

总结：场上按C□K□G□顺序通过了四个题；  
这几个题也都是本场的签到题QAQ□

## C

过得比较顺利也比较快~

## K

其实思路想得挺快的，但是一直在纠结  $O(n^3)$  的dp会不会卡不过去，反复思考和纠结了很久后  $O(n^3)$  过掉了QAQ

## G

签到题-13这就是废物吧呜呜~

一开始Stockholm想了一个奇怪的结果为  $\frac{n}{2}$  的算法，先WA了几次。

然后我去debug的时候，感觉  $\frac{n}{2}$  是不够优的，于是想到了分为  $\sqrt{n}$  组，每组递增（我也不知道我为什么要递增），这样答案会由  $\frac{n}{2}$  优化到  $2\sqrt{n}$ ；

“如此大程度的优化一定不会有问题吧！”——再次WA > <

“一定是我没考虑什么特殊情况！3似乎不对...7也不对”——WA 6 > <

Toby-Shi：“你都分为  $\sqrt{n}$  组了，每组递减排列不就  $\sqrt{n}$  了吗”

“你说得对..AC了呜呜~”

## J

看到的太晚了QAQ□线性回归公式就可以了~

不过还是由于精度问题WA了两次(#`O`)

## 赛后补题及其感想

### 关于D

二分答案 + spfa 判断负环；

场上Toby-Shi WA了好多好多遍，结束后一看通过了97.3%\$..(°—°”)

然后我把 spfa 的负环条件改吧改吧了一下就 AC 了，感觉是奇怪的精度问题 > <


### 关于E

比较常见的容斥原理  $F_{n,i} = \sum_{j=i}^n (-1)^{j-i} \times C_j^i \times 26^{n-3j} \times C_{n-2}^j$

将组合数拆开，并将只与  $i$  相关的项移到左边：

$$F_{n,i} = \sum_{j=i}^n \frac{(-1)^{j-i}}{(j-i)!} \times j! 26^{n-3j} C_{n-2}^j$$
  
然后就可以用ntt求解了！ヽ( ° ° )ノ

From: <https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link: [https://wiki.cvbbacm.com/doku.php?id=2022-2023:teams:loaf\\_on\\_contest:front\\_page:nowcoder2&rev=1659277202](https://wiki.cvbbacm.com/doku.php?id=2022-2023:teams:loaf_on_contest:front_page:nowcoder2&rev=1659277202) 

Last update: **2022/07/31 22:20**