

## Problema H

# Batalha Naval

Autor: *Leandro Vianna*

Tempo limite: *1 segundo*

*Batalha Naval* é o jogo preferido de Gustavo. Sempre que não está estudando, ele joga partidas de Batalha Naval com seus amigos. Neste jogo, cada participante começa com um certo número de navios posicionados em uma área delimitada. Cada jogador não sabe onde estão posicionados os navios do adversário. A cada jogada, o jogador escolhe uma posição  $(x, y)$  do mapa do adversário, para a qual será “disparado um torpedo”. Caso nesta posição estiver uma parte de um navio, esta parte é danificada. Se todas as partes de um navio são danificadas, o navio é considerado destruído. Vence quem destruir todos os navios do adversário primeiro.

Apesar de toda a prática, Gustavo tem uma dificuldade quando joga as partidas. Sempre que ocorre uma jogada do seu adversário, Gustavo demora a entender quantos de seus navios ainda não afundaram. Ele gostaria de um programa que, dado o mapa inicial com os seus navios e uma sequência de jogadas do seu adversário, respondesse a cada jogada quantos navios ainda não afundaram.

Um mapa de *Batalha Naval* será representado por uma matriz de  $m$  linhas e  $n$  colunas. As linhas são numeradas de 0 à  $m - 1$ , de cima para baixo. As colunas são numeradas de 0 à  $n - 1$ , da esquerda para à direita. Cada elemento dessa matriz é um caractere, e pode ser:

‘.’ : se for apenas mar.

‘\*’ : se for uma parte de um navio de Gustavo.

Duas posições com \* pertencem ao mesmo navio se compartilham um lado. Se duas posições compartilham um canto, elas não pertencem ao mesmo navio (veja o exemplo de entrada 2). Além disso, navios ocupam mais de uma posição apenas em uma mesma linha ou apenas em uma mesma coluna, nunca ao mesmo tempo nas duas. Por exemplo, o mapa da esquerda é um mapa válido. O mapa da direita tem um navio com mais de uma posição em linha e coluna, logo não é válido.

. . . . . * . . . . . * . * * * . * . . . . .	. . . . . . . . . . . * * * * * . . . . * .
--	--

### Entrada:

A primeira linha dos dados de entrada contém dois inteiros  $m$  ( $1 \leq m \leq 1000$ ) e  $n$  ( $1 \leq n \leq 1000$ ), os números de linhas e números de colunas, respectivamente, da matriz que representa o mapa. As próximas  $m$  linhas contém, cada uma,  $n$  caracteres . ou \*, descrevendo uma linha da matriz. A próxima linha contém um inteiro  $j$  ( $1 \leq j \leq 1000$ ), o número de jogadas que serão do adversário. As próximas  $j$  linhas contém, cada uma, dois inteiros  $\ell$  ( $0 \leq \ell \leq m - 1$ ) e  $c$  ( $0 \leq c \leq n - 1$ ), os quais especificam a a posição para a qual o adversário vai disparar o torpedo nesta jogada.

### Saída:

Para cada jogada da entrada, imprimir em uma linha um inteiro  $x$ , a quantidade de navios de Gustavo que não afundaram até o momento.

Exemplo de entrada 1	Exemplo de saída 1
4 6 .....* .....* ..***.* ..... 3 0 5 1 5 2 5	2 2 1

Exemplo de entrada 2	Exemplo de saída 2
6 6 .....* .....* ..***.* ..... ..**.. ...*.. 5 5 3 1 5 2 5 4 1 4 2	3 3 3 3 2