

Problem H. Trading

Input file: **trading.in**
Output file: **trading.out**
Time limit: 2 seconds
Memory limit: 64 megabytes

There are N small villages close to the highway between Almaty and Taraz numbered from 1 to N . At the beginning of the winter M unknown traders began trading knitted hats in these villages. They have only two rules: never trade in one place more than once (one day) and increase the price on hats each day. More formally, each i -th trader:

1. begins trading in village L_i with starting price X_i .
2. each day he moves to the next adjacent village, i.e. if he was trading in village j yesterday, then today he is trading in village $j + 1$.
3. each day he increases the price by 1, so if yesterday's price was x , then today's price is $x + 1$.
4. stops trading at village R_i (after he traded his knitted hats in village R_i).

The problem is for each village to determine the maximal price that was there during the whole trading history.

Input

Each line contains two integer number N ($1 \leq N \leq 300000$) and M ($1 \leq M \leq 300000$) — number of villages and traders accordingly.

Next M lines contains 3 numbers each: L_i, R_i ($1 \leq L_i \leq R_i \leq N$) and X_i ($1 \leq X_i \leq 10^9$) — numbers of first and last village and starting price for i -th trader.

Output

Output N integer numbers separating them with spaces — i -th number being the maximal price for the trading history of i -th village. If there was no trading in some village, output 0 for it.

Examples

trading.in	trading.out
5 2 1 3 2 2 4 6	2 6 7 8 0
6 4 4 4 3 1 2 5 5 6 1 6 6 1	5 6 0 3 1 2