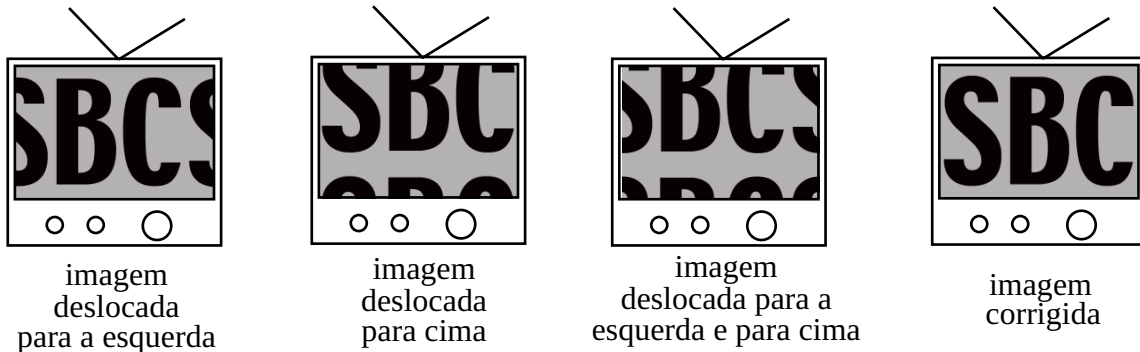


TV da Vovó

Arquivo fonte: *tv.c*, *tv.cc*, *tv.cpp* ou *tv.pas*

A vovó tem um televisor muito antigo, que ultimamente está exibindo um defeito incômodo: a imagem aparece ‘deslocada’ (para cima ou para baixo, para o lado direito ou para o lado esquerdo). Quando a imagem está deslocada para cima, a parte da imagem que deixa de ser vista na parte superior reaparece na parte de baixo da tela. Da mesma forma, quando a imagem está deslocada a direita, a parte da imagem que deixa de ser vista à direita reaparece na tela do lado esquerdo.



A imagem do televisor pode ser vista como uma matriz de pontos organizados em linhas e colunas. Para consertar o televisor da vovó, você pode ajustar a imagem introduzindo uma série de ‘comandos de correção’ em um painel de ajuste. Cada comando de correção desloca a imagem de um certo número de linhas (para cima ou para baixo) e um certo número de colunas (para a direita ou para a esquerda).

1. Tarefa

Dada uma matriz que representa uma imagem defeituosa e uma série de comandos de correção, seu programa deve calcular a matriz que representa a imagem resultante após todos os comandos terem sido aplicados sequencialmente.

2. Entrada

A entrada possui vários conjuntos de teste. Cada conjunto de teste inicia com a descrição da matriz que representa a imagem do televisor. A primeira linha contém dois inteiros M e N representando o número de linhas e o número de colunas da matriz ($1 \leq M \leq 1000$ e $1 \leq N \leq 1000$). As M linhas seguintes da entrada contém cada uma N inteiros, descrevendo o valor de cada ponto da imagem. Após a descrição da imagem, segue-se a descrição dos comandos de correção. Cada comando de correção é descrito em uma linha contendo dois inteiros X e Y . O valor de X representa o deslocamento na direção horizontal (valor positivo representa deslocamento para a direita, valor negativo para a esquerda), e o valor de Y representa o deslocamento da direção vertical (valor positivo para cima, valor negativo para baixo). O final da lista de comandos é indicado por $X = Y = 0$, e o final da entrada é indicado por $M = N = 0$.

Exemplo de Entrada

```
3 3
1 2 3
4 5 6
7 8 9
1 0
1 -1
0 0
3 4
6 7 8 5
10 11 12 9
2 3 4 1
-3 2
0 0
0 0
```

3. Saída

Para cada conjunto de teste, o seu programa deve produzir uma imagem na saída. A primeira linha da saída deve conter um identificador do conjunto de teste, no formato “Teste n ”, onde n é numerado seqüencialmente a partir de 1. A seguir deve aparecer a matriz que representa a imagem resultante, no mesmo formato da imagem de entrada. Ou seja, as N linhas seguintes devem conter cada uma M inteiros que representam os pixels da imagem. Após a imagem deixe uma linha em branco. A grafia mostrada no Exemplo de Saída, abaixo, deve ser seguida rigorosamente.

Exemplo de Saída

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

Teste 2
1 2 3 4
5 6 7 8
9 10 11 12
```

(esta saída corresponde ao exemplo de entrada acima)

4. Restrições

$0 \leq N \leq 1000$ ($N = 0$ apenas para indicar o final da entrada)
 $0 \leq M \leq 1000$ ($M = 0$ apenas para indicar o final da entrada)
 $0 \leq X \leq 1000$
 $0 \leq Y \leq 1000$
 $0 \leq \text{número de comandos de correção em cada conjunto de teste} \leq 1000$