NOI2023 省选

DAY2

时间：2023 年 4 月 2 日 08:30 ～ 13:00

题目名称	过河卒	填数游戏	染色数组
题目类型	传统型	传统型	传统型
目录	zu	game	color
可执行文件名	zu	game	color
输入文件名	zu.in	game.in	color.in
输出文件名	zu.out	game.out	color.out
每个测试点时限	1.0 秒	2.0 秒	3.0 秒
内存限制	1 GiB	1 GiB	1 GiB
测试点数目	20	25	20
测试点是否等分	是	是	是

提交源程序文件名

对于 C++　语言	zu.cpp	game.cpp	color.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各省的具体要求。
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的内存限制一致。
8. 全国统一评测时采用的机器配置为：Intel(R) Core(TM) i7-8700K CPU @3.70GHz，内存 32GB。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。

过河卒（zu）

【题目背景】

棋盘上有一个过河卒，需要走到底线。卒行走的规则是可以向左移动一格，向右移动一格或者向前移动一格。同时在棋盘上有两个另一方的棋子，需要拦截这个卒走到底线。这两个棋子的走法和帅一致，可以走到前后左右四个方向上相邻的格子。因此本题可以称为“帅拦过河卒”。

【题目描述】

有一个 n 行 m 列的棋盘。我们用 (i,j) 表示第 i 行第 j 列的位置。棋盘上有一些障碍，还有一个黑棋子和两个红棋子。

游戏的规则是这样的：红方先走，黑方后走，双方轮流走棋。红方每次可以选择一个红棋子，向棋盘的相邻一格走一步。具体而言，假设红方选择的这个棋子位置在 (i,j)，那么它可以走到 $(i-1,j),(i+1,j),(i,j-1),(i,j+1)$ 中的一个，只要这个目的地在棋盘内且没有障碍且没有红方的另一个棋子。

黑方每次可以将自己的棋子向三个方向之一移动一格。具体地，假设这个黑棋子位置在 (i,j)，那么它可以走到 $(i-1,j),(i,j-1),(i,j+1)$ 这三个格子中的一个，只要这个目的地在棋盘内且没有障碍。

在一方行动之前，如果发生以下情况之一，则立即结束游戏，按照如下的规则判断胜负（列在前面的优先）：

- 黑棋子位于第一行。此时黑方胜。
- 黑棋子和其中一个红棋子在同一个位置上。此时进行上一步移动的玩家胜。
- 当前玩家不能进行任何合法操作。此时对方胜。

现在假设双方采用最优策略，不会进行不利于自己的移动。也就是说：

- 若存在必胜策略，则会选择所有必胜策略中，不论对方如何操作，本方后续获胜所需步数最大值最少的操作。
- 若不存在必胜策略，但存在不论对方如何行动，自己都不会落败的策略，则会选择任意一种不败策略。
- 若不存在不败策略，则会选择在所有策略中，不论对方如何操作，对方后续获胜所需步数最小值最大的操作。

如果在 $100^{100^{100}}$ 个回合之后仍不能分出胜负，则认为游戏平局。请求出游戏结束时双方一共移动了多少步，或者判断游戏平局。

【输入格式】

从文件 ***zu.in*** 中读入数据。

本题有多组测试数据。

输入的第一行包含两个整数 id, T，分别表示测试点编号和数据组数。特别地，样例的 id 为 0。

接下来包含 T 组数据，每组数据的格式如下：

第一行包含两个正整数 n, m，表示棋盘的行数和列数。

接下来 n 行，每行包含一个长度为 m 的字符串，其中第 i 行的第 j 个字符表示棋盘上 (i, j) 这个位置的状态。

在这些字符中：`'.'` 表示空位；`'#'` 表示障碍物；`'X'` 表示黑棋；`'O'` 表示红棋。

保证黑棋恰好有一个，红棋恰好有两个，且黑棋不在第一行。

【输出格式】

输出到文件 ***zu.out*** 中。

对于每组数据，输出一行字符串。

如果游戏平局，请输出一行 `"Tie"`。

如果红方胜，请输出一行 `"Red t"`。其中 `t` 为游戏结束时双方移动的步数之和。显然这应该是一个奇数。

如果黑方胜，请输出一行 `"Black t"`。其中 `t` 为游戏结束时双方移动的步数之和。显然这应该是一个偶数。

注意，请输出双引号内的字符串，不包含双引号。

【样例 1 输入】

```
0 5
4 5
...#O
.#..#
#O#..
.#..X
3 3
#.#
O.O
.X.
3 3
O..
.#X
.O.
5 5
.....
```

```
.....
..O..
#..#.
O#.X.
9 9
...######
.#.......
.#######.
.#.#.....
.#O#.####
.#.#.....
.#######.
.#X......
.O.......
```

