

联合省选 2022 模拟赛

Sacrilegious

2022 年 3 月 2 日

题目名称	恰空	唐璜	塔德伍施
题目类型	传统	传统	传统
目录	ciaccona	giovanni	tadeusz
可执行文件名	ciaccona	giovanni	tadeusz
输入文件名	ciaccona.in	giovanni.in	tadeusz.in
输出文件名	ciaccona.out	giovanni.out	tadeusz.out
每个测试点时限	1.0 秒	1.0 秒	3.0 秒
内存限制	256MB	256MB	256MB
测试点数目	10	4	4
测试点是否等分	是	否	否

提交源程序程序名

对于 C++ 语言	ciaccona.cpp	giovanni.cpp	tadeusz.cpp
-----------	--------------	--------------	-------------

编译选项

对于 C++ 语言	<code>-lm -O2 -std=c++14</code>
-----------	---------------------------------

【注意事项（请仔细阅读）】

1. 文件名（包括程序名和输入输出文件名）必须使用英文小写。
2. C++ 中函数 `main()` 的返回值类型必须是 `int`，值必须为 0。
3. 若无特殊说明，结果比较方式为忽略行末空格、文末回车后的全文比较
4. 程序可使用的栈空间大小与该题内存空间限制一致。
5. 在终端中执行命令 `ulimit -s unlimited` 可将当前终端下的栈空间限制放大，但你使用的栈空间大小不应超过题目限制。
6. 若无特殊说明，输入文件与输出文件中同一行的相邻整数均使用一个空格分隔。
7. 输入文件中可能存在行末空格，请选手使用更完善的读入方式（例如 `scanf` 函数）避免出错。
8. 使用 `std::deque` 等 STL 容器时，请注意其内存空间消耗。

恰空 (ciaccona)

【问题描述】

给定 n 和一个长度为 n 的序列 a_i 。

Alice 和 Bob 在玩游戏，Alice 先手。每人轮流从序列中删去任意一个数，如果某人操作前序列的异或和为 0，那么这个人获胜。

空序列的异或和为 0。

假设 Alice 和 Bob 都使用最优策略，问谁会获胜。

【输入格式】

从文件 `ciaccona.in` 中读入数据。

输入数据第一行一个正整数 T ($T \leq 10$)，表示数据组数。

对于每组数据，第一行包含一个正整数 n 。

第二行包含 n 个非负整数，表示序列 a_i 。

【输出格式】

输出到文件 `ciaccona.out` 中。

对于每组数据，输出一行一个字符串，表示获胜的人。

【样例 1 输入】

```
1 1
2 2
3 1 2
```

【样例 1 输出】

```
1 Alice
```

【测试点约束】

测试点编号	$n \leq$	$a_i \leq$
1 ~ 4	10	$2^{30} - 1$
5 ~ 8	100,000	1
9 ~ 10	100,000	$2^{30} - 1$

唐璜 (giovanni)

【问题描述】

给定 $n \times m$ 的字符矩阵 c 。

请输出所有的 k ，满足可以用 $1 \times k$ 或 $k \times 1$ 的长条将矩阵不重不漏地覆盖，并且将长条上的字母从左到右，从上到下写下来以后，所有长条上的字符串都相同。

【输入格式】

从文件 `giovanni.in` 中读入数据。

输入数据第一行包含两个正整数 n, m ，表示字符矩阵的大小

接下来 n 行，表示字符矩阵。

【输出格式】

输出到文件 `giovanni.out` 中。

第一行一个整数，表示合法的 k 的个数。

第二行 k 个正整数，表示所有合法的 k ，按升序输出。

【样例 1 输入】

```
1 4 8
2 aabaaaaa
3 babaabbb
4 aabaaaaa
5 aabaaaaa
```

【样例 1 输出】

```
1 1
2 4
```

【样例 1 解释】

$k = 4$ 时，一种合法的方案如下：

a	a	b	a	a	a	a	a
b	a	b	a	a	b	b	b
a	a	b	a	a	a	a	a
a	a	b	a	a	a	a	a

【测试点约束】

本题采用子任务捆绑测试。

子任务编号	分值	$n \leq$	$m \leq$
1	3	1	1000
2	13	3	1000
3	39	20	20
4	45	1000	1000

数据保证矩阵仅包含小写字母。

塔德伍施 (tadeusz)

【问题描述】

你需要维护一张图，初始有编号为 1 和 2 的两个结点，以及连接 1 和 2 的一条边。
你需要维护一下三种操作：

- ? u 表示询问点 u 的度数。
- W u 新建一点与 u 连边。
- Z u 新建一点与所有与 u 相邻的点连边。

新建点的标号为当前标号最大值 +1。

【输入格式】

从文件 `tadeusz.in` 中读入数据。
输入数据第一行包含一个正整数 n ，表示操作数。
接下来 q 行，每行表示一次操作。

【输出格式】

输到文件 `tadeusz.out` 中。
输出若干行，每行表示一次询问的答案。

【样例 1 输入】

```
1 7
2 ? 1
3 Z 2
4 ? 1
5 Z 1
6 W 2
7 ? 2
8 ? 3
```

【样例 1 输出】

1 1
2 2
3 3
4 2

【样例 2】

见下发文件中的 `tadeusz/tadeusz2.in` 和 `tadeusz/tadeusz2.out`。
该样例与子任务 2 满足同样的约束条件。

【样例 3】

见下发文件中的 `tadeusz/tadeusz3.in` 和 `tadeusz/tadeusz3.out`。
该样例与子任务 3 满足同样的约束条件。

【测试点约束】

本题采用子任务捆绑测试。

子任务编号	分值	$n \leq$	特殊性质
1	4	5,000	无
2	7	1,000,000	不存在第二种操作
3	41	1,000,000	所有询问在操作之后
4	48	1,000,000	无