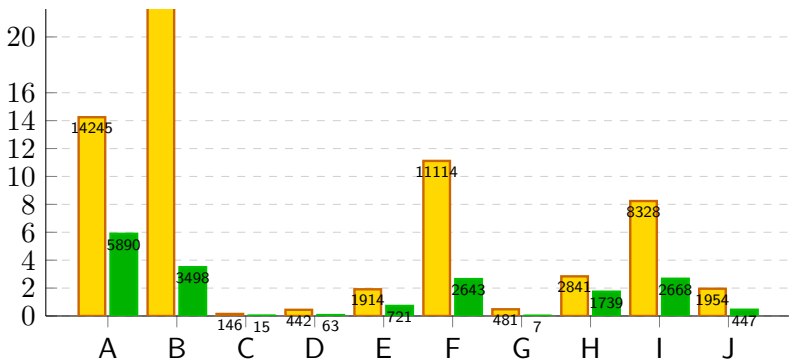
难：G 宝藏拾取

提交与通过统计

各题提交与通过数量统计

■ 总提交数量 ■ 通过数量

数量 ($\times 10^3$)



A. 比赛安排

题目描述

T 组询问，每组询问给定小白月赛，练习赛，挑战赛的场次 a, b, c ，要求排期使得对于任意连续三场的比赛类型都不相同，求是否可能做到。

数据范围: $1 \leq T \leq 10^4, 1 \leq a, b, c \leq 10^9$

A. 比赛安排

解题思路

对于任意连续的 3 场比赛都不同，故三场可为一个固定周期，最后只能其中一种比赛类型或者两种比赛类型多一场，故最多比赛场次和最少比赛场次的差值不能超过 1。

- 时间复杂度: $\mathcal{O}(T)$

B.NCPC

题目描述

给定 n 个整数，每次可以选定两个整数，若两个数字不等，留下较大值，否则全部删除，问对于每个整数能否是最后剩下的数字。

数据范围: $1 \leq n \leq 2 \times 10^5$

B.NCPC

解题思路

对于一个最大值的数字，需要删除其他最大值数字，且该数字不能删除，故最大值个数必须为奇数个，该数字便可以做到，否则做不到。

对于一个非最大值的数字，需要删除更大值数字，所以最大值的个数必须为偶数个（最大值先删除其他数字，再相互抵消），该数字便可以做到，否则做不到。

- 时间复杂度: $\mathcal{O}(n)$

C. 炮火轰炸

题目描述

给定 n 个炮火点，每个炮火点 (x, y) 之间满足八联通性质则认为连接在一起，给定 q 次询问 (a, b) ，回答该点是否在某个炮火圈区域内。

数据范围: $4 \leq n \leq 5 \times 10^4, 1 \leq q \leq 10^5, 1 \leq x, y, a, b \leq 10^6$

C. 炮火轰炸

解题思路

我们考虑把一个炮火点附近的八个点全部存起来，对于每个点找四个方向可以达到的最远点，然后通过并查集合并这些点。此时设置一个超级源点 $(-1, -1)$ 认为是可以走出炮火圈的点，当四个方向不存在最远可以到达的点，合并时与该超级源点合并。

找的过程可以将障碍点 (x, y) 放进 x 和 y 所对应的二维容器内，通过二分查找。

最后对于每个询问，只需要看和超级源点是否在一个连通集合内即可。

- 时间复杂度: $\mathcal{O}(4 \times (8 \times n + q) \times \log(8 \times n))$
- 内测也有非常省常数的做法，做法类似，此处不再展开。

D. 数字积木

题目描述

给定一棵 n 个结点的树，树上的权值是 $0 \sim n-1$ 的组成的集合，求树上所有连通子图的 MEX 之和，结果对 $10^9 + 7$ 取模，
数据范围： $1 \leq n \leq 2 \times 10^5$

D. 数字积木

解题思路

首先常规思路是枚举 MEX，然后通过暴力合并 $0 \sim \text{MEX}-1$ 的结点，同时标记结点权值为 MEX 的结点是不可选的，然后通过树形 dp 可以计算得到所求答案，该做法的时间复杂度是 $\mathcal{O}(n^2)$ ，无法通过。

