

# 会飞的小猪

## 【问题描述】

“天哪，我们家的猪怎么飞起来了！”佳佳看到自家上空飘着的一头头小猪，呆呆的站在那里，“怎么样才能把它们弄回各自的圈中呢？”

“佳佳，起床啦！”听到妈妈的声音，佳佳眼前一亮，一下子从床上坐了起来。“原来刚才是在做梦。好神奇啊……”佳佳吃早饭的时候还在回想着，“如果梦里的情景真的出现了，不是很有趣吗？”佳佳顾不得吃完早饭，立刻找来一张纸，趴在桌子上开始画了起来。

佳佳画了一个  $n \times n$  的方格地图，代表猪圈上空被分成  $n$  行  $n$  列，从上到下依次为第  $0, 1, 2, \dots, n-1$  行，从左到右依次为第  $0, 1, 2, \dots, n-1$  列。同时处于第  $x$  行和第  $y$  列的方格记为格子  $(x, y)$ 。佳佳在其中  $m$  个格子里各画了一个星号“\*”，表示那些格子就是猪圈，而在另外  $m$  个格子里各画了一只不同运动方向的小猪，我们从上到下从左到右把这些小猪编号为  $0, 1, 2, 3, \dots, m-1$ 。换句话说，设两只初始位置分别为  $(x_1, y_1)$  和  $(x_2, y_2)$  的小猪的编号分别为  $l_1$  和  $l_2$ ，则  $l_1 < l_2$  当且仅当①  $x_1 < x_2$ ；或者②  $x_1 = x_2$  并且  $y_1 < y_2$ 。佳佳定义了小猪运动方向的数字表示，数字 5 表示不动，数字 2 表示向南，4 向西，6 向东，8 向北运动。每一时刻，小猪都会沿着各自的运动方向行进一格，除非它已落进猪圈。如果在地图边界的小猪向地图外的方向前进一格之后被视作又回到了相对的边界。比如，在右边界上朝东运动的小猪经过一个时刻之后会进入左边界的格子中。为便于理解，处在格子  $(x, y)$  的小猪下一时刻的位置如下表所示。

运动方向	数字	下一时刻位置
不动	5	$(x, y)$
南	2	$((x+1) \bmod n, y)$
西	4	$(x, (y-1+n) \bmod n)$
东	6	$(x, (y+1) \bmod n)$
北	8	$((x-1+n) \bmod n, y)$

表 1. 朝向和下一时刻位置

佳佳希望自己能够控制小猪们的行动，因此设计了五种“吹气”操作，每个时间单位执行一次，每次执行可以改变一行或者一列中所有小猪的运动方向，五种操作如下表所示：

操作格式	参数含义	效果
X0	无（不想吹气时使用）	全部小猪运动方向均不变
Nc	作用于第 $c$ 列 $(0 \leq c \leq n-1)$	把第 $c$ 列小猪的运动方向改为北
Er	作用于第 $r$ 行 $(0 \leq r \leq n-1)$	把第 $r$ 行小猪的运动方向改为东
Wr	作用于第 $r$ 行 $(0 \leq r \leq n-1)$	把第 $r$ 行小猪的运动方向改为西
Sc	作用于第 $c$ 列 $(0 \leq c \leq n-1)$	把第 $c$ 列小猪的运动方向改为南

表 2. 五种吹气操作

请注意，后四种吹气操作都只改变运动方向，而不会影响到小猪的位置。

小猪的运动是以秒为单位的，初始时刻为 0。每秒初佳佳先进行吹气操作，改变一些（也可能是 0 个）小猪的运动方向，然后所有小猪按照表 1 的方式运动。如果在下一时刻会有多只猪运动到某一猪圈，让其中编号最小的一只进圈。其他猪仍可进入该格，但不视为进圈。换句话说，编号最小的那只猪和圈同时消失。这是因为一个圈只能容纳一只小猪，因此在有小猪落入其中以后，我们可以认为该猪圈和那只小猪一起消失。

编号为  $i$  的小猪进圈的时刻记为  $t_i (i=1,2,\dots,n)$ ， $T_1=\text{Max}\{t_i\}$ ， $T_2=\sum t_i$ 。佳佳的目标就是让  $T_2$  尽量小。佳佳知道这个任务很困难，所以他会根据你的  $T_2$  和目前已知最好的  $T_2$  的相对优劣程度给你奖励。

### 【输入格式】

输入文件 `airpig1.in` 到 `airpig10.in` 已经放在用户目录中，文件第一行为测试数据编号 `caseN`。第二行仅有一个整数  $n$ ，为地图的大小。以下  $n$  行每行  $n$  个字符，描述整个地图。字符“.”表示空地，“\*”表示猪圈，字符 5, 2, 4, 6, 8 分别表示天上小猪的运动方向，5 是不动，2 是向南运动，4 是向西、6 是向东、8 是向北运动。

### 【输出文件】

本题是一道提交答案式的题目。你应当提供十个输出文件 `airpig1.out` 到 `airpig10.out`，放在用户目录中。每个文件的第一行为测试数据编号 `caseN`，这个数应该和输入文件保持一致。第二行有两个整数  $T_1$  和  $T_2$ ， $T_1$  表示最后一只猪的入圈时刻， $T_2$  表示所有猪入圈时刻的总和。以下  $T_1$  行每行有一个字符串，表示每个时刻进行的吹气操作。吹气操作的输出格式见表 2。

### 【输入输出样例】

<code>airpig.in</code>	<code>airpig.out</code>
1	1
5	4 7
...*.	E2
..5..	E1
..5..	N3
*.....	S0
.....	

## 【样例解释】

时刻	1	2	3	4
当前地图	<pre> ...*. ..5.. ...6. *..... ..... </pre>	<pre> ...*. ...6. ...6 *..... ..... </pre>	<pre> ..... ..... 6..... *..... ..... </pre>	<pre> ..... ..... ..... ..... ..... </pre>
备注			$t_1=3$	$t_2=4$

## 【评分标准】

对于每个测试点，如果你的输出不正确，得 0 分，否则你的得分按下式计算：

$$score = [\text{目前已知最优输出} / \text{你的输出} * 9] + 1。$$

其中 $[x]$ 表示不超过  $x$  的最大整数。

## 【你如何测试自己的输出】

输出测试程序 `check` 放在用户目录下。使用方法为: `check <测试点编号 X>`。程序会自动读取输入文件 `airpigX.in` 和你的输出文件 `airpigX.out`，其中  $X=1,2,\dots,10$ 。程序会在屏幕上输出你的结果是否正确。