

2024 洛谷网校基础-提高衔接计划

模拟比赛

第三场

时间：2024 年 9 月 7 日 08:30 ~ 12:00

题目名称	试卷分发	卡牌游戏	驿站管理	童话镇
题目类型	传统型	传统型	传统型	传统型
目录	paper	card	express	fairy
可执行文件名	paper	card	express	fairy
输入文件名	paper.in	card.in	express.in	fairy.in
输出文件名	paper.out	card.out	express.out	fairy.out
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	1.0 秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB
测试点数目	10	20	20	20
测试点是否等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	paper.cpp	card.cpp	express.cpp	fairy.cpp
-----------	-----------	----------	-------------	-----------

编译选项

对于 C++ 语言	-O2 -std=c++14 -static
-----------	------------------------

注意事项（请仔细阅读）

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C/C++ 中函数 main() 的返回值类型必须是 int，程序正常结束时的返回值必须是 0。
3. 提交的程序代码文件的放置位置请参考本省的具体要求。
4. 因违反以上三点而出现的错误或问题，申诉时一律不予受理。
5. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
6. 选手提交的程序源文件必须不大于 100KB。
7. 程序可使用的栈空间内存限制与题目内存限制一致。
8. 统一评测时采用的机器配置为 Intel(R) Core(TM) i5-12400 CPU @ 2.50GHz，内存 16GB。上述时限以此配置为准。
9. 只提供 Linux 格式附加样例文件。
10. 评测在当前最新公布的 NOI Linux 下进行，各语言的编译器版本以此为准。

试卷分发 (paper)

【题目描述】

小 F 将负责 H 省 2024 年 CSP 第一轮考试试卷的分发。试卷已经按 30 份一袋与 10 份一袋的规格封装完毕。

某考点共有 n 个考场，第 i 个考场共安排了 a_i 名考生。每个考场可以分配多袋试卷，分配的规则如下所示：

- 若 $a_i = 30k$ ，其中 k 为非负整数，则分配 k 袋 30 份的试卷。
- 若 $a_i = 30k + p$ ，其中 k 为非负整数且 $0 < p \leq 10$ ，则分配 k 袋 30 份的试卷与 1 袋 10 份的试卷。
- 若 $a_i = 30k + p$ ，其中 k 为非负整数且 $10 < p \leq 20$ ，则分配 k 袋 30 份的试卷与 2 袋 10 份的试卷。
- 若 $a_i = 30k + p$ ，其中 k 为非负整数且 $20 < p < 30$ ，则分配 $k + 1$ 袋 30 份的试卷。

请问，30 份与 10 份的试卷袋，该考点分别需要多少袋？

【输入格式】

从文件 `paper.in` 中读入数据。
输入的第一行为一个正整数 n 。
输入的第二行为 n 个正整数 a_1, a_2, \dots, a_n 。

【输出格式】

输出到文件 `paper.out` 中。
输出一行两个整数，分别表示考点所需的 30 份与 10 份的试卷袋数目，用一个空格隔开。

【样例 1 输入】

```
1 5
2 15 30 40 50 55
```

【样例 1 输出】

```
1 5 5
```

【样例 2】

见选手目录下的 *paper/paper2.in* 与 *paper/paper2.ans*。

【子任务】

对于 40% 的测试数据， $1 \leq n \leq 1000$ ， $1 \leq a_i \leq 1000$ ；

对于另外 20% 的测试数据， a_i 为 30 的倍数；

对于 100% 的测试数据， $1 \leq n \leq 10^6$ ， $1 \leq a_i \leq 10^9$ 。

卡牌游戏 (card)

【题目描述】

小 F 非常喜欢一种卡牌游戏，这种游戏需要 n 个人参加。

游戏开始时，每个人会领取到一张牌。之后每一轮游戏，第 i 个人会将手上的卡牌交给第 p_i 个人。为了保证每一轮游戏后，每个人手上都有一张卡牌，在设计游戏时，小 F 保证了 p_1, p_2, \dots, p_n 是一个排列。