D. 数字积木

解题思路

通过以上的暴力手段，我们发现直接枚举很难做，会发现算权值过程中本身是计算连通子图个数的问题，考虑一下计算权值为 MEX 的连通子图都计算了哪些更小的连通子图，这些更小的连通子图又贡献给了哪些大的连通子图，从该切入点思考就不难发现，问题转化为了：当前权值为 MEX 的所有结点合并的连通子图被计算了多少次，这样就无需去标记 a_{MEX} 的结点再树形 dp，可通过 dp 预处理包含节点 i 的连通块的个数，在合并连通块时通过增删解决。最后把 $1 \sim n$ 的答案统计累加即可。特别的，在赛前一天，我们删除了规避逆元除 0 的情况，所以在统计的时候需要额外计算连通子图个数乘了多少次 0，将除法改成减法。

- 时间复杂度: $\mathcal{O}(n \times \log(n))$

E.01 矩阵

题目描述

给定一个整数 n ，要求构造一个 $n \times n$ 的 01 矩阵同时满足下列条件：

1. 每一行数字的和构成 $\{0, 1, 2, 3, \dots, n-1\}$ 的数字集合。
2. 每一列数字的和构成 $\{0, 1, 2, 3, \dots, n-1\}$ 的数字集合。
3. 数字连通块恰好有 n 个。

数据范围： $1 \leq n \leq 1000$

E.01 矩阵

解题思路

内测中给出了非常多的做法，这里给出两种。

做法一：

首先构造第一步可以完成 1, 2 两个要求，如矩阵一下所示，接着需要不破坏原有行列构造数字集合的前提下拆分连通块，考虑偶数行的 1 变成列，从右边依次往左放置即可，如矩阵二所示。

例如： $n = 6$

111110

111100

111000

110000

100000

000000

矩阵一

111110

000000

111010

000010

101010

001010

矩阵二

E.01 矩阵

做法二:

例如: $n = 6$

```
000000
011111
010000
010111
010100
010101
```

考虑上图的矩阵, 行从 $0 \sim n-1$, 列从 $0 \sim n-1$, 对于 (i, j) 的答案为 $\min(i, j) \text{ AND } 1$ 。

AND: 位运算中的按位与。

- 时间复杂度: $\mathcal{O}(n^2)$

F.x?y?n!

题目描述

给定一个整数 n , 构造两个不相等的整数 x, y 满足 $\gcd(x, y) = n$ 且 $x \oplus y$ 最小。

数据范围: $1 \leq T \leq 10^4, 1 \leq n \leq 10^6$

F.x?y?n!

解题思路

因为 $|x - y| \leq x \oplus y$, $\gcd(x, y) = n$ 且 $x \neq y$, 所以 x 和 y 是 n 的不同倍数。

这意味着 $|x - y|$ 至少是 n 的 1 倍。

因为 $|x - y| \leq x \oplus y$

$n \leq |x - y| \leq x \oplus y$

所以 $x \oplus y$ 的最小值是 n 。

F.x?y?n!

解题思路

接着考虑如何构造出两个整数 x, y 满足上述条件。

这两个整数需要满足在二进制位上与 n 无交集，异或结果便等于 n 。

而对于 $\gcd(x, y) = n$ ，我们只需要考虑乘以两个连续的整数即可。令 $k = 31$ ，这样便可以得出整数 $x = n \times 2^k$ ， $y = n \times (2^k + 1)$ 。

- 时间复杂度: $\mathcal{O}(T)$

G. 宝藏拾取

题目描述

给定一个长度为 n 的宝藏数组 a_1, a_2, \dots, a_n 。Bingbong 从起点 s 出发，向索引增大的方向移动。对于经过的每个位置 i ：

- 若是起点 $i = s$ ，则直接获得 a_i ；
- 若 $i > s$ ，当且仅当当前持有的宝藏总和 $sum \geq a_i$ 时，可以获得 a_i ；
- 每个位置的宝藏只能在其被经过时领取一次，且移动过程不可逆。

