

16^o Treino para alunos da UFPR

26 de Junho de 2015



Flávio Zavan
Ricardo Oliveira

A: Pegman

Arquivo: `pegman.[c|cpp|java|pas]`

Ao usar o *Google Maps*, você já deve ter usado o *Pegman* – um boneco amarelo que ativa o *Street View* no ponto em que é colocado. Hoje, um usuário irá colocar o *Pegman* em uma matriz com R linhas e C colunas. Cada célula desta matriz pode estar em branco ou conter uma seta apontado para alguma das quatro direções (esquerda, direita, cima, baixo).

Se o *Pegman* é colocado em uma célula em branco, ele não se move. Entretanto, se o *Pegman* é colocado em uma célula contendo uma seta, ele muda sua direção para a direção da seta e começa a caminhar. Durante a caminhada, se o *Pegman* encontra uma célula em branco, ele a atravessa e continua caminhando na mesma direção em que está atualmente. Se o *Pegman* encontra uma célula com uma seta, ele muda sua direção para a direção da seta, e continua caminhando na nova direção.

Como você pode observar, o *Pegman* pode ficar caminhando infinitamente. Entretanto, ele também pode sair dos limites da matriz. Sua tarefa é determinar se é possível *alterar a direção* de algumas setas ¹ de tal forma que o *Pegman* nunca saia dos limites da matriz, *independentemente* da célula em que é colocado inicialmente. Se isto for possível, determine também o menor número de setas que necessitam ser alteradas.

Entrada

A primeira linha contém dois inteiros R e C ($1 \leq R, C \leq 100$). As próximas R linhas contém C caracteres cada, descrevendo a matriz. Os caracteres `<`, `>`, `^` (circunflexo), `v` e `.` indicam uma seta para esquerda, direita, cima, baixo e uma célula em branco, respectivamente.

Saída

Imprima uma linha contendo o menor número de setas a serem alteradas, ou a palavra `IMPOSSIBLE` se não for possível configurar a matriz como se pede.

Exemplo de entrada	Exemplo de saída
2 1 ^ ^	1

Exemplo de entrada	Exemplo de saída
2 2 >v ^<	0

¹Cada seta pode ter sua direção alterada para as outras três direções conhecidas

Exemplo de entrada	Exemplo de saída
3 3^. ...	IMPOSSIBLE

Exemplo de entrada	Exemplo de saída
1 1 .	0

B: Pikachu em ASCII

Arquivo: pikachu.[c|cpp|java|pas]

A criatividade dos calouros dos cursos de informática é incrível! Esta semana alguns deles criaram um programa capaz de imprimir uma arte ASCII do famoso pokémon Pikachu. O desenho ficou tão bom que a imagem está bombando nas redes sociais da universidade!

Neste problema, considere que o Pikachu foi desenhado em uma matriz de N linhas e M colunas. Na matriz deste problema, todos os pixels têm ou a cor preta ou a cor branca. Após desenhar o pokémon, os calouros agora querem contar os *espaços de indentação* do desenho. São considerados espaços de indentação todos os pixels brancos de uma linha que ocorrem antes do primeiro pixel preto da linha, assim como todos os pixels brancos de uma linha que ocorrem depois do último pixel preto da mesma. Se uma linha contém apenas pixels brancos, então todos são considerados espaços de indentação.

No primeiro exemplo de entrada², há 4 espaços de indentação na primeira linha, 3 na segunda, 1 na terceira e 1 na quarta, totalizando 9 espaços ao todo.

Entrada

A primeira linha da entrada contém os inteiros N e M ($1 \leq N, M \leq 100$). As próximas N linhas contém M inteiros cada, todos iguais a 0 ou 1. O número 0 indica um pixel branco, enquanto o número 1 indica um pixel preto.

Saída

Imprima uma única linha contendo o número total de espaços de indentação no desenho.

Exemplo de entrada	Exemplo de saída
<pre>4 5 0 0 1 0 0 0 1 1 0 0 0 1 0 0 1 0 1 1 1 1</pre>	<pre>9</pre>

Exemplo de entrada	Exemplo de saída
<pre>3 3 1 1 1 1 0 1 1 1 1</pre>	<pre>0</pre>

²que claramente não é um Pikachu completo



C: High Five

Arquivo: `high.[c|cpp|java|pas]`

A seleção Nlogonense de Basquetebol está na grande final da Copa do Mundo de Basquete, e o jogo está prestes a começar! Neste momento, os N atletas da seleção se preparam para entrar em quadra.

Os atletas são numerados de 1 a N . Inicialmente, os atletas estão em uma fila na beira do campo. Os atletas entram no campo um por um, na ordem de seus números. Assim, o atleta 1 é o primeiro a sair da fila e entrar na quadra. Em seguida, o atleta 2 sai da fila e entra na quadra, e assim por diante.

Ao sair da fila, cada atleta cumprimenta todos os atletas que estão em sua frente na fila com um *high five*. Por exemplo, considere que $N = 4$ e inicialmente a fila é 3 1 2 4, onde o atleta 4 é o mais próximo da quadra. O atleta 1, ao ir para a quadra, faz um *high five* com os atletas 2 e 4. O atleta 2, por sua vez, faz um *high five* apenas com o atleta 4 ao sair da fila. O atleta 3 também faz um *high five* apenas com o atleta 4 (note que os atletas 1 e 2 já saíram da fila). Por fim, o último atleta não cumprimenta ninguém.

