

# 牛客寒假算法基础集训营1

By attack204/fastle



牛客网  
NOWCODER

# I\小a的计算器

---

考点：模拟

本场的良心签到题。由于我们得到了最终的数，而且整个操作都是可逆的，因此直接倒序模拟即可

# I/a与"204"

考点：贪心 模拟

输入的序列其实用处不大，因为最终不需要输出方案，我们只需要记录下2/0/4分别出现的次数即可  
一个显然的构造策略是首先放置4, 0, 4, 0，直到其中一个用光。

接下来如果4多余，那么可以按4, 0, 4, 0, ..., 4, 2, 4, 2, ... (先4后2)的方法构造

如果0多余，可以按照4, 0, 4, 0 ... 4, 0, 2, 0, 2 ... (先2后0)的方法构造

std中的a数组展示了其中一种最优的构造方案

实际上此题还可以推广到更一般的情况，也就是第一个位置放最大的，第二个位置放最小的，第三个位置放第二大的以此类推，这种思路写起来也会更简单一些

# I/a与星际探索

考点：排序 背包dp

首先可以按照 $p_i$ 从大到小排序，这样每个点能走到的位置就变成了一段连续的后缀，我们直接在新序列上做背包，用 $f[i][j]$ 表示到达第 $i$ 个位置，当前耐久度为 $j$ 是否可行

转移的时候分两种情况讨论：

1. 到达该点
2. 不到达该点

对 $n$ 位置特殊判断一下，最后枚举 $n$ 所在位置 $f$ 数组的第二维，判断一下即可

注意一个小细节：dp数组的第二维不能只开到3000

时间复杂度： $O(np)$

# 小a与黄金街道

考点：欧拉函数 / 容斥

首先不难发现，若 $\gcd(n, x) = 1$ ，那么 $\gcd(n, n - x)$ 一定等于1

证明：反证法，假设 $\gcd(n, x) = 1$ 且 $\gcd(n, n - x) = k$ ，

若  $\begin{cases} n = k \cdot p \\ n - x = k \cdot q \end{cases}$ ，那么 $x = k \cdot p - k \cdot q = k \cdot (p - q)$ ，那么 $n, x$ 都有 $k$ 这个约数，与假设冲突

同时不难发现，若小a在某个位置获得了 $k^x$ 的贡献，

那么一定存在一个位置会获得 $k^{n-x}$ 的贡献，而且两个人的贡献是相同的！

把小a的贡献单独写出来即为 $A * k^a * k^{n-a} * k^b * k^{n-b} \dots = A * k^{Rn}$

考虑如何快速得到 $R$ 的值，观察题目描述不难发现，能产生答案的数一定是与 $n$ 互质的数，这与 $\phi$ 函数的定义是相同的！

又因为前 $n$ 个数的欧拉函数之和为 $\frac{n}{2} \varphi(n)$  (当然不知道这个结论也没关系，可以直接容斥算约数的贡献)

因此最终的答案为 $(A + B) * K^{\frac{\phi(n)}{2}n}$

欧拉函数可以 $O(\sqrt{n})$ 计算

# I\小a的轰炸游戏

---

考点：差分

由于 $n, m$ 只有1000，且没有修改操作。

那么考虑二维差分。直接上可以不太好实现，我们菱形分为上下两部分

对于上半部分来说：把标记分为两种(向左下放/向右下放)，首先在最顶端打上+1的标记，再分别在最下端的两边打上-1的标记

每次分别下放即可

注意不要越界，一个不错的解决方法是对所有操作加一个偏移量，最后只统计原矩形的贡献

时间复杂度： $O(q + n^2)$

# I\小a的子序列

考点：dp

考虑直接dp

设 $f[i][j]$ 表示到第 $i$ 个位置，以 $j$ 结尾的所有合法子序列的值的和

$g[i][j]$ 表示前 $i$ 个位置，最大的数为 $j$ 的所有合法子序列的中所有数乘积的和（乘积与值的区别在于是否统计最大元素）

转移的时候显然 $f[i][j] = g[i-1][j-1]$

$g$ 的转移可以采用简单的容斥，先加上所有的方案，再减去重复的，类似求二维前缀和的过程

std的做法是分成三部分

1. 由前 $i-1$ 个数组成，最大值为 $j$ 的贡献，这部分是 $g[i-1][j]$
2. 由前 $i$ 个数组成，最大值为 $j-1$ 贡献，为 $g[i][j-1]$
3. 钦定选了第 $i$ 个数，最大值为 $j$ 的贡献，为 $f[i][j] * j$