【样例 1 输出】

```
Black 0
Black 2
Black 2
Tie
Red 75
```

【样例 1 解释】

第一组数据，红方第一步没有可行的移动，所以黑方胜。

第二组数据，无论第一步红方怎么移动，黑方都可以在下一步让黑棋子与红棋子在同一个位置。

第三组数据，无论第一步红方怎么移动，黑方都可以将自己的棋子往上移动一格来达成胜利。

第四组数据，有一个红棋子不能动。另一个红棋子可以在第三行移动来防止黑棋子进入第一行。黑棋子也可以一直在第五行移动。如果红棋子到达第五行，黑棋子可以选择从另一边逃走。

第五组数据，在最后一行的那个红棋子可以从左边绕一圈抓住黑棋子。注意另一个红棋子可以移动。

【样例 2】

见选手目录下的 ***zu/zu2.in*** 与 ***zu/zu2.ans***。

【样例 2 解释】

这个样例中的每一组数据都满足测试点 5 到 13 中某一个测试点的限制。

【子任务】

对于所有的数据，保证：$1 \le T \le 10$；$2 \le n \le 10$；$1 \le m \le 10$；id 等于测试点编号。

对于每组数据保证：棋盘上的黑棋恰好有一个，红棋恰好有两个，且黑棋不在第一行。

- 测试点 $1 \sim 4$：保证要么平局，要么红方在开始时无法移动。
- 测试点 $5 \sim 6$：保证 $n \ge 4$。保证棋盘上第 $n-1$ 行的每一个格子都是障碍物，且棋盘上其他行没有障碍物。保证黑棋在前 $n-2$ 行，有一个红棋在前 $n-2$ 行，另一个红棋在第 n 行。
- 测试点 $7 \sim 9$：保证 $m = 1$。
- 测试点 $10 \sim 13$：保证要么平局，要么存在策略可以在 9 步之内结束游戏。
- 测试点 $14 \sim 20$：无特殊限制。

填数游戏（game）

【题目描述】

众所周知，Alice 和 Bob 是一对好朋友。今天，他们约好一起玩游戏。

一开始，他们各自有一张空白的纸条。接下来，他们会在纸条上依次写 n 个 $[1,m]$ 范围内的正整数。等 Alice 写完，Bob **在看到 Alice 写的纸条之后开始写他的纸条。**

Alice 需要保证她写下的第 i 个数在集合 S_i 中，Bob 需要保证他写下的第 i 个数在集合 T_i 中。**题目保证** $1 \le |S_i|, |T_i| \le 2$。

Alice 喜欢相同，因此，她希望她写下的数与 Bob 写下的数对应位置相同的个数尽量多。Bob 喜欢不同，因此，他希望他写下的 n 个数 $b_1, \cdots, b_n$ 互不相同。在此基础上，Bob 希望他写下的数与 Alice 写下的数对应位置相同的个数尽量少。

即设 Alice 写下的数为 $a_1, \cdots, a_n$，Bob 写下的数为 $b_1, \cdots, b_n$，记 X 为满足 $1 \le i \le n, a_i = b_i$ 的下标 i 的个数，则

- Alice 希望最大化 X，
- Bob 在**保证** $b_1, \cdots, b_n$ **互不相同的前提**希望最小化 X。

你首先想知道 Bob 能否保证他写下的 n 个数互不相同。如果 Bob 能够做到，你想知道**在双方均采取最优策略的前提下** X 的值会是多少。

【输入格式】

从文件 ***game.in*** 中读入数据。

本题有多组测试数据。

输入的第一行包含一个正整数 T，表示测试数据组数。

接下来包含 T 组数据，每组数据的格式如下：

第一行包含两个正整数 n, m，表示纸条上需要写的数的个数和数的值域。

接下来 n 行，每行输入的第一个整数为 $|S_i|$ 表示集合 S_i 的元素个数，接下来输入 $|S_i|$ 个正整数描述 S_i 中的元素。

接下来 n 行，每行输入的第一个整数为 $|T_i|$ 表示集合 T_i 的元素个数，接下来输入 $|T_i|$ 个正整数描述 T_i 中的元素。

【输出格式】

输出到文件 ***game.out*** 中。

对于每组测试数据输出一行：若 Bob 无法做到他写下的 n 个数互不相同，输出 -1；否则输出在双方均采取最优策略的前提下 X 的值。

【样例 1 输入】

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

【样例 1 输出】

```
1
```

【样例 1 解释】

在这组样例中，$S_1 = \{3\}, S_2 = T_1 = \{1, 2\}, S_3 = T_3 = \{3, 4\}, T_2 = \{2, 3\}$。
Alice 的填法有 4 种，列举如下：
第一种：$a_1 = 3, a_2 = 1, a_3 = 3$。
第二种：$a_1 = 3, a_2 = 1, a_3 = 4$。
第三种：$a_1 = 3, a_2 = 2, a_3 = 3$。
第四种：$a_1 = 3, a_2 = 2, a_3 = 4$。