Sua tarefa é determinar o número total de *high fives* que serão feitos.

Entrada

A primeira linha contém o inteiro N ($1 \leq N \leq 10^5$). A segunda linha contém N inteiros, indicando a ordem em que os atletas estão inicialmente na fila. O último inteiro presente na linha indica o atleta mais próximo da quadra.

Saída

Imprima uma linha contendo o número total de *high fives* que serão feitos.

Exemplo de entrada	Exemplo de saída
4 3 1 2 4	4

Exemplo de entrada	Exemplo de saída
5 5 4 3 2 1	0

Exemplo de entrada	Exemplo de saída
5 1 2 3 4 5	10

D: Cacheiro Viajante

Arquivo: `cacheiro.[c|cpp|java|pas]`

Clóvis Eduardo é um cacheiro viajante, ou seja, ele viaja o mundo para praticar *geocaching*. *Geocaching* é um jogo mundial de GPS no qual os jogadores encontram caixas escondidas (conhecidas como *geocaches*) por outros jogadores fazendo uso das coordenadas terrestres.

Devido a sua grande reputação e experiência (mais de 32768 *geocaches* encontrados em todos os continentes), Clóvis Eduardo foi convidado pelo ministério de Lazer e Esporte da Nlogônia a ajudar a esconder o *geocache* mais importante da história do país, uma grande caixa de ouro maciço em forma de cavalo, em comemoração aos 50 anos da instauração da república na Nlogônia.

Entretanto, não basta apenas encontrar o local perfeito. Novos *geocaches* devem seguir as regras nacionais de colocação atualizadas este ano:

- Nenhum *geocache* novo pode ser colocado no mesmo lugar de outro já existente;
- A soma das distâncias aos K *geocaches* já existentes mais próximos deve ser igual ou superior a D ;
- Se existirem menos de K *geocaches*, deverão ser considerados todos os existentes na soma das distâncias.

A Nlogônia é representada por um plano e, atualmente, possui N *geocaches* ativos. O ministério de Lazer e Esporte escolheu C possíveis localizações para a colocação do *geocache* comemorativo e pediu para que Clóvis Eduardo os informasse quantas destas localizações são viáveis de acordo as regras nacionais de colocação.

Como Clóvis Eduardo é preguiçoso, ele pediu para que você escrevesse um programa para resolver o seu problema.

Entrada

A primeira linha contém quatro inteiros separados por espaços N , C ($1 \leq N, C \leq 10^4$), K ($1 \leq K \leq 8$) e D ($1 \leq D \leq 10^5$).

As próximas N linhas contém dois inteiros separados por um espaço, x_i e y_i ($0 \leq x_i, y_i \leq 10^4$), a posição (em metros a partir da origem) do i -ésimo *geocache* já existente na Nlogônia. Não há posições repetidas.

As próximas C linhas contém dois inteiros separados por um espaço, x_j e y_j ($0 \leq x_j, y_j \leq 10^4$), a j -ésima possível posição (em metros a partir da origem) do novo *geocache*.

Saída

Imprima exatamente uma linha com o número de posições válidas sugeridas pelo Ministério de Lazer e Esporte.

Exemplo de entrada	Exemplo de saída
10 3 3 10 0 0 10 10 0 10 5 9 15 18 3 10 2 2 2 4 3 8 4 5 1 1 0 0 100 100	1

E: Feira de Adoção

Arquivo: `adocao.[c|cpp|java|pas]`

Carmem Lucrécia é uma pessoa muito solitária, todo sábado de manhã ela passeia sozinha pelas ruas do centro da capital da Nlogônia. A ocasião sempre é aproveitada para pensar na vida e no quão frio os Nlogonenses são.

Desta vez a caminhada não foi como sempre, haviam carros de som e grandes faixas anunciando a XVIII Feira Nacial de Adoção de Filhotes da Nlogônia, evento o qual mudaria a vida de Carmem Lucrécia para sempre.

Decidida a encontrar um companheiro canino, Carmem Lucrécia logo entrou na feira e começou a avaliar as opções.

Há exatamente N filhotes para adoção. Entretanto, Carmem Lucrécia é muito chata e tem uma série de exigências para seu novo companheiro:

- O animal deve ser da espécie **cachorro**;
- O animal deve ter nome composto;
- Pelo menos uma das palavras do nome do animal deve começar com o mesmo caractere do nome de sua raça.

O atendente da feira ficou tão confuso com as exigências de Carmem Lucrécia e pediu para que você fizesse um programa para ajudá-lo.

Entrada

A primeira linha contém o inteiro N ($1 \leq N \leq 100$), o número de filhotes para adoção. As $4N$ linhas seguintes descrevem os filhotes.

Cada filhote é descrito por três linhas. A primeira linha contém a espécie do animal, a segunda, a raça e a terceira, o nome. Cada descrição é seguida de uma linha em branco.

Todas as linhas contém apenas letras minúsculas, número e espaços. Nenhuma linha tem mais de 100 caracteres. Nenhuma começa nem termina com espaços e não há espaços consecutivos.

Saída

Imprima uma linha contendo o número total de filhotes que cumprem as exigências de Carmem Lucrécia.

Exemplo de entrada	Exemplo de saída
7 lobo guara do mato robson cachorro schnauzer schnepps schnitzel cachorro vira latas valdecir cachorro vira latas wanderson luiz gato siames soraia maria cachorro pastor alemao luan paulo cachorro bulldog luciano fabio romario mateus astrogildo bruno bianca	3