

In one distant world, in the glorious city of Erbovle, a new bank was opened. The bank has m employees working with clients, and one chief accountant.

To solve their problems, gnomes come to the bank. It is known that the i -th gnome comes to the bank through t_i minutes after opening the bank. First he needs to hold a_i minutes at one of m employees, and then b_i minutes in the office of the chief accountant.

Of course, several gnomes cannot be at the same time at the same employee or in the office of the chief accountant, so queues are formed for the employees and the chief accountant.

The queue for employees is common, while the gnome from the queue goes to the first vacant employee. If two gnomes come to the bank at the same time, then the first one in the queue is the one whose number is less. If the gnome began to be serviced by an employee at time x , then it is released at time $x + a_i$, at this moment another gnome can start to be serviced by the same employee. A dwarf who comes to the bank at the time of t can start servicing with an employee at any time, starting with t .

Having solved his problems with an employee, the dwarf goes in queue to the chief accountant. Similarly, if two gnomes come to this queue at the same time, the gnome with the lower number gets up first, at the moment when one of the gnomes finishes servicing, the next one can start servicing immediately, the gnome can get to the chief accountant, starting from the moment when he finished servicing employee.

Today, n gnomes are going to come to the bank, everyone knows about what time he goes to the bank, how much time he wants to spend at the window and how much time he wants to spend at the accountant. You need to tell the exit time from the bank for each gnome.

The first line contains two integers n and m ($1 \leq n \leq 100\,000$; $1 \leq m \leq 10$) – the number of gnomes and employees, respectively. Next, n lines contain three integers t_i , a_i and b_i ($1 \leq t_i, a_i, b_i \leq 10^9$) – the arrival time of the i -th gnome, how many minutes the i -th gnome should spend with a bank employee and how many minutes he should spend in the office of the chief accountant. It is known that gnomes are given in the order of arrival at the bank, that is, for any pair $i < j$, $t_i \leq t_j$.

Print n integers, the i -th number should be equal to the number of minutes after opening, when the i -th gnome leaves the bank.

Sample input:

```
4 2
1 3 3
1 2 2
2 2 1
2 1 4
```

Sample output:

```
8
5
9
13
```