

写在前面

此套题完全符合普及组考纲，但较历年普及组题目，T1T2较简单。T3T4较难。题目实际难度排序为 $T1 < T2 < T4 < T3$ ，因为为了满足考纲T4为DP题，所以调整了难度排序。

题目标准算法分别为：T1为枚举题，T2为数学排列组合题，T3为二分答案+贪心题（或者二分答案+网络流），T4为DP题。

此套题基础分应为241分，而且使用各种其它算法也可以获得更多的分数。但为了保证分数梯度，特意将T3调整为思维难度较高的题目，T4调整为代码难度较高的题目。另外T3T4特意构造了很多需要注意细节处理和边界处理的数据。

注意：绝对搜不到原题！

内心OS：其实我T4出的是两行的，但本题主人公小H强烈要求说T4太水没难度，于是按照他的要求调整了题目难度。T4Solution还是要小H自己写吧，我才懒得写。

T1

算法一（期望得分100分）

直接枚举，对于第 i 位同学，他与第 $i - 1$ 位同学之间的距离应该为 $\max(a[i], a[i - 1])$ ，最后求和即可。

此题为普及组第一题难度，绝对没有超出。

算法二（期望得分0分）

玄学做法，我也不知道

T2

这道题部分分给得很多，也给很多数学不好的同学留了一条生路。

另外优秀的验题人恶意的提示我应该要将样例改得很小。

算法一（期望得分20分）

暴力枚举每一位同学的编号，DFS或BFS即可，期望时间复杂度为 $O(\prod_{i=1}^n a[i])$ ，大概会小一点点但也不会太多。

算法二（期望得分10分）

观察特殊性质，小学生都可以发现此时的方案数为 $a[i]!$ ，期望时间复杂度为 $O(a[i])$ 。

算法三（期望得分40分）

记忆化搜索，具体可以看代码。期望时间复杂度玄学

算法四（期望得分70分）

不难发现，对于第 i 同学来说，他的方案数为 $a[i] - i + 1$ ，根据乘法原理最后的答案就是 $\prod_{i=1}^n a[i] - i + 1$ 。另外注意，对于这些同学来说，其可选择的编号应该是递增的，所以需要排序。冒泡排序期望时间复杂度为 $O(n^2 + n)$ 。

另外优秀的验题人好心的提示我应该给不会使用 `sort` 的同学们一个友善的帮助。

算法五（期望得分100分）

按照算法四的思路快排即可，期望时间复杂度 $O(n \log_2 n + n)$ 。

T3

算法一（期望得分10分）

暴力枚举每一个时间选择完成哪一个账户，期望时间复杂度为 $O(n!)$ 。大概会快一点不过也快不了多少。

算法二（期望得分10分）

这个分数是留给更加优秀的算法（即数据点编号3），因为可能有些乱搞可以过这个特殊性质。另外，这个可能通过某些特判也可以过，答案反正是0或1对吧。

算法三（期望得分30分）

我们观察题目，可以发现最后的答案就是每一个账户操作时间 x 乘以总账户 n ，即 xn 。那么不难想到暴力枚举每一个 x 判断是否可以成为答案。

对于 x 来说，其可以选择的范围为 $[0, \max(l_i, r_i)]$ ，即数据范围中的 $[0, 3 * 10^2]$ 。

我们可以考虑贪心。对于第 i 个账户来说，只有账户 j 保证 $r_j \leq r_i$ 才可以对他造成影响。所以说，我们首先可以以 $r_i < r_j$ 作为第一关键字进行排序。另外，对于 $r_i = r_j$ 我们也需要考虑如何排序，可以得到当操作时间 $(r_i - l_i) < (r_j - l_j)$ ，即可操作时间越短我们就越先处理，所以第二关键字就是 $l_i > l_j$

排完序后我们需要考虑对于第 i 个账户如何进行选择，显然可以根据贪心思想优先选择越早的时间。

期望时间复杂度为 $O((r_i - l_i) * n * \max(l_i, r_i))$ ，注意一下常数，时间给得比较紧。

算法四（期望得分100分）

我们可以观察得到，其实可选的时间应是由0开始连续递增的，到了某一个最大值及答案便不可以继续选择的。这满足了二分答案的连续性和规律性，所以考虑二分答案。所以算法三就是我们二分答案的判断函数。注意二分答案的边界处理即可。

期望时间复杂度为 $O((r_i - l_i) * n * \log_2(\max(l_i, r_i)))$ ，这个时间也蛮近，不过只要头不太铁也应该可以过。

T4

前言

这道题是LJF为了防止AK的神题

9 + 8 pts算法

(数据第1 ~ 4行)

直接DFS。

期望：17 pts

7 pts算法

```
1 | printf("0\n");
```

why?

棋盘上空很少

0的期望很大

期望：7 pts

23 pts算法

相当于只有2行，考虑贪心

从前往后扫，如果一列都空，就放。

放的时候先看前面能不能塞一个，不行就看后面。

塞积木可以任意塞，反正不影响（LJFdalao上台证明一下吧）

期望：23 pts

100 pts算法

考虑dp。

记 $f_{i,j}$ 表示第 j 行在状态 i 时不往后放的情况下最多方案数，状态 i 可以状压表示，也可以 $f_{0/1,0/1,0/1,j}$ 表示

具体转移方程看std：

总结

LJF太强了！

这题太毒了！

std太难打了！

难度：

NOI/NOI+/CTSC

小H🔫gun!!!!这是你自己改得题面，管我什么事！！！！