

NOIP2022 模拟赛

p_b_p_b

2022.9

题目名称	彩票	战斗	扑克	排队
题目类型	传统型	传统型	传统型	传统型
可执行文件名	lottery	battle	poker	queue
输入文件名	lottery.in	battle.in	poker.in	queue.in
输出文件名	lottery.out	battle.out	poker.out	queue.out
每个测试点时限	2.0 秒	4.0 秒	3.0 秒	2.0 秒
内存限制	1024 MB	1024 MB	1024 MB	1024 MB
比对方式	全文比较	全文比较	全文比较	全文比较

提交源程序文件名

对于 C++ 语言	lottery.cpp	battle.cpp	poker.cpp	queue.cpp
-----------	-------------	------------	-----------	-----------

编译选项

对于 C++ 语言	-lm -O2 -std=c++17
-----------	--------------------

注意事项与提醒（请选手务必仔细阅读）

1. 出题人太菜了，如果 AK 了请不要大声喧哗。
2. 出题人不会造数据，请自觉只拿自己该拿的分，谢谢。

1 彩票

1.1 Statement

你买了若干张彩票，每张彩票上都有一个六位数字，彩票上的数字两两不同。有三种奖：

1. 一等奖一份，300000 元，只有当彩票上的六位数字完全匹配一等奖的六位数字 X 时才会获奖。
2. 二等奖若干份，每份 4000 元，在彩票上的后四位数字匹配二等奖的四位数字 Y 时会获奖。
3. 三等奖若干份，每份 500 元，在彩票上的后两位数字匹配三等奖的两位数字时会获奖。特别地，三等奖有三个不同的两位数字 Z_1, Z_2, Z_3 ，彩票只要能匹配任意一个就可以获奖。

以上提到的所有两位四位六位数字都可能带有前导零。

为了保证趣味性，一二三等奖数字的后两位必须两两不同，即 X, Y, Z_1, Z_2, Z_3 的后两位两两不同。

给定你手上的彩票，求出你最多能获得多少奖金。

1.2 Input Format

第一行一个正整数 n 。

接下来 n 行，每行一个六位数字，表示你手中的一张彩票，注意可能存在前导零。

1.3 Output Format

一行一个非负整数，表示能获得的最大奖金。

1.4 Sample 1 Input

```
7
034207
924837
372745
382947
274637
```

```
083907
294837
```

1.5 Sample 1 Output

```
309500
```

1.6 Sample 2 Input

```
10
012389
456789
234589
678989
890189
567889
123489
263784
901289
345689
```

1.7 Sample 2 Output

```
304500
```

1.8 Constraints

本题采用子任务捆绑测试。

对于所有数据，保证 $1 \leq n \leq 10^5$ 。

Subtask 1 (20 points) : $n \leq 40$ 。

Subtask 2 (20 points) : $n \leq 100$ 。

Subtask 3 (60 points) : 无特殊限制。



2 战斗

2.1 Statement

有 n 个人排成一排打群架，编号为 1 到 n ，每个人有能力值 a_i ，而故事的主角是第 K 个人。

打架会进行 $n - 1$ 轮，每轮等概率随机抽取一对相邻的人，他们两个打一架。如果一个人的能力值是 X 而另一个人的能力值是 Y ，那么前者获胜的概率就是 $X/(X + Y)$ ，后者获胜的概率是 $Y/(X + Y)$ 。战败的一方退出战斗，其他人按照原顺序仍然排成一排，没有空隙。

求出第 K 个人最终获胜的概率。不取模，你输出的答案合法当且仅当它与正确答案的绝对误差不超过 10^{-9} 。

2.2 Input Format

第一行两个正整数 n, K 。

第二行 n 个正整数 a_1, \dots, a_k 。

2.3 Output Format

输出一行一个实数，表示答案。

2.4 Sample 1 Input

```
4 2
2103 2019 1911 2331
```

2.5 Sample 1 Output

```
0.17753927527829697
```

2.6 Sample 1 Explanation

第二个人肯定需要和第一个人打一架，赢的概率是 0.489810771470。

右边的情况则比较复杂，需要分类讨论比赛是如何进行的，以及谁是胜者。

1. 第二个人把第三个人和第四个人都干烂。概率是 $0.5137 \times 0.4641 = 0.2384$ 。
2. 第三个人把第四个人干烂，然后被第二个人干烂。概率是 $0.4505 \times 0.5137 = 0.2314$ 。

