Project Sekai Event Planner

SEEM & sahuang*

1 Problem Statement

Project Sekai Event: Wishing You Happiness Upon the Blue Sky

Project Sekai: Colorful Stage! feat. Hatsune Miku is a Japanese mobile rhythm game featuring Vocaloid-based artists. As you may know, *Project Sekai CTF* has its namesake originally derived from this game.

In *Project Sekai*, events are periods of 7 to 10 days where players earn event points and event badges for completing a live[1]. Each event is accompanied by a new story and a new gacha. For each event, there are several ranks ranging from "1st Place" to "Top 100000". During the event, players need to compete against each other and gain as many event points as possible. Of course, it can be extremely time and money consuming[2] to claim the top ranks.

sahuang has designed a customized event planner for *Project Sekai*. Competitive programmers often think differently - instead of maximizing event points, the planner is attempting to solve a completely different problem described below.

*Authored by `SEEM`. Problem statement and translation prepared by `sahuang`.
[1] https://www.sekaipedia.org/wiki/Events
[2] See the Tweet from `stypr`, *Project Sekai CTF* Team Ambassador

Suppose each event consists of n days and the game has m songs labeled from 0 to $m-1$. The planner arranges several rounds per day, allowing players to rest in between. On the i^{th} day, $r_i - l_i + 1$ rounds are scheduled.

To minimize accuracy loss and point deductions from switching songs, the planner ensures that the same song is played within each round. Consequently, in j^{th} round ($0 \leq j \leq r_i - l_i$), only the song labeled $k(j + l_i) + b_i \pmod m$ will be played a_i times.

Once all operations are completed, you need to select p different songs with **non-adjacent labels**. The *value* of each selection is defined by **multiplying the number of times each of the p songs is played.**

Compute the sum of the *values* for all possible selections, modulo $10^9 + 7$.

2 Input

The first line of input contains four space-separated integers n ($1 \leq n \leq 7,000$), m ($1 \leq m \leq 10^{18}$), k ($0 < k < m$), p ($1 \leq p \leq 4$), which represents number of days, number of songs, and the two planner parameters as described before.

The next n lines describe the operation parameters for each day. The i^{th} line contains four space-separated integers: l_i, r_i ($0 \leq l_i \leq r_i < m$), b_i ($0 \leq b_i < m$), and a_i ($0 \leq a_i < 10^9 + 7$) respectively.

3 Output

Sum of the *values* for all possible selections modulo $10^9 + 7$.

Note: In the final result, two selections with the same set of songs but in different order will be counted only once.

4 Samples

Sample Input 1	Sample Output 1
2 5 3 2 1 3 2 2 2 3 1 1	13

Sample Input 2	Sample Output 2
3 1000000 89 4 2 222 19 2 4 66666 1 9 5 114514 8 10	299126098

5 Explanation

5.1 Sample 1

The event has 2 days and there are 5 songs in total.

On day 0, 3 rounds are played:

- Round 1: Song 0 is played 2 times
- Round 2: Song 3 is played 2 times
- Round 3: Song 1 is played 2 times

On day 1, 2 rounds are played:

- Round 1: Song 2 is played 1 time
- Round 2: Song 0 is played 1 time

After the event, 5 songs are played $3, 2, 1, 2, 0$ times respectively. To select 2 non-adjacent songs, all possible indexes are $(0, 2)$, $(0, 3)$, $(0, 4)$, $(1, 3)$, $(1, 4)$, $(2, 4)$. Their sum is $3 + 6 + 0 + 4 + 0 + 0 = 13$.

Note that $(2, 0)$ and $(0, 2)$ are considered identical and only counted once.