



## GBOX

Trò chơi lật hộp được mô tả như sau:

Cho một lưới ô vuông có  $n$  dòng và  $m$  cột, các dòng được đánh số từ 0 đến  $n-1$  từ trên xuống dưới, các cột được đánh số từ 0 đến  $m-1$  từ trái sang phải. Ô ở vị trí dòng  $i$ , cột  $j$  gọi là  $(i,j)$  – ô. Tại  $(x,y)$  – ô có đặt một khối hình hộp chữ nhật kích thước  $1*1*3$  (đặt đứng gọn vào  $(x,y)$  – ô). Bạn cần di chuyển hộp bằng cách lật nó trên lưới theo bất kỳ hướng nào (tất nhiên là lật theo 1 cạnh của hộp đang tiếp xúc với lưới, và không để bất kỳ phần nào của đáy hộp nằm ngoài lưới) để đưa hộp đến đặt đứng gọn ở  $(z,t)$  – ô. Tuy nhiên, một số ô của lưới đã bị hư hại và không thể di chuyển hộp vào đó (tức là mặt đáy của hộp phải luôn không nằm đè lên bất cứ ô bị hư hại nào)

Là một game thử giỏi, bạn hãy chỉ ra cách di chuyển hộp sao cho số lần lật hộp là ít nhất (nếu có nhiều phương án cùng có số lần lật hộp ít nhất, đưa ra phương án bất kỳ); hoặc thông báo không có cách nào để di chuyển được hộp theo yêu cầu

Dữ liệu đầu vào nhập từ file GBOX.inp, có cấu trúc:

- Dòng đầu chứa 6 số nguyên không âm  $n,m,x,y,z,t$  ngăn cách bởi 1 dấu cách
- $n$  dòng tiếp theo, mỗi dòng chứa  $m$  ký tự là 0/1 tương ứng là không bị hoặc bị hư hại. Ký tự thứ  $j$  trên dòng thứ  $i$  cho biết  $(i-1,j-1)$  – ô có bị hư hại hay không
- Dữ liệu đảm bảo  $1 \leq n,m \leq 100$ .  $(x,y), (z,t)$  là các ô nằm trên lưới và không bị hư hại

Dữ liệu đầu ra xuất ra file GBOX.out theo cấu trúc:

- Nếu không có cách nào di chuyển hộp theo yêu cầu, in ra “inf” (không có dấu ngoặc kép). Nếu không:
  - Dòng đầu tiên chứa số nguyên dương  $k$ : Số lần ít nhất phải lật hộp
  - Dòng thứ 2 mô tả phương án lật, gồm  $k$  số nguyên thuộc tập  $\{0,1,2,3\}$  tương ứng là lật hộp sang phải, xuống dưới, sang trái, lên trên

Dữ liệu ví dụ:

GBOX.inp	GBOX.out
4 6 0 0 3 5	8
000000	1 0 3 0 1 1 1 0
000001	
000000	
000000	