K и L пак започнаха някакво съревнование. Този път играят игра в двуизмерен лабиринт.

Лабиринтът се състои от n колони и m реда. Всяка клетка е или коридор, или завой. Пионката стои на някоя клетка. Играчът, който е наход, избира една от двете посоки, наляво или нагоре, и може да движи пионката в тази посока. Когато пионката мине през завой, посоката, в която пионката се движи, се обръща към другата и в рамките на хода, тя може да се движи само в новата посока (докато не се стигне друг завой). Играчът може да остави пионката на всяка от валидните клетки, през която тя може да мине по пътя си, стига клетката да е коридор.

Когато някой играч не може да премести пионката, той губи.

В момента L е на ход и ви дава таблицата, както и позицията, на която пионката се намира. Той се чуди, ако и двамата играят оптимално, кой ще спечели. Напишете програма **note.cpp** , която отговаря на въпроса му.

**Вход**

От първия ред на файла **note.in** се въвеждат n и m – броят колони и броят редове. Всеки от следващите n реда съдържа m долепени символа: 0 ако клетката е коридор, 1 ако е завой или L ако пионката е на тази клетка.

**Изход**

На единствения ред във файла **note.out** отпечатайте 1 символ: L ако L ще спечели или K ако K ще спечели.

**Ограничения**

$$1\leq n\*m\leq 10^{6}$$

**Ограничение по време: 0.3 sec.**

**Ограничение по памет: 64 MB**

**Примерен тест**

|  |  |  |
| --- | --- | --- |
| **Вход (note.in)** | **Изход (note.out)** | **Обяснение** |
| 3 30000000L0 | L | L ще премести пионката на клетка (2,2). След това K ще я премести или на (1,2) или на (2,1). И в двата случая след това L ще я премести на (1,1) и K няма да има ход. |
| 2 5010000001L | K | L има две опции: да отиде нагоре и да остави пионката на (1,5) или да отиде наляво. Във втория случай, той ще мине през завоя и ще промени посоката си нагоре, при което трябва да остави пионката на (1,4). И в двата случая K ще премести пионката на (1,3). Тогава L може да се движи само наляво, защото нагоре няма право. Но вляво от него има завой, който ще го обърне нагоре, където пак няма клетка, в която може да отиде. Така че L няма да има ход. |