由于 Bob 必须保证他所填的数互不相同，所以他有以下填法：
第一种：$b_1 = 1, b_2 = 2, b_3 = 3$。
第二种：$b_1 = 2, b_3 = 3, b_3 = 4$。
第三种：$b_1 = 1, b_2 = 2, b_3 = 4$。
第四种：$b_1 = 1, b_2 = 3, b_3 = 4$。

若 Alice 选择第一种填法，则 Bob 为最小化 X，选择第二种填法，得到 $X = 0$。
若 Alice 选择第二种填法，则 Bob 为最小化 X，选择第一种填法，得到 $X = 0$。
若 Alice 选择第三种填法，则 Bob 为最小化 X，选择第二种填法，得到 $X = 0$。
若 Alice 选择第四种填法，则 Bob 无论选择哪种填法，X 均不小于 1。
因此，Alice 为最大化 X 的值，她会选择第四种填法。

【样例 2】

见选手目录下的 ***game/game2.in*** 与 ***game/game2.ans***。

这个样例满足测试点 1 ∼ 5 的条件。

【样例 3】

见选手目录下的 ***game/game3.in*** 与 ***game/game3.ans***。
这个样例满足测试点 6 的条件。

【样例 4】

见选手目录下的 ***game/game4.in*** 与 ***game/game4.ans***。
这个样例满足测试点 7 的条件。

【样例 5】

见选手目录下的 ***game/game5.in*** 与 ***game/game5.ans***。
这个样例满足测试点 8 的条件。

【样例 6】

见选手目录下的 ***game/game6.in*** 与 ***game/game6.ans***。
这个样例满足测试点 9 的条件。

【样例 7】

见选手目录下的 ***game/game7.in*** 与 ***game/game7.ans***。
这个样例满足测试点 10 ∼ 11 的条件。

【样例 8】

见选手目录下的 ***game/game8.in*** 与 ***game/game8.ans***。
这个样例满足测试点 12 ∼ 13 的条件。

【样例 9】

见选手目录下的 ***game/game9.in*** 与 ***game/game9.ans***。
这个样例满足测试点 23 ∼ 25 的条件。

【子任务】

对于所有的数据，保证：$1 \le T \le 1000$；$1 \le n, m \le 10^6$；$1 \le |S_i|, |T_i| \le 2$；集合 S_i, T_i 都是 $\{1, 2, \cdots, m\}$ 的子集且集合内元素两两不同。

表格中 $\sum n, \sum m$ 分别表示同个测试点内所有测试数据的 n 总和和 m 总和。$\sum n^2, \sum m^2, \sum n^3, \sum m^3$ 的含义类似。

测试点编号	数据范围	特殊性质
1	$T \le 20, n, m \le 10$	
2		
3		
4		
5		
6	$n, m \le 200, \sum n^3, \sum m^3 \le 4 \cdot 10^7$	A
7		B
8		C
9		D
10		
11		
12	$n, m \le 2000, \sum n^2, \sum m^2 \le 4 \cdot 10^7$	
13		
14	$n, m \le 1.5 \cdot 10^5, \sum n, \sum m \le 3 \cdot 10^5$	A
15		B
16		
17		C
18		
19		D
20		
21		
22		
23	$n, m \le 10^6, \sum n, \sum m \le 1.5 \cdot 10^6$	
24		
25		

特殊性质 A：对于任何 $1 \le i \le n$，S_i 和 T_i 互不相交，即 $S_i \cap T_i = \emptyset$。

特殊性质 B：$n \ge 3$，且对于任何 $1 \le i < n$，$T_i = \{i, i+1\}$，且 $T_n = \{n, 1\}$。

特殊性质 C：对于任何 $1 \le i \le n$，$|S_i| = 1$。

特殊性质 D：对于任何 $1 \le i \le n$，$S_i = T_i$。

【提示】

本题部分测试点读入规模较大，我们建议你采取效率较高的读入方式。

染色数组（color）

【题目描述】

给定一个长度为 n 的正整数数组 A，其中每个数都在 1 到 m 之间，从左到右排成一排。现在要将每个数字染成红色或者绿色，我们定义一个染色方案为优秀的染色方案，当且仅当它满足：

