

Зима в Лилипутии

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Лилипутия представляет собой прямоугольную таблицу $n \times m$. В этом году в Лилипутии выдалась суровая зима: на город падают массивные снежинки.

Синоптики собрали срочное совещание и выяснили, в какие точки будут падать снежинки. Каждая снежинка будет покрывать некоторые клетки таблицы: центральную клетку с лучами длины k клеток в 8 сторон: в обе стороны по горизонтали, в обе стороны по вертикали и в четыре стороны по диагоналям.

У центральной клетки снежинки толщина k . По мере удаления от центральной клетки на каждом из 8 лучей толщина уменьшается на один с каждой следующей клеткой: $k, k - 1, \dots, 1$. Например, если $k = 3$:

```
0000000
0101010
0022200
0123210
0022200
0101010
0000000
```

Помогите лилипутам узнать для каждой клетки города, какой суммарной толщиной снега она будет покрыта.

Формат входных данных

В первой строке вводятся два целых числа n, m ($1 \leq n \cdot m \leq 10^6$) — размеры города.

Во второй строке вводится единственное целое число q ($1 \leq q \leq 2 \cdot 10^5$) — количество снежинок, которые упадут на Лилипутью.

В последующих q строках вводятся описания снежинок: на i -й строке три целых числа a_i, b_i, k_i ($1 \leq a_i \leq n, 1 \leq b_i \leq m, 1 \leq k_i \leq 10^6$) — координаты центральных клеток снежинок, а также длины лучей.

Обратите внимание, что луч снежинки может выходить за пределы города.

Формат выходных данных

Выведите n строк по m чисел: в i -й строке на j -й позиции суммарная толщина частей снежинок, которые упали на эту клетку.

Система оценки

Решения, верно работающие при $n \cdot m \cdot q \leq 1e8$, будут получать не менее 24 баллов. Решения, верно работающие при $\max(n, m) \cdot q \leq 1e8$, будут получать не менее 48 баллов.

Примеры

стандартный ввод	стандартный вывод
7 7 1 4 4 3	0 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0 2 2 2 0 0 0 1 2 3 2 1 0 0 0 2 2 2 0 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0
7 7 2 4 4 3 2 3 10	0 9 9 9 0 0 0 8 10 10 10 8 8 6 0 9 11 11 2 0 0 8 1 10 3 10 1 0 0 0 9 2 2 7 0 0 1 6 1 0 1 6 0 0 5 0 0 0 0