Triangle Grid

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	1024 megabytes

Little L has a regular triangular grid, where each grid line is a segment connecting the corresponding n division points of adjacent edges. These grid lines divide the regular triangle into several congruent smaller regular triangles.

For each intersection point of the grid lines in the figure, we represent it with a coordinate (x, y). The x coordinate from top to bottom ranges from 1 to $n+1$. $x = 1$ represents the topmost point, while $x = n+1$ represents the bottom line. For the i-th row, the y coordinates of the grid points from left to right range from 1 to i.

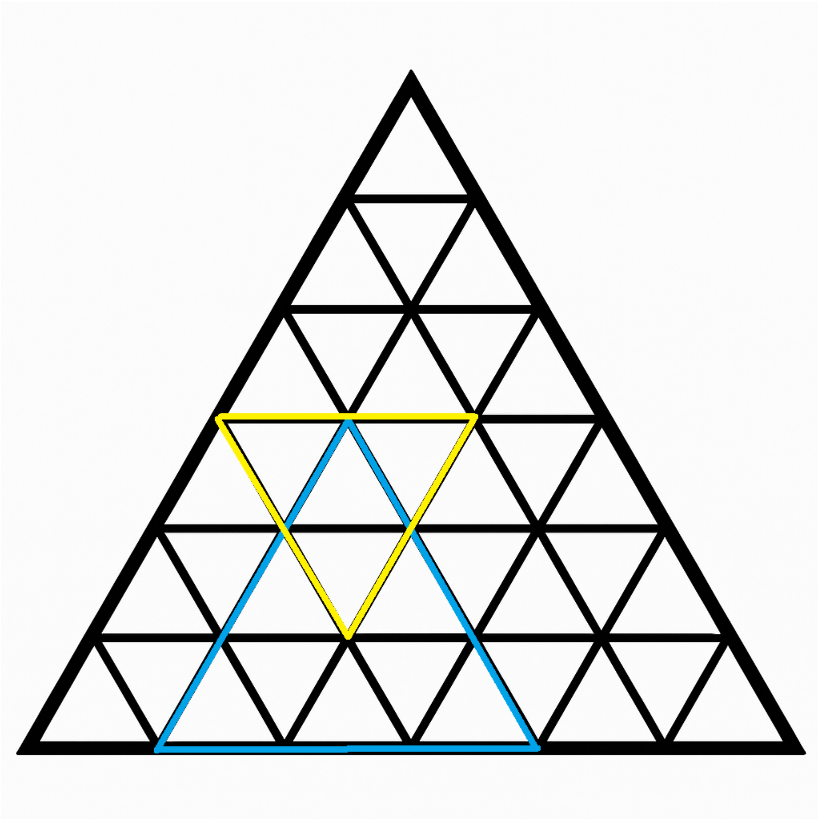
In the triangular grid, there are several regular triangles, which may be oriented upright or inverted. For each orientation of the triangle, we represent it with three numbers x, y, d. Here, (x, y) represents the coordinates of the diagonal vertices of its edge parallel to the grid's bottom line, and d represents its side length (the length of the segment).

For example, in the figure below, the blue regular triangle is an upright triangle, with $(x, y, d) = (4, 2, 3)$. The yellow regular triangle is an inverted triangle, with $(x, y, d) = (6, 3, 2)$.

Now, for each unit small regular triangle grid (a regular triangle with a side length of 1, regardless of orientation), there is a number written on it. Initially, all numbers on the grids are 0. Little L will perform the following operations:

- Choose a regular triangle and add w to all numbers on the grids within it.
- Choose a regular triangle and query the sum of all numbers on the grids within it.

He hopes you can compute the answer for each query operation. Since Little L does not like large numbers, you only need to output the result of the answer modulo 2^{32}.



Input

The first line of input contains two integers n, q ($2 \le n \le 10^5, 1 \le q \le 10^5$).

The next q lines each represent an operation.

Each line starts with five numbers $opt, type, x, y, d$ ($opt \in \{1, 2\}, type \in \{0, 1\}, 1 \le x \le n+1, 1 \le y \le x, 1 \le d \le n$). opt represents the type of operation, where $opt = 1$ is a modification operation, and $opt = 2$ is a query operation. $type$ indicates the type of triangle for this operation, where $type = 0$ is for an upright triangle, and $type = 1$ is for an inverted triangle. x, y, d describe this triangle. If $opt = 1$, there is an additional number w ($0 \le w < 2^{32}$) that indicates the number added during this modification operation.

It is guaranteed that all input triangles have a non-zero area and are valid triangles completely contained within the larger triangle.

Output

For each query with $opt = 2$, output a single number on a new line, representing the sum within the area modulo 2^{32}.

Examples

standard input	standard output
5 5 1 0 3 3 2 2 1 1 6 4 2 3 2 0 1 1 5 2 0 3 2 3 2 1 5 3 2	20 11 3
10 10 1 0 1 1 5 2 1 1 3 2 1 6 1 1 6 4 2 1 2 0 10 3 1 2 1 5 2 1 2 1 6 4 2 1 0 7 7 1 7 1 1 9 3 1 3 1 0 9 5 2 4 2 0 6 1 2	0 2 12 0