1. 每个数 A_i 要么被染成红色，要么被染成绿色。
2. 红色的数从左到右依次严格递增，绿色的数从左到右依次严格递减。

例如：1 9 3 4 7 6 中，将 1 3 4 7 染成红色，9 6 染成绿色是优秀的染色方案（193476）；1 3 4 6 染成红色，9 7 染成绿色也是优秀的染色方案（193476）。但是将 1 4 7 6 染成红色，9 3 染成绿色则**不是**优秀的染色方案，因为 1 4 7 6 不是递增的。1 9 5 5 中，将 1 和任意一个 5 染色红色，9 和另一个 5 染成绿色，也是优秀的染色方案（其中一种是 1955）。

如果一个数组**至少存在两个不同的**优秀的染色方案，那么称这个数组是**完美**的。（两个染色方案不同当且仅当至少存在一个位置上的数字被染成不同的颜色）。

例如，1 9 3 4 7 6 和 1 9 5 5 都是完美的，因为上面已经分别给出了 2 种优秀的染色方案。而 2 3 3 3 则**不是**完美的，因为找不到任何一种优秀的染色方案。同时 1 5 3 6 4 也不是完美的，因为仅存在一种优秀的染色方案（15364）。

补充说明：如果红色的数只有 0 个或者 1 个，我们也认为它严格递增；同理如果绿色的数只有 0 个或者 1 个，我们也认为它严格递减。例如 123，123都是优秀的染色方案，因此 1 2 3 是完美的数组。

我们定义一种给染色方案打分的方式。

对于每个的有序元素对 $A_i, A_j (i < j)$：

1. 如果 A_j 染成红色，且 $A_j < A_i$，则该元素对得 $m - A_j + 1$ 分；
2. 如果 A_j 染成绿色，且 $A_j > A_i$，则该元素对得 A_j 分；
3. 不满足 1 或 2，则该元素对得 0 分。

则一个染色方案的得分为所有有序元素对的得分和。

一个完美的数组的得分为它所有优秀的染色方案的得分的**最大值**。

现在确定数组 A 的前 t 个数 $A_1, A_2, \ldots, A_t$，你需要回答以下两个问题：

- 第一问：有多少种确定 A 中后 $n - t$ 个数的方案使得 A 是一个完美数组？
- 第二问：所有可能的完美数组的得分和是多少？

由于答案太大，你只需要输出答案在模 998244353 下的结果即可。

【输入格式】

从文件 ***color.in*** 中读入数据。

本题有多组测试数据。

输入的第一行包含一个正整数 C，表示数据组数。

接下来包含 C 组数据，每组数据的格式如下：

第一行包含三个整数 n, m, t，分别表示数组长度，数组内数字的最大值和确定的前缀长度。

第二行包含 t 个正整数 $A_1 \dots A_t$，表示已经确定的前缀。

【输出格式】

输出到文件 ***color.out*** 中。

对于每组测试数据输出一行包含两个用单个空格隔开的整数。

- 第一个整数表示优秀数组的数量模 998244353 的值；
- 第二个整数表示优秀数组的得分之和模 998244353 的值。

【样例 1 输入】

```
5
6 10 6
1 9 3 4 7 6
5 8 4
1 7 2 6
9 10 2
3 6
6 11 6
1 7 5 8 3 9
9 10 5
5 10 6 4 7
```

【样例 1 输出】

```
1 63
8 245
29378 1267731
1 17
78 1820
```

【样例 2】

见选手目录下的 ***color/color2.in*** 与 ***color/color2.ans***。

【评分方式】

每个测试点 5 分。

每一行应按顺序输出两问的答案，不符合输出格式的输出得 0 分。

程序仅回答对第一问得 1 分，仅回答对第二问得 4 分，两问都答对得 5 分。

如果你不回答第一问或第二问，也需要在对应位置上输出任意一个整数以满足输出格式。

【子任务】

对于所有的数据，保证：$1 \leq C \leq 5$；$2 \leq n \leq 50$；$1 \leq t \leq n$；$1 \leq m \leq 200$；$1 \leq A_i \leq m$。

测试点编号	n	m	t
$1 \sim 2$	≤ 50	≤ 200	$= n$
$3 \sim 5$	≤ 8	≤ 200	无特殊限制
$6 \sim 7$	≤ 20	≤ 50	无特殊限制
$8 \sim 10$	≤ 25	≤ 70	无特殊限制
$11 \sim 13$	≤ 35	≤ 100	无特殊限制
$14 \sim 16$	≤ 50	≤ 200	$= 1$
$17 \sim 20$	≤ 50	≤ 200	无特殊限制