Problem E. Queuing

Problem Description

Little T loves queuing! Whether it's for enjoying a bowl of ramen, riding amusement park attractions, or exploring a comic convention, you can always find Little T in the midst of a queue. However, this time, Little T isn't queuing for leisure; he's preparing for the future of humanity.

After three days and nights of observation, Little T noticed that the people in line in front of the pharmacy are remarkably patient. They won't leave the queue until their turn comes. Little T has identified three possible scenarios:

1. The pharmacy provides each person in the queue with k masks.
2. The first x people at the front of the queue leave content.
3. y more people join the back of the queue.

Little T wants to know how many masks the person with the most masks in the queue has after each of these scenarios occurs. So, he has provided you with this data and hopes you can write a program to calculate the results.

Input Format

The first line contains a positive integer q, representing the number of observed scenarios by Little T.

The following q lines each start with a positive integer op_i, followed by some parameters indicating the upcoming scenario. Each scenario can be one of the following three:

- `1` k: Everyone in the line gets k masks.
- `2` x: x people in the queue leave content.
- `3` y: Another y people join the queue.

Output Format

For each scenario, please output an integer representing how many masks the person with the most masks in the queue has right now.

Constraints

- $1 \leq q \leq 500\,000$.
- $op_i \in \{1, 2, 3\}$ for $i = 1, 2, \ldots, q$.
- $1 \leq k, x, y \leq 10^6$ for each scenario.
- It is guaranteed that the number of people in the queue will never be negative at any moment, and if the queue is empty, there will be no mask distribution operations.
- All the inputs are integers.

Subtasks

1. (20 points) $q \leq 1000$; $x = y = 1$.
2. (50 points) $q \leq 1000$.
3. (30 points) No additional constraints.

No.	Testdata Range	Time Limit (ms)	Memory Limit (KiB)
Samples	1-3	1000	262144
1	4-12	1000	262144
2	1-18	1000	262144
3	1-26	1000	262144

Samples

Sample Input 1

```
3
3 5
1 4
1 9
```

This sample input satisfies the constraints of Subtasks 2, 3.

Sample Output 1

```
0
4
13
```

Sample Input 2

```
7
3 1
1 6
3 1
3 1
1 2
2 1
1 1
```

This sample input satisfies the constraints of Subtasks 1, 2, 3.

Sample Output 2

```
0
6
6
6
8
2
3
```

Operation	Queue
3 1	$[0]$
1 6	$[6]$
3 1	$[6, 0]$
3 1	$[6, 0, 0]$
1 2	$[8, 2, 2]$
2 1	$[2, 2]$
1 1	$[3, 3]$

Sample Input 3

```
8
3 1000000
1 1000000
2 999999
3 1000000
1 1000000
2 999999
3 1000000
1 1000000
```

This sample input satisfies the constraints of Subtasks 2, 3.

Sample Output 3

```
0
1000000
1000000
1000000
2000000
1000000
1000000
2000000
```