## Задача 3 – Овчар

Бат ви Жоро от Каспичан решил да става овчар, защото дочул, че било по-доходоносна професия от програмирането (така става като вярваш на телевизията). Купил си овце и един коч (мъжкото на овцата). Направил си и огради, в които да ги пази да не бягат. Някои огради били оградени от други огради, някои се пресичали, но вършели работа да пречат на овцете да се разбягват. Жоро даже си направил карта на оградите под формата на двумерна координатна система.

Всичко било наред до момента, в който Ники (кочът на Жоро), , решил да прави революция. Разбягал се наляво-надясно, заблъскал оградите, и успял да избяга. Някои от другите овце видели това и също се разбягали (овце са, като видят нещо – повтарят).

Сега Жоро се е видял в чудо. Решил да подходи методично и първо да преброи колко от овцете са останали в оградите, за да си направи сметка дали си заслужава да събира останалите. После се сетил, че ако това му се случва често, ще е добре да има програма, която да брои вместо него. За времето в което е бил овчар, Жоро е забравил как се програмира и има нужда от вашата помощ.

Всички огради на Жоро са правоъгълници и имат страни успоредни на осите на координатната система на картата. Една ограда се описва с координатите на 2 срещуположни ъгъла на тази ограда. Т.е. оградата с ъгли (0, 0), (5, 0), (5, 5), (0, 5) може да се опише с върховете (0, 0) и (5, 5), както и с върховете (0, 5), (5, 0).

Местоположението на всяка една овца е точка в двумерната координатна система. Казваме, че една овца не е избягала, ако местоположението ѝ е вътре в поне една ограда (Забележка: ако местоположението на овцата се намира върху страна или връх на тази ограда, тя не се счита „вътре“ в оградата – овца качила се на оградата се счита за избягала).

Напишете програма, която по зададено множество от огради и местоположения на овце намира колко от тези овце не са избягали.

### Вход

Входните данни се четат от стандартния вход (конзолата).

На първия ред се намират целите числа **F** и **S** – съответно броят огради и броят овце.

На следващите **F** **реда** има **по 4 числа** – съответно 2-те координати (X и Y) на първия връх и 2-те координати (X и Y) на втория връх описващи оградата.

На следващите **S реда** има **по 2 числа** – координатите (X и Y) на местоположението на всяка една овца.

Входните данни ще са винаги валидни и в описания формат.

### Изход

Изходните данните трябва да се изведат на стандартния изход (конзолата).

На единствения ред на изхода трябва да се отпечата едно число – броят на овцете, които не са избягали.

### Ограничения

* 0 < F < 101
* 0 < S < 101
* Координатите на точките (върхове и местоположения) са реални числа в интервала [-1000, 1000]
* Разрешено време за работа на програмата: 0.10 секунди. Разрешена памет: 16 MB.

### Примери

|  |  |  |
| --- | --- | --- |
| **Примерен вход** | **Примерен изход** | **Обяснение** |
| 1 3  30 15 -78 95  -24 55  88 -92  -14 95 | 1 | Единствената ограда тук е правоъгълник с координати (-78, 95), (30, 95), (30, 15), (-78, 15). В него се съдържа само една овца – тази с координати (-24, 55) |
| 3 1  10 48 -21 -71  -12 -18 81 19  63 -47 94 92  97 -41 | 0 | Единствената овца тук не се съдържа в никой правоъгълник, описващ ограда, т.е. тя е избягала и броят на овците в огради е 0 |