给出排列 p ，小 F 想知道，最少多少轮游戏后，所有人曾再次取得初始的卡牌。

【输入格式】

从文件 `card.in` 中读入数据。

输入的第一行为一个整数 n ，代表参加游戏的人数。

输入的第二行为 n 个整数，第 i 个代表 p_i ，含义如题所示。

【输出格式】

输出到文件 `card.out` 中。

输出一行一个整数表示答案。

【样例 1 输入】

```
1 5
2 2 3 4 5 1
```

【样例 1 输出】

```
1 5
```

【样例 1 解释】

假设一开始拿到的卡牌依次为 $[1, 2, 3, 4, 5]$ 。

第一轮游戏后，为 $[5, 1, 2, 3, 4]$ 。

第二轮游戏后，为 $[4, 5, 1, 2, 3]$ 。

第三轮游戏后，为 $[3, 4, 5, 1, 2]$ 。

第四轮游戏后，为 $[2, 3, 4, 5, 1]$ 。

第五轮游戏后，为 $[1, 2, 3, 4, 5]$ 。

因此，答案为 5。

【样例 2】

见选手目录下的 *card/card2.in* 与 *card/card2.ans*。

【子任务】

对于 30% 的测试数据，满足 $1 \leq n \leq 15$ ；

对于另外 20% 的测试数据，满足 $p_i = i \bmod n + 1$ ；

对于另外 20% 的测试数据，满足 n 为偶数。当 i 为奇数， $p_i = i + 1$ ；当 i 为偶数， $p_i = i - 1$ 。

对于 100% 的测试数据，满足 $1 \leq n \leq 10^5$ ， $1 \leq p_i \leq n$ ，保证 p 构成一个排列。

驿站管理 (express)

【题目描述】

小 F 运营了一家快递驿站，驿站共有 n 个货架，依次编号为 $1, 2, \dots, n$ 。每个货架有 m 层，依次编号为 $1, 2, \dots, m$ ，不同的货架编号独立。每层最多容纳 k 个快递，依次编号为 $1, 2, \dots, k$ 。

快递根据尺寸不同，可以分为三种类型：大件、中件、小件，依次用数字 $1 \sim 3$ 表示。每个货架每层可以存放的快递类型是提前规定好的。

驿站每收入一件快递，需要依照存放的位置给予该快递一个编号，形如 $x - y - z$ ，其中 x 代表快递所在货架的编号， y 代表快递所在层的编号， z 是快递在该层的编号。所有仍存储在驿站中的快递编号必须互不相同，一件快递被取走后，其编号可以被再次使用。

一件快递的编号按照如下原则确定：

1. 在所有可以存储该快递的货架中，选择编号最小的货架
2. 在该货架所有可以存储该快递的层中，选择编号最小的层
3. 在该层中给予该快递尽可能小的编号

在驿站的日常运营中，既会出现收入快递的事件，也会出现客户取走快递的事件：

- **1 S** 表示驿站收入一件尺寸为 S ($1 \leq S \leq 3$) 的快递，你需要输出该快递的编号。若没有足够的空间收入该快递，则拒收，并输出 **Reject**。
- **2 id** 表示客户取走编号为 id 的快递，保证编号符合上面题目中所给格式且该快递存在。

【输入格式】

从文件 *express.in* 中读入数据。

输入的第一行为三个正整数 n, m, k 。

接下来 n 行，每行 m 个正整数 $a_{i,j}$ ，表示第 i 行第 j 层存放尺寸为 $a_{i,j}$ 的快递。

接下来一行一个整数 T ，表示驿站日常运营中发生的事件总数。

接下来 T 行，每行描述一个事件，格式如题目描述所示。

【输出格式】

输出到文件 *express.out* 中。

对于每个事件 1，输出一行，表示该快递的编号，或输出 **Reject** 表示拒收。

【样例 1 输入】

```
1 3 3 2
2 2 2 1
3 1 2 1
4 1 2 2
5 20
6 1 2
7 1 1
8 1 1
9 2 3-1-2
10 1 1
11 2 3-1-1
12 1 2
13 2 3-3-2
14 2 3-3-1
15 1 1
16 2 3-1-1
17 2 3-1-2
18 1 2
19 1 2
20 1 1
21 1 2
22 2 3-1-2
23 1 2
24 2 3-2-1
25 1 1
```

