

NOI2025 广东省队集训

GDOI

题目名称	这是第一题	这是第二题	这是第三题
题目类型	传统题	传统题	传统题
目录	first	second	third
可执行文件名	first	second	third
输入文件名	first.in	second.in	third.in
输出文件名	first.out	second.out	third.out
每个测试点时限	1 秒	2 秒	1 秒
内存限制	512 MB	512 MB	512 MB
子任务数目	3	4	4
测试点是否等分	否	否	否

提交源程序文件名

对于 C++ 语言	first.cpp	second.cpp	third.cpp
-----------	------------------	-------------------	------------------

编译选项

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

注意事项（请仔细阅读）

1. 测试机器：CPU(AMD Ryzen 7 5700G 3.80 GHz), RAM 32G。
2. 测试环境：Windows 10, **g++** 13.2.0, Lemonlime 0.3.4。
3. 文件名（程序名和输入输出文件名）必须使用英文小写。
4. **C/C++** 中函数 **main()** 返回类型必须是 **int**，程序正常结束返回值必须是 0。
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的内存限制一致。

这是第一题 (first)

【题目描述】

Alice 和 Bob 在一个 n 行 m 列的网格上博弈。初始时，网格中的一些格子里有障碍，一些格子里有硬币。这个网格的列编号是循环的，即 $m+1$ 列就是第 1 列， -1 列就是第 m 列。

两个玩家分别进行操作，Alice 先手，操作如下：

- 选择一枚硬币，设其处于第 i 行第 j 列，记作 (i, j) ，
- 将其移动到 $(i+1, j)$ ， $(i, j-1)$ ， $(i, j+1)$ 中的一个非障碍格子上，
- 不存在此前的某个时刻，该硬币在目标格子上。即硬币的移动不能成环。

注意：不同的硬币之间没有限制，允许硬币重叠。

不能进行任何操作的玩家被判负。

假设两名玩家都采取最优策略，请求出谁会获胜。

【输入格式】

从文件 *first.in* 中读入数据。

第一行包含两个整数 n, m ，表示网格图的大小。

接下来 n 行，每行 m 个字符。第 i 行的第 j 个字符表示格子 (i, j) 的状态：

- 若其为 `.`，表示该格子没有任何东西，
- 若其为 `#`，表示该格子上有障碍，
- 若其为 `B`，表示该格子上有一个硬币。

【输出格式】

输出到文件 *first.out* 中。

输出一行一个字符串。若 Alice 获胜，需输出 `Alice`，否则输出 `Bob`。

【样例 1 输入】

```
1 2 3
2 B.#
3 #..
```

【样例 1 输出】

1 Alice

【样例 2 输入】

1 1 1
2 B

【样例 2 输出】

1 Bob

【样例 3】

见选手目录下的 *first/first3.in* 与 *first/first3.ans*。

【样例 4】

见选手目录下的 *first/first4.in* 与 *first/first4.ans*。

【数据范围】

对于所有测试点， $1 \leq n, m \leq 1000$ ，每个字符都在 `.#B` 中。

子任务编号	特殊性质	分值
1	$1 \leq n, m \leq 10$	10
2	保证第一行以外的每一行都至少存在一个障碍	40
3	无特殊限制	50

这是第二题 (second)

【题目描述】

有一个 $2n \times 2m$ 的网格，第 i 行第 j 列记作坐标 (i, j) ，行列编号均从 0 开始。
初始时 (i, j) 格上有数字 $2mi + j + 1$ ，并以 2×2 为单位进行黑白交替染色。以 $n = 2, m = 3$ 为例，下为初始网格：

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

其中加粗的格子即为被染为黑色的格子。注意一个格子的颜色与其上面的数字无关。
对网格进行 q 次操作，每次操作为下列五个中的一个：

- 1. 对每个 $0 \leq j < m$ ，交换 $2j$ 列与 $2j + 1$ 列；
- 2. 对每个 $0 \leq j < m - 1$ ，交换 $2j + 1$ 列和 $2j + 2$ 列；
- 3. 对每个 $0 \leq i < n$ ，交换 $2i$ 行和 $2i + 1$ 行；
- 4. 对每个 $0 \leq i < n - 1$ ，交换 $2i + 1$ 行与 $2i + 2$ 行；
- 5. 对每个黑色 2×2 小网格 $\begin{smallmatrix} \mathbf{x} & \mathbf{y} \\ \mathbf{z} & \mathbf{w} \end{smallmatrix}$ ，变换为 $\begin{smallmatrix} \mathbf{z} & \mathbf{x} \\ \mathbf{w} & \mathbf{y} \end{smallmatrix}$ ，对每个白色 2×2 小网格 $\begin{smallmatrix} x & y \\ z & w \end{smallmatrix}$ ，

变换为 $\begin{smallmatrix} y & w \\ x & z \end{smallmatrix}$ 。

给定每次操作，求最终网格上每个位置的数字。

【输入格式】

从文件 `second.in` 中读入数据。
第一行输入 n, m, q 。
第二行输入 q 个整数，第 i 个整数 x_i 表示第 i 次操作为第几个操作。

【输出格式】

输出到文件 `second.out` 中。
输出 $2n$ 行每行 $2m$ 个整数，表示最终网格。

【样例 1 输入】

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

【样例 1 输出】

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

【样例 2】

见选手目录下的 `second/second2.in` 与 `second/second2.ans`。

【数据范围】

对所有测试点，保证 $1 \leq n, m, q \leq 10^6$ 且 $nm \leq 10^6$ ， $x_i \leq 5$ 。

子任务编号	特殊性质	分值
1	$x_i \in \{1, 3, 5\}$	20
2	$x_i \in \{1, 2\}$	20
3	$x_i \in \{1, 2, 3, 4\}$	20
4	无特殊限制	40

这是第三题 (third)

【题目描述】

一个 $n \times m$ 的棋盘上有若干位置被放上棋子，需要在部分空位上放上棋子，使得存在**唯一**的在棋子间两两匹配的方案，满足每对棋子在同一行或同一列。
请你求出额外添加的棋子数的最小值，或报告无解。

【输入格式】

从文件 *third.in* 中读入数据。
第一行两个整数 n, m ，表示棋盘大小。
接下来 n 行，每行一个长为 m 的字符串，第 i 行的第 j 个字符表示位置 (i, j) 的状态：

- 若其为 `.`，表示没有棋子，
- 若其为 `#`，表示有棋子。

【输出格式】

输出到文件 *third.out* 中。
输出一行一个整数。 -1 表示无解，否则表示最少添加的棋子数。

【样例 1 输入】

```
1 2 3
2 ###
3 ...
```

【样例 1 输出】

```
1 1
```

【样例 2 输入】

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

【样例 2 输出】

1 0

【样例 3 输入】

1 2 4
2 #.##
3 .#. #

【样例 3 输出】

1 -1

【样例 4 输入】

1 9 10
2#.....
3#.....
4#....
5 ..#.....
6 ...#.....
7 #.....#.....
8
9 .#.....
10#..

【样例 4 输出】

1 7

【数据范围】

对于所有测试点， $1 \leq n, m \leq 10^3$ ，每个字符都在 `.#` 中。

子任务编号	特殊性质	分值
1	$1 \leq n, m \leq 3$	15
2	$1 \leq n, m \leq 10$, 初始时没有两个棋子在同一行或同一列	10
3	$1 \leq n, m \leq 10$	30
4	无特殊限制	45