我们重复计算了由前 $i-1$ 个数组成，最大值为 $j-1$ 的部分，所以要减去

具体来说 $g[i][j] = g[i][j-1] + f[i][j] * j + g[i-1][j] - g[i-1][j-1]$

最后输出  $\sum_{i=1}^n \sum_{j=1}^V f[i][j]$  即可

复杂度： $O(nV)$

# I/a的排列

考点：模拟 性质分析

首先，一个区间 $[l, r]$ 是连续区间当且仅当 $\max_{l,r} - \min_{l,r} = r - l$

我们维护出每个数出现的位置，即 $pos_x$ 表示数 $x$ 的位置，考虑每次迭带更新答案。

维护四个变量 $l, r, mx, mn$ 分别表示当前区间的左右端点，最大最小值

首先找到 $[pos_x, pos_y]$ 中的每个数位置的最大最小值来找到下一轮的 $l, r$

接下来不断扩充当前的 $l, r$ ，扩充的同时更新 $mx, mn$ ，再不断用新的 $mx, mn$ 更新 $l, r$ ，直到找到一段连续区间

当找到第一个满足条件的区间即为长度最小的区间，实际上长度最小的区间的左右端点只有一种方案，因为两段相交的连续区间的交集也是连续区间

时间复杂度： $O(n)$



# I小a的学期

考点：组合数 组合数取模

我们可以把不写作业看做+1，写作业看做-1

那么，转化后的题意为：

长度为 $2n$ 的序列， $n$ 个-1， $n$ 个+1，问不存在一个前缀值 $\geq K$ 的方案数有多少

首先，若存在一个不合法的前缀 $\geq K$ ，那么一定存在一个位置 $= K$ 。

$K = 1$ 显然是经典的Catalan数

考虑其证明过程

若数列不合法，则一定存在一个位置，在这之前有 $m + k$ 个+1， $m$ 个-1

我们把之后的1变为-1，-1变为1

则该序列含有 $n + K$ 个+1， $n - K$ 个-1

也就是说，一种不合法的方案进行上述转化后，一定唯一对应了长度为 $2n$ 且含 $n + k$ 个1的序列

那么转化后序列的方案数为 $C_{2n}^{n+k}$ ，用总的 $C_{2n}^n$ 减去即可

注意模数可能不为素数，我们可以考虑每个质因子的贡献，也就是把公式 $C_n^m = \frac{n!}{m! \cdot (n-m)!}$ 中的阶乘展开，同时用

线性筛预处理出每个数的最小质因子，对于每个数将其分解成质数乘积的形式，最后统计每个质数的贡献即可

复杂度： $O(n \log n)$

# I/a与军团模拟器

考点：启发式合并

我们发现给定树上的所有节点的颜色，就可以 $O(n)$ 求得军团个数。方法是进行dfs，如果一个结点和它父亲节点颜色不同，答案就++，然后暴力更改暴力统计答案

我们可以优化一下上面那个暴力，用vector记录下每种颜色存在的位置，然后枚举要更改的颜色位置，每个位置原来对于答案的贡献是和他相邻不同色的节点个数，我们减去原来的答案加上更改后的答案就得到了当前的答案。然后再暴力合并vector就行了。

然而这样还是无法通过。

我们考虑优化合并操作，我们发现把x变成y和把y变成x得到的答案是相同的，所以我们在合并的时候以及更改颜色的时候都是把颜色少的改成颜色多的，并且维护每种颜色实际的vector是哪个，然后就相当于启发式合并。

每个点的复杂度是他的度数乘以它被统计答案的次数，最差情况下每个点也只会统计 $\log N$ 次答案，最终复杂度为 $O(N \log N)$

# I/a的集合

---

考点：线段树 set

直接做可能不好做，因为我们不能直接维护每个集合中的数。

那么换个角度，我们对于每个加入过的数开一个set来维护其出现的区间(显然最多只有 $m$ 个)

注意这些区间都是不相交的

加入区间的时候可以直接二分出包含其左右端点的区间直接合并，删除操作同理。

所求的值可以用线段树打标记维护

由于每个数在set中最多被插入删除一次，同时在线段树上也最多会被加入删除一次，

因此总的时间复杂度为 $O(n\log n)$

# Istd

友情提示：std的代码是本着尽量不出错的原则写的，所以无论是代码量还是常数都有很大的优化空间，因此还是建议大家理清思路后自己去写

以下链接密码均为“nowcoder”

- A、小a的计算器 —标程：<https://pasteme.cn/3474>
- B、小a与"204 " —标程：<https://pasteme.cn/3475>
- C、小a与星际探索 —标程：<https://pasteme.cn/3476>
- D、小a与黄金街道 —标程：<https://pasteme.cn/3477>
- E、小a的轰炸游戏 —标程：<https://pasteme.cn/3478>
- F、小a的子序列 —标程：<https://pasteme.cn/3479>
- G、小a的排列 —标程：<https://pasteme.cn/3480>
- H、小a的学期 —标程：<https://pasteme.cn/3481>
- I、小a的集合 —标程：<https://pasteme.cn/3482>
- J、小a与军团模拟器 —标程：<https://pasteme.cn/3483>

# I 吐槽

---

关于这场比赛，首先跟大家说声抱歉。主要是因为更换D题数据这件事，由于扩展欧拉定理这个东西实在太毒瘤，又为了让大家多一点时间做后面的题，所以删除了大数据。非常抱歉。

希望大家通过本场比赛能够学习到更多的算法/解题方法，帮助自己更好的成长！

# Thanks