

# Labirinto

## AdHoc - Médio

Nome do arquivo fonte: `lab.c`, `lab.cpp`, ou `lab.pas`

Um amigo seu está muito empolgado com um novo joguinho que baixou em seu celular. O jogo consiste em uma espécie de labirinto que pode ser representado por um quadriculado de células quadradas com  $N$  linhas e  $M$  colunas. Cada célula do labirinto contém uma plataforma que está a uma determinada altura do chão, que pode ser representada por um inteiro  $a$  que varia de 0 (a mais baixa) a 9 (a mais alta). Você inicia na célula  $(1, 1)$  (canto superior esquerdo) e o objetivo é chegar na saída do labirinto que fica na célula  $(N, M)$  (canto inferior direito).

Para sair do labirinto, você deve fazer movimentos entre células adjacentes. O problema é que seu bonequinho não consegue pular muito alto, então se a célula destino estiver duas ou mais unidades acima da sua altura atual, você não consegue movê-lo. Mais especificamente, a cada turno você pode mover para uma das 4 células adjacentes (cima, baixo, direita, esquerda) *caso a altura da célula destino seja menor ou igual à altura da sua célula atual mais uma unidade*. Ou seja, se a altura da sua célula for  $A$ , você só pode mover para uma célula adjacente caso a altura dela seja menor ou igual a  $A + 1$ .

Para complicar um pouco mais o jogo, a cada turno, *após o jogador realizar sua ação*, cada célula aumenta em uma unidade sua altura, até o valor máximo de 9. Caso a altura de uma determinada célula seja 9, ela passa a ser 0.

Note que, em um dado turno, o jogador não é obrigado a se mover, ele pode simplesmente esperar as plataformas subirem ou descerem. Além disso, repare que nem todas as células têm 4 vizinhos, uma vez que não é permitido ao jogador se mover para fora dos limites do labirinto.

Você, como bom programador que é, resolve escrever um programa que calcule a menor quantidade de turnos possível para chegar à saída de um dado labirinto.

## Tarefa

Escreva um programa que, dado um labirinto, retorne a menor quantidade de turnos necessária para chegar à saída, de acordo com as restrições dadas.

## Entrada

A entrada contém um único conjunto de testes, que deve ser lido do *dispositivo de entrada padrão* (normalmente o teclado). A primeira linha da entrada contém dois inteiros  $N$  e  $M$  ( $2 \leq N, M \leq 50$ ) separados por um espaço em branco, que representam, respectivamente, a quantidade de linhas e colunas do labirinto. As  $N$  linhas seguintes contêm, cada uma,  $M$  inteiros que representam a altura inicial (no turno 0) da respectiva plataforma. As alturas estão sempre entre 0 e 9 (inclusive).

## Saída

Seu programa deve imprimir, na *saída padrão*, uma única linha, contendo a menor quantidade de turnos possível para sair do labirinto.

<b>Entrada</b> 4 3 0 0 0 0 0 0 0 0 0 0 0 0  <b>Saída</b> 5	<b>Entrada</b> 3 3 1 2 3 4 5 6 7 8 9  <b>Saída</b> 12	<b>Entrada</b> 3 5 1 3 1 1 1 1 3 1 3 1 1 1 1 3 1  <b>Saída</b> 10
--	--	--