

# CAIXEIRO VIAJANTE UNIDIRECIONAL

Nome do arquivo fonte: caixeiro. .[clcpp|java]

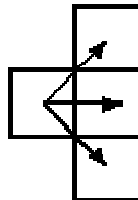
## Background

Problemas que exigem caminhos mínimos em algum domínio aparecem em diversas áreas da Ciência da Computação. Por exemplo, uma das restrições em problemas de roteamento de VLSI é minimizar o comprimento das trilhas. O Problema do Caixeiro Viajante (TSP) - descobrir se todas as cidades na rota de um vendedor podem ser visitadas apenas uma vez com um limite especificado no tempo de viagem - é um dos exemplos canônicos de um problema NP-completo; soluções parecem exigir uma enorme quantidade de tempo para serem geradas, mas são fáceis de verificar.

Esse problema lida com encontrar um caminho mínimo através de uma grade de pontos durante a viagem somente de esquerda para a direita.

## O Problema

Dada uma matriz  $m \times n$  de inteiros, você deve escrever um programa que calcula um caminho de peso mínimo. Um caminho começa em qualquer lugar na coluna 1 (primeira coluna) e consiste de uma sequência de passos que terminam na coluna  $n$  (última coluna). Um passo consiste em viajar de coluna  $i$  para coluna  $i + 1$  em uma linha (horizontal ou diagonal) adjacente. As primeiras e últimas linhas (linhas 1 e  $m$ ) de uma matriz são considerados adjacentes, ou seja, a matriz se “enrola” tal que ela representa um cilindro horizontal. Passos legais são ilustrados abaixo.



O peso de um caminho é a soma dos números inteiros em cada uma das  $n$  células da matriz que são visitadas. Por exemplo, duas matrizes 5x6 ligeiramente diferentes são mostradas abaixo (a única diferença são os números da linha inferior).

3	4	1	2	8	6
6	1	8	2	7	4
5	9	3	9	9	5
8	4	1	3	2	6
3	7	2	8	6	4

3	4	1	2	8	6
6	1	8	2	7	4
5	9	3	9	9	5
8	4	1	3	2	6
3	7	2	1	2	3

O caminho mínimo está ilustrado em cada matriz. Note que o caminho para a matriz da direita tira proveito da propriedade adjacência das primeira e última fileira.

## Entrada

A entrada consiste de uma especificação da matriz contendo as dimensões  $m$  e  $n$  onde  $m$  é a dimensão da linha e  $n$  é a dimensão da coluna, seguida por  $m \cdot n$  inteiros. Os  $n$  primeiros inteiros constituem a primeira linha da matriz, os próximos  $n$  inteiros constituem a segunda linha e assim por diante. Os inteiros em uma linha são separados de outros inteiros por um ou mais espaços. Nota: números inteiros não são necessariamente positivos.

Para cada especificação, o número de linhas será entre 1 e 10, inclusive; o número de colunas irá estar entre 1 e 100 inclusive. Nenhum peso de caminho irá exceder os valores inteiros representáveis utilizando 30 bits.

## Saída

Duas linhas devem ser mostradas para cada especificação de matriz. A primeira linha representa o caminho de peso mínimo, e a segunda é o custo de um caminho mínimo. O caminho é composto por uma sequência de  $n$  inteiros (separados por um espaço) representando as linhas que constituem o caminho mínimo. Se houver mais do que um caminho de peso mínimo, o caminho que é lexicograficamente menor deve ser impresso.

## Exemplos

Exemplos de entrada	Exemplos de saída
5 6 3 4 1 2 8 6 6 1 8 2 7 4 5 9 3 9 9 5 8 4 1 3 2 6 3 7 2 8 6 4	1 2 3 4 4 5 16
5 6 3 4 1 2 8 6 6 1 8 2 7 4 5 9 3 9 9 5 8 4 1 3 2 6 3 7 2 1 2 3	1 2 1 5 4 5 11
2 2 9 10 9 10	1 1 19