

Jogos Olímpicos

Por Maratona de Programação da SBC 2016  Brazil**Timelimit: 1**

Um grupo de investidores está pensando em investir pesado em atletas da delegação brasileira após as olimpíadas do Rio. Para isso, eles vêm observando N atletas e perceberam que alguns estão em decadência e outros em ascensão. Em especial, o grupo está de olho em dois fatores sobre cada atleta: seu cansaço e sua habilidade. Eles anotaram os valores de habilidade e cansaço de cada atleta logo ao final das olimpíadas de 2016. Em seguida, o grupo estimou a taxa com a qual cada atleta perde ou ganha habilidade e a taxa com a qual cada atleta se cansa ao longo do tempo, e percebeu que essas taxas são constantes para os dois atributos.

Os investidores perceberam que esses dados lhes permitem definir o que resolveram chamar de atleta de ouro: um atleta que, em um determinado período de tempo, é o atleta menos cansado e o mais habilidoso. Ficou decidido que investimentos serão feitos apenas em atletas de ouro. Descubra quantos jogadores, entre os observados inicialmente, receberão algum investimento. Considere que o tempo $t = 0$ é o tempo das olimpíadas do Rio: nenhum atleta que foi de ouro antes desse tempo pode receber investimento. Considere também que qualquer tempo após as olimpíadas do Rio deve ser considerado, por maior que seja. Um atleta que é de ouro exatamente no tempo $t = 0$ deve ser contado.

Entrada

A primeira linha da entrada contém um inteiro, N ($1 \leq N \leq 10^5$), o número de atletas. Seguem N linhas, cada uma com quatro números inteiros: H_i, H_i^t, C_i, C_i^t ($-10^6 < H_i, H_i^t, C_i, C_i^t \leq 10^6, H_i^t, C_i^t \neq 0$), representando, respectivamente, a habilidade ao final das olimpíadas, a taxa de variação da habilidade, o cansaço ao final das olimpíadas e a taxa de variação do cansaço do i -ésimo atleta.

Saída

Seu programa deve produzir uma única linha com um inteiro O , representando o número de atletas que receberão algum investimento do grupo.

Exemplos de Entrada	Exemplos de Saída
3 3 2 1 2 2 2 2 2 1 2 3 2	1
6 1 10 5 8 8 7 12 -5 10 -2 -3 8 -3 -5 -8 -12 0 1 10 2 8 3 9 -3	0