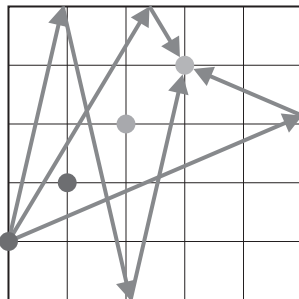


Adott egy  $N \times N$ -es négyzet alakú biliárdasztal, amelynek szélén a golyók tökéletesen rugalmasan ütközhetnek. A golyók kiterjedés nélküliek és kezdetben az asztal egész koordinátájú pontjaiban állhatnak.

Készítsünk programot `i298` néven, amely megadja és a standard kimenetre kiírja, hogy egy adott  $(k; l)$  koordinátájú pontból gurítva egy golyót, azzal egy másik adott  $(x; y)$  koordinátájú golyót hányféle úton találhatunk el.



*Példa három lehetséges útra*

A golyó útja akkor érvényes, ha legfeljebb két falon pattan vissza anélkül, hogy a megcélzott golyón kívül más golyót érintett volna. Ha egy golyó pont az asztal sarkát találja el, akkor önmagába verődik vissza, mivel ez mindkét falról történő visszapattanásnak számít. A program eredménye – az asztal szélén történő visszapattanások száma szerint – a kétpattanásos, az egypattanásos és a pattanás nélküli utak száma.

A program parancssori argumentuma legyen a kezdőfeltételeket leíró adatállomány neve. A fájl első sorában  $N$  ( $2 \leq N \leq 50$ ) az asztal méretét,  $M$  ( $1 < M \leq 20$ ) a golyók számát adja meg. Az ezt követő  $M$  sor a golyók koordinátáit, majd az utolsó sor a golyóindítás  $(k; l)$  koordinátáit ( $1 \leq k, l \leq N$ ), és a célgolyó  $(x; y)$  koordinátáit írja le. A biliárdasztal bal alsó sarka legyen az  $(1; 1)$  koordinátájú pont, és az első koordináta jobbra, a második koordináta felfelé nő.

*Példa* (lásd az *ábrát*):

Bemenet	Kimenet
6 4	Két pattanas: 8
1 2	Egy pattanas: 3
2 3	Nulla pattanas: 0
3 4	
4 5	
1 2 4 5	

Beküldendő a program forráskódja (`i298.pas`, `i298.cpp`, ...) és rövid dokumentációja (`i298.txt`, `i298.pdf`, ...), amely tartalmazza a megoldás rövid leírását, és megadja, hogy a forrásállomány melyik fejlesztői környezetben fordítható.