3. 第四个人把第三个人干烂，然后被第二个人干烂。概率是 $0.5495 \times 0.4641 = 0.2550$ 。

因此答案是 $0.489810771470 \times (0.5 \times 0.238446433272 + 0.5 \times (0.231437533066 + 0.255046090816)) = 0.177539275278$ 。

2.7 Sample 2 Input

```
12 6
42 88 13 11 71 55 32 13 72 53 37 50
```

2.8 Sample 2 Output

```
0.06697114679670368
```

2.9 Sample 3,4

见下发文件。

2.10 Constraints

本题采用子任务捆绑测试。

对于所有数据，保证 $1 \leq n \leq 500, 1 \leq K \leq n, 1 \leq a_i \leq 3000$ 。

Subtask 1 (10 points) : $n \leq 8$ 。

Subtask 2 (10 points) : $n \leq 20$ 。

Subtask 3 (20 points) : $n \leq 70$ 。

Subtask 4 (30 points) : $n \leq 150$ 。

Subtask 5 (30 points) : 无特殊限制。

3 扑克

3.1 Statement

你有 n 张牌，每张牌上写了一个质数，可能有重复。

你需要把这些牌分成两个非空集合，使得第一个集合的数之和恰好等于第二个集合的数之积。

对于一个合法拆分，你的得分是第一个集合的数之和（也就是第二个集合的数之积）。

求出最大得分。若没有合法拆分则输出 0。

3.2 Input Format

第一行一个正整数 T ，表示数据组数。每组数据的输入格式如下：

第一行一个正整数 m ，表示牌上不同的质数个数。

接下来 m 行，每行两个正整数 p_i, n_i ，表示质数 p_i 有 n_i 个。

3.3 Output Format

对于每组数据输出一行，先输出 Case # n :，其中 n 是数据编号，然后输出一个非负整数，表示答案。

3.4 Sample Input

```
4
5
2 2
3 1
5 2
7 1
11 1
1
17 2
2
2 2
3 1
1
2 7
```

3.5 Sample Output

Case #1: 25

Case #2: 17

Case #3: 0

Case #4: 8

3.6 Constraints

本题采用子任务捆绑测试。

对于所有数据，保证 $1 \leq T \leq 7, 1 \leq m \leq 95, 2 \leq p_i \leq 499, n_i \geq 1, \sum n_i \leq 10^{15}$ ，保证 p_i 是质数。

Subtask 1 (20 points) : $\sum n_i \leq 10$ 。

Subtask 2 (37 points) : $\sum n_i \leq 100$ 。

Subtask 3 (43 points) : 无特殊限制。

4 排队

4.1 Statement

有 n 个人准备排队吃饭。第 i 个人来自 c_i 号小组，素质值是 a_i 。注意素质值越高表示素质越低。

这些人会按照编号顺序挨个进入食堂排队。队伍初始为空。

第 i 个人进入食堂时队伍由前 $i - 1$ 个人组成。他会尝试插队，一个插队的位置是合法的，当且仅当插了不超过 a_i 个人的队且相邻的位置有同一小组的人。如果有多个合法的插队位置则会选择最靠前的位置，如果没有合法的插队位置则会排在最后。

求出最终的队伍会是什么样子。

4.2 Input Format

第一行一个正整数 n 。

接下来 n 行，第 i 行两个正整数 c_i, a_i 。

4.3 Output Format

一行 n 个正整数，从左到右分别表示从队头到队尾的人的编号。

4.4 Sample Input

```
8
1 0
2 1
2 2
1 1
3 2
1 3
2 3
2 5
```

4.5 Sample Output

```
1 3 8 2 7 6 4 5
```


4.6 Constraints

本题采用子任务捆绑测试。

对于所有数据，保证 $1 \leq n \leq 4 \times 10^5, 1 \leq c_i \leq n, 0 \leq a_i \leq 4 \times 10^5$

Subtask 1 (20 points) : $n \leq 3000$ 。

Subtask 2 (20 points) : $c_i \leq 50$ 。

Subtask 3 (20 points) : 保证数据随机。

Subtask 4 (40 points) : 无特殊限制。