【样例 1 输出】

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

9	2-3-1
10	2-2-1
11	2-2-2
12	2-3-2

【样例 2】

见选手目录下的 *express/express2.in* 与 *express/express2.ans*。

【子任务】

对于 100% 的测试数据， $1 \leq n, m, k \leq 100$ ， $1 \leq T \leq 10^5$ ， $1 \leq S \leq 3$ 。

	$n =$	$m =$	$k =$	
1	1	1	1	
$2 \sim 3$			100	
$4 \sim 5$	100	100	1	
$6 \sim 7$	1		100	
$8 \sim 9$	100	1		$T \leq 1000$
$10 \sim 12$		100		
$13 \sim 15$				
$16 \sim 20$				

童话镇 (fairy)

【题目背景】

总有一条蜿蜒在童话镇里梦幻的河
分隔了理想分隔现实
又在前方的山口汇合

【题目描述】

童话是最美好的梦，一篇童话可以用一个小写英文字符串 s 表示，字符 s_i 表示这篇童话的第 i 句话。两篇童话 s, t 意境相同，当且仅当 $s = t$ 。

遗憾的是，童话的片段可能丢失。字符串中可能出现一些 `_`，表示童话在这里丢失了若干句话（不少于一句），也可能出现一些 `%`，表示童话在这里丢失了恰好一句话。现在，给出两个有部分片段丢失的童话 S, T 。请问在片段丢失前，童话 S, T 的意境是否可能相同。

【输入格式】

从文件 `fairy.in` 中读入数据。

每个测试点包含多组测试数据。

输入的第一行为一个正整数 Q ，表示测试数据组数。

对于每组测试数据，输入两行：

- 第一行为一个字符串 S 。
- 第二行为一个字符串 T 。

【输出格式】

输出到文件 `fairy.out` 中。

对于每组测试数据，输出一行一个字符：

- 若可以配对，输出 `Y`
- 若不可配对，输出 `N`

【样例 1 输入】

```
1 5
2 abc
3 abc
4 _bc
5 abc
```

```
6 a%
7 abc
8 abc%
9 acdef
10 a_b%_
11 aaaaaaaaaaaaaaaaaabaaaaaaa
```

【样例 1 输出】

```
1 Y
2 Y
3 N
4 N
5 Y
```

【样例 2】

见选手目录下的 *fairy/fairy2.in* 与 *fairy/fairy2.ans*。

【子任务】

对于 100% 的测试数据， $1 \leq Q \leq 10$ ， $1 \leq |S|, |T| \leq 1000$ ，其中 $|s|$ 表示字符串 s 的长度。字符串仅由小写英文字母与 %、_ 组成。

测试点	$ S \leq$	$ T \leq$	特殊性质
1	1	1	无
2 ~ 3		10	
4 ~ 5		10^3	
6	AB		
7	AD		
8	AF		
9	BC		
10	BE		
11	CD		
12	CF		
13	DE		
14	EF		
15	A		
16	B		
17 ~ 20	无		

特殊性质 A：字符串 S 仅由英文小写字母组成。

特殊性质 B：字符串 T 仅由英文小写字母组成。

特殊性质 C：字符串 S 仅由英文小写字母和字符 % 组成。

特殊性质 D：字符串 T 仅由英文小写字母和字符 % 组成。

特殊性质 E：字符串 S 仅由英文小写字母和字符 _ 组成。

特殊性质 F：字符串 T 仅由英文小写字母和字符 _ 组成。