

# 浙江农林大学第二十六届程序设计竞赛

## 暨百度之星选拔赛

2024 年 6 月 2 日

## 乘法 (easy)

- 这题是签到，记得开 long long。

## Love and Pease

- 简单的模拟。
- 只要根据题目意思一场比赛一场比赛处理，到 100 分或所有比赛判完就停止即可。

## 来不及了，就这个吧

- 双指针，可以先尽可能地选第一个题单，然后每次第一个题单少选一个，看看第二个题单能选几个，复杂度  $O(n)$ 。
- 如果贪心的看序号小的优先选肯定是错的。
- 当然也可以二分，复杂度更高一些但也能做。

## 活塞高手

- 诈骗，可以发现其实题目中的推动上限没有什么意义。只要先推后面的再推前面的就可以保证推得动。
- 发现性质后模拟即可。
- 考虑活塞对答案的贡献，由于数据限制了两端一定会有活塞存在，于是不考虑最后一个活塞，将前面的每一个活塞的贡献视作 2（自己一个位置，推动一个位置）将答案初始化为  $-1$  若当前的活塞位置大于答案则直接更新答案即可。复杂度为  $O(n)$ 。

## 开倒

- 很简单的背包问题。先用一个 dp 来处理每个空间大小能赚多少钱，再用一个 dp 来处理每个空间大小需要花多少钱来购买升级服务。
- 虽然空间升级服务的空间很大，但是能买的商品有限，因此最需要处理和总空间相同的空间即可。
- 当然，比准备的空间大的服务需要特殊处理。复杂度  $O(n \times d)$ 。

## 学术不端 (easy)

- 前缀和枚举起点终点， $n^2$  暴力即可。

## 学术不端 (hard)

- 方差体现的是数据的偏离程度，如果一个序列的长度为 1，根据定义方差为 0，在序列长度增大时方差似乎总是变大，有时可能变小，但可以在另一端去掉几个数使方差更小。
- 容易猜到这题很可能序列越短方差越小。尝试打表会发现，取到方差最小的子段长度要么是 2 要么是 3。这其实就是正确的结论。
- 要严格证明比较麻烦。这里跳过一些数学上的步骤，直接给出推好的式子（感兴趣的同学可以自己搜索推导过程）。



## 学术不端 (hard)

- 有一个长度为  $n$  的序列  $x$  和一个长度为  $m$  序列  $y$ ，他们的平均数分别为  $E(x), E(y)$ ，方差分别为  $D(x), D(y)$ 。
- 现在将两个序列合并为一个序列  $z$ ，方差可以表示为：

$$D(z) = \frac{nD(x) + mD(y) + \frac{mn(E(x)-E(y))^2}{n+m}}{m+n}$$

- 容易看出， $D(x)$  和  $D(y)$  中至少会有一个比  $D(z)$  小。所以任何一个长度大于 1 的序列都可以分成 2 个子段，其中至少会有 1 个子段方差更小。
- 但是考虑到长度为 1 的序列方差一定是 0，题目要求的子段一定是长度在 2 或以上的，因此最小表现出这个性质的长度是 4（可以拆成  $2+2$ ，如果子段长度为 3，则拆成的子段会包含一个长度为 1 的）。
- 所以方差最小的子段长度要么是 2 要么是 3。接下来只需要扫一遍即可。

## 对拍

- 对于每一个点，后面不能先出现一个比它大的数再出现一个在两者之间的数，因此可以对每一个位置，找到他后面的最大值，然后在后面找一个次大值，这可以用 `std::set` 的 `lower_bound` 来做，时间复杂度  $O(n\log n)$ 。
- 但是这题有  $O(n)$  的做法，如果这个序列可以通过一个栈来存入输出，那么就是 `yes`，否则就是 `no`。

## 拯救飞船

- 由于费用是该链所在树大小 链的长度，而每个节点都会选到，因此最终减去的部分合起来就是所在链的长度，答案为 (关的次数-1) \* 节点数。
- 考虑除了每棵树上第一条被关闭的链以外，其它被关闭的链至少有一端与其它链上的节点距离等于 1 的限制，第一次取可以从叶子走到叶子，后面都只能从中间走到叶子。因此关的次数等于叶子-1。

## 这辈子算是有了

- 题目中限制了树的高度为 1000，因此我们可以考虑树形 dp，设状态为与当前节点距离为  $j$  的点的数量，然后从叶子节点开始暴力转移即可，复杂度  $O(n \times 1000)$ 。

## 乘法 (hard)

- 由于给出的数字是一段一段连续的并且由同一个数字组成，由此我们考虑用科学计数法来表示某一段数，例如 1111111，我们可以用  $(10^7 - 1)/9$  来表示，其余连续段均可以由该形式乘上一个常数来表示。
- 对于一个数来说，只需将每一段用该形式在十进制下左移若干位加起来即可，我们可以利用科学计数法的高精度来进行乘法运算，将最后的答案除以  $9^4$  即可。

# 结语

Thanks!