

Blefe

Nome do arquivo fonte: `blefe.c`, `blefe.cpp`, `blefe.pas`, `blefe.java`, ou `blefe.py`

Pedro está desenvolvendo um jogo on-line para dois jogadores, em que o objetivo é forçar um erro do adversário, blefando. A questão é que, à medida que o jogo prossegue, mais tempo é necessário para verificar se uma jogada é válida ou não, ou seja, se é um blefe ou não. Daí que Pedro precisa da sua ajuda para implementar um algoritmo rápido para verificar se uma jogada é ou não um blefe.

Considere um conjunto A fixo de N números inteiros, positivos ou negativos, e uma sequência de números inteiros B , inicialmente vazia. Os jogadores se alternam em jogadas que consistem em incluir um número por vez no final da sequência B . Quando chega a sua vez, um jogador deve fazer uma de duas jogadas válidas possíveis: (i) incluir em B qualquer um dos números do conjunto A ; (ii) ou incluir em B um número que é a soma de dois números quaisquer que já estejam em B (note a soma não é de números necessariamente distintos, pode ser a soma de um número com ele mesmo).

Nesta tarefa, você deve escrever um programa que, dado o conjunto A e uma sequência B , diga se todas as jogadas foram válidas, ou mostre qual é a primeira jogada inválida em B .

Entrada

A entrada consiste de três linhas. A primeira linha contém dois números N e M , respectivamente o tamanho do conjunto A e o tamanho da sequência B . A segunda linha contém os N números inteiros do conjunto A . A terceira linha contém os M números inteiros da sequência B .

Saída

Seu programa deve produzir uma única linha. A linha deve conter a palavra “**sim**” caso todas as jogadas em B sejam válidas; se houver alguma jogada inválida em B , a linha deve conter o primeiro número inválido em B .

Restrições

- $1 \leq N \leq 10^3$ e $1 \leq M \leq 10^4$
- O valor de todos os números em A e em B está entre -10^9 e 10^9

Informações sobre a pontuação

- Em um conjunto de casos de teste equivalente a 30 pontos, $N \leq 50$ e $M \leq 500$.

Exemplos

Entrada	Saída
6 11 34 9 -2 77 -11 5 34 5 -2 32 -11 -6 28 66 -2 -22 33	sim

Entrada	Saída
6 8 34 9 -2 77 -11 5 -11 77 -2 75 9 48 7 5	48