Problem C. Watson and Intervals

This contest is open for practice. You can try every problem as many times as you like, though we won't keep track of which problems you solve. Read the Quick-Start Guide to get started.

Small input 8 points	Solve C-small
Large input 17 points	Solve C-large

Problem

Sherlock and Watson have mastered the intricacies of the language C++ in their programming course, so they have moved on to algorithmic problems. In today's class, the tutor introduced the problem of merging one-dimensional intervals. **N** intervals are given, and the ith interval is defined by the inclusive endpoints [$\mathbf{L_i}$, $\mathbf{R_i}$], where $\mathbf{L_i} \le \mathbf{R_i}$.

The tutor defined the *covered area* of a set of intervals as the number of integers appearing in at least one of the intervals. (Formally, an integer p contributes to the covered area if there is some j such that $\mathbf{L_j} \le p \le \mathbf{R_j}$.)

Now, Watson always likes to challenge Sherlock. He has asked Sherlock to remove exactly one interval such that the covered area of the remaining intervals is minimized. Help Sherlock find this minimum possible covered area, after removing exactly one of the **N** intervals.

Input

Each test case consists of one line with eight integers **N**, $\mathbf{L_1}$, $\mathbf{R_1}$, **A**, **B**, $\mathbf{C_1}$, $\mathbf{C_2}$, and **M**. **N** is the number of intervals, and the other seven values are parameters that you should use to generate the other intervals, as follows:

First define $x_1 = \mathbf{L_1}$ and $y_1 = \mathbf{R_1}$. Then, use the recurrences below to generate x_i, y_i for i = 2 to **N**:

- $x_i = (\mathbf{A}*x_{i-1} + \mathbf{B}*y_{i-1} + \mathbf{C_1})$ modulo **M**.
- $y_i = (\mathbf{A}*y_{i-1} + \mathbf{B}*x_{i-1} + \mathbf{C_2})$ modulo **M**.

We define $\mathbf{L_i} = \min(x_i, y_i)$ and $\mathbf{R_i} = \max(x_i, y_i)$, for all i = 2 to **N**.

Output

For each test case, output one line containing `Case #x: y`, where x is the test case number (starting from 1) and y is the minimum possible covered area of all of the intervals remaining after removing exactly one interval.

Limits

$1 \le \mathbf{T} \le 50$.
$0 \le \mathbf{L_1} \le \mathbf{R_1} \le 10^9$.
$0 \le \mathbf{A} \le 10^9$.
$0 \le \mathbf{B} \le 10^9$.
$0 \le \mathbf{C_1} \le 10^9$.
$0 \le \mathbf{C_2} \le 10^9$.
$1 \le \mathbf{M} \le 10^9$.

Small dataset

$1 \le \mathbf{N} \le 1000$.

Large dataset

$1 \le \mathbf{N} \le 5 * 10^5 (500000)$.

Sample

Input	Output
3 1 1 1 1 1 1 1 1 3 2 5 1 2 3 4 10 4 3 4 3 3 8 10 10	Case #1: 0 Case #2: 4 Case #3: 9

In case 1, using the generation method, the set of intervals generated are: {[1, 1]}. Removing the only interval, the *covered area* is 0.

In case 2, using the generation method, the set of intervals generated are: {[2, 5], [3, 5], [4, 7]}. Removing the first, second or third interval would cause the covered area of remaining intervals to be 5, 6 and 4, respectively.

In case 3, using the generation method, the set of intervals generated are: {[3, 4], [1, 9], [0, 8], [2, 4]}. Removing the first, second, third or fourth interval would cause the covered area of remaining intervals to be 10, 9, 9 and 10, respectively.