现在你需要对于任意的 $1 \leq s \leq n$ ，回答以 s 为起点可以获得的最多宝藏数量，每次回答相互独立，互不影响。

数据范围： $1 \leq n \leq 2 \times 10^6, 1 \leq a_i \leq 10^9$

G. 宝藏拾取

解题思路

对于每个点维护一个 val_s ，表示从 s 走到当前位置 i 的答案。对于旧起点 $s (s < i)$ ，如果满足 $a_i \leq val_s$ 的，就给它加上。所以每一步 i 的操作就是：把所有 $a_i \leq val_s$ 的起点整体加上 a_i ，再插入一个值为 a_i 的新起点。按照阈值 a_i 将集合分成两个部分 L 和 R ，同时记录每个 val 的下标 idx 。 L 满足 $a_i > val_s$ ， R 满足 $a_i \leq val_s$ 。故对 R 做整体区间加的操作，插入新元素 $val = a_i, idx = i$ ，再合并回去，此步用 `treap` 实现即可。

- 时间复杂度: $\mathcal{O}(n \times \log(n))$

H. 权值计算

题目描述

定义一个数组的权值为其所有前缀中不同数字的个数，求所有子数组的权值之和。

数据范围: $1 \leq n \leq 10^5, 1 \leq a_i \leq 10^9$

H. 权值计算

解题思路

正难则反，考虑一个数字能给多少个子数组贡献答案以及贡献的大小。在算 a_i 时，考虑的子数组必然是包含 i 这个位置的，所以计算过程只会被上一个 a_i 的位置 j 影响，影响的区间左端点个数为 $i-j$ 个，而以 $l(j < l \leq i)$ 为左端点， $r(i \leq r \leq n)$ 为右端点的区间贡献多少次呢？考虑如下：

$r = i$ 时: 1 次 $r = i+1$ 时: 2 次 $r = i+2$ 时: 3 次

...

$r = n$ 时: $n-i+1$ 次

归纳之后便可以得出答案为 $(i-j) \times \frac{(1+n-i+1) \times (n-i+1)}{2}$ ，最后对每个位置都这样求和即可。

● 时间复杂度: $\mathcal{O}(n \times \log(n))$

I.01 回文

题目描述

给定 $n \times m$ 的 01 矩阵，询问任意一个位置开始拼接字符，任意非起始位置结束，能否拼接得到一个回文串。

数据范围: $1 \leq n \times m \leq 10^6$

I.01 回文

解题思路

不断考虑下一步，对于起始位置字符为 c ，若下一个位置无法拼接得到回文串，则是 $c \oplus 1$ ，不断重复以上过程。

若最后可以存在一个字符 c ，则可以拼接得到回文串，说明只要矩阵中存在两个相同数字，便可以做到题意所求。否则无法做到。

- 时间复杂度: $\mathcal{O}(n \times m)$

J. 终于再见

题目描述

n 个城市，编号为 1 到 n ，城市之间通过 m 条无向道路连接，每条道路的边权为 1，对于一个城市的繁华度是该城市所连接的道路数量。

将每个城市的繁华度列出并去重后，从小到大排序得到 a_1, a_2, \dots, a_k ，现在将所有繁华度为 a_i ($1 \leq i \leq k$) 的城市称作 i 级城市。

现在需要回答对于任意编号为 x ($1 \leq x \leq n$) 的城市，从该城市出发，前往比该城市级别更高的城市的最短路径长度，或报告无法到达。

注意：此处 k 级城市为最高级城市。数据范围：

$$1 \leq n \leq 2 \times 10^5, 1 \leq m \leq 2 \times 10^5$$

J. 终于再见

解题思路

正着对每个点去寻找更高级别城市，时间复杂度会退化为 $\mathcal{O}(n \times (n + m))$ 。

不妨反向思考，从最高级城市出发，计算到其他城市的最短距离即可。可以发现边最多 m 条，构造出的不同等级城市最多 \sqrt{m} 个，所以对 \sqrt{m} 个不同等级的城市做多源 BFS 即可完成。

- 时间复杂度: $\mathcal{O}(\sqrt{m} \times (n + m))$

祝大家 2026 年新年快乐喵~