

Program Verification II

Translated by GPT 5

Problem Background

Did you know that His Majesty Emperor `__int128` is actually a huge fan of computer games?

One day, the Emperor's computer refused to boot, because the core of the Flea Kingdom's Internet infrastructure — the CloudFlea service — suddenly exploded.

The Emperor `__int128` ordered that every working computer in the whole country be collected, but all of them had crashed. In the end, only a single six-state Turing machine belonging to `ccz181078` (rating **1658**) was found.

However, this Turing machine is currently running, so it can only be used after it halts.

You need to help Emperor `__int128` speed up the computation of this Turing machine — the faster, the better!

Problem Description

Inside `ccz181078`'s Turing machine, there are n pre-stored operations

$$(x_i, y_i, a_i, b_i) \quad (1 \leq i \leq n),$$

and the machine needs to process m queries.

For each query, you are given l, r, X . The Turing machine performs the following procedure:

- Initialize $A = 1, B = 0$.
- For every integer i from l to r in increasing order:
 - If $X \leq x_i$, then update

$$X \leftarrow X + y_i, \quad A \leftarrow a_i A, \quad B \leftarrow a_i B + b_i.$$

- In the end, you must output

$$X, \quad (A \bmod 2^{32}), \quad (B \bmod 2^{32}).$$

Input Format

The first line contains two integers n and m .

Each of the next n lines contains four integers x_i, y_i, a_i, b_i .

Each of the next m lines contains three integers l', r', X' describing a query *in encrypted form*.

Since the Turing machine must finish processing one query before it can process the next, the problem is **strictly online**. You must **decrypt** l', r', X' using the answers from the previous query to obtain the actual parameters l, r, X :

- Let A', B' be the values of $(A \bmod 2^{32})$ and $(B \bmod 2^{32})$ from the **previous** query.
- For the **first** query, define $A' = B' = 0$.
- For each query, you must apply, **in this order**, to each of l', r', X' :
 - 1) First truncate the value to the least significant **32** bits.
 - 2) Then XOR with A' .
 - 3) Then XOR with B' .

The resulting three integers are the actual l, r, X .

Important: The encrypted values l', r', X' **before** decryption may lie outside the range of 32-bit signed integers.

Output Format

Output m lines.

For each query, output three integers X, A, B (after finishing the operations for that query) on a single line, representing the result of that query.

Sample 1

Input

```
10 10
80 1 8 9
47 2 8 5
30 1 0 0
51 0 2 10
11 -3 1 7
87 -3 0 4
11 -1 0 0
11 2 10 8
11 0 7 6
49 2 5 0
8 10 26
3 15 28
16 29 29
57 52 94
0 8 38
6 14 21
21 30 52
21 29 26
1 13 39
5 12 76
```

Output

```
28 5 0
24 0 20
4 0 62
96 1 0
39 0 4
19 0 20
34 0 20
15 0 4
32 0 4
70 0 4
```

Decrypted Input

```
10 10
80 1 8 9
47 2 8 5
30 1 0 0
51 0 2 10
11 -3 1 7
87 -3 0 4
11 -1 0 0
11 2 10 8
11 0 7 6
49 2 5 0
8 10 26
6 10 25
4 9 9
7 10 96
1 9 39
2 10 17
1 10 32
1 9 14
5 9 35
1 8 72
```

Sample 2

Input

```
20 20
83 -2 6 1
62 -3 0 5
79 -2 5 9
3 4 6 3
4 1 0 9
19 1 9 0
71 -3 8 2
87 -4 5 5
2 -1 2 3
23 3 7 4
44 -3 7 7
16 1 0 7
2 2 8 10
66 -1 10 7
25 -2 9 8
0 4 6 1
24 3 4 7
55 -1 5 9
8 -1 10 8
9 0 3 10
8 15 56
10 27 65
14492 14473 14589
7 16 26
982964 982973 983021
0 14 26
702 693 645
446 433 488
13 20 22
701 675 679
689 672 745
7 15 83
9 20 1
21641566 21641542 21641562
421957 421969 421982
2162314 2162309 2162385
3 14 81
1 14 94
7 18 24
2162308 2162331 2162327
```

Output

```
51 50 57
62 12000 5757
96 1 0
20 705600 277491
94 1 0
11 0 701
47 200 375
83 5 5
21 1800 1464
22 1800 1464
89 1 0
78 5 5
5 0 21641554
7 0 421954
20 3528000 1387464
89 1 0
76 5 5
94 1 0
17 3528000 1387464
29 50 44
```

Decrypted Input

```
20 20
83 -2 6 1
62 -3 0 5
79 -2 5 9
3 4 6 3
4 1 0 9
19 1 9 0
71 -3 8 2
87 -4 5 5
2 -1 2 3
23 3 7 4
44 -3 7 7
16 1 0 7
2 2 8 10
66 -1 10 7
25 -2 9 8
0 4 6 1
24 3 4 7
55 -1 5 9
8 -1 10 8
9 0 3 10
8 15 56
1 16 74
1 20 96
6 17 27
7 14 94
1 15 27
3 8 56
1 14 87
13 20 22
13 19 23
1 16 89
6 14 82
9 20 1
12 20 8
7 19 28
2 13 89
2 15 80
1 14 94
6 19 25
12 19 31
```

Data Range

This problem is divided into subtasks. You earn the score for a subtask only if you pass **all** test cases in that subtask.

Subtask	Score	Constraints
1	5	$n, m \leq 3000$ (no special conditions)
2	20	$n, m \leq 10^5$, $y_i = 0$
3	20	$n, m \leq 2 \times 10^5$, $y_i = 0$
4	30	$n, m \leq 10^5$ (no special conditions)
5	25	$n, m \leq 2 \times 10^5$ (no special conditions)

For all test data, the following hold:

$$\begin{aligned}
&1 \leq n, m \leq 2 \times 10^5, \\
&0 \leq x_i, a_i, b_i, X \leq 10^9, \\
&|y_i| \leq 10^9, \\
&1 \leq l \leq r \leq n, \\
&l', r', X' \in [-2^{63}, 2^{63} - 1], \\
&\text{for any integer interval } [l, r] \subseteq [1, n], \left| \sum_{i=l}^r y_i \right| \leq 10^9.
\end{aligned}$$

Time Limit: 5 s

Memory Limit: 4 GiB