

В. Зима в Лилипутии

✓ Полное решение

Ограничение времени	1 секунда
Ограничение памяти	512 Мб
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

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 ввода

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 Лилипутию.

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 вывода

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

Система оценивания

Решения, верно работающие при $n \cdot m \cdot q \leq 10^8$, будут получать не менее 24 баллов.

Решения, верно работающие при $\max(n, m) \cdot q \leq 10^8$, будут получать не менее 48 баллов.

Пример 1

Ввод	Вывод
7 7	0 0 0 0 0 0 0
1	0 1 0 1 0 1 0
4 4 3	0 0 2 2 2 0 0

Ввод

Вывод

```
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
```

Пример 2

Ввод

Вывод

```
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
```

Ответ

Язык

Python 3.13.2



Набрать здесь

Отправить файл

```
1 import sys
2
3 def solve() -> None:
4     data = sys.stdin.read().split()
5     if not data:
6         return
7     it = iter(data)
8     n = int(next(it))
9     m = int(next(it))
10    q = int(next(it))
11
12    ans = [[0] * m for _ in range(n)]
13
14    for _ in range(q):
15        a = int(next(it)) - 1      # row (0-based)
16        b = int(next(it)) - 1      # column (0-based)
17        k = int(next(it))
18
19        # cells that can be affected lie inside a square of side 2·(k-1)+1
20        r0 = max(0, a - (k - 1))
21        r1 = min(n - 1, a + (k - 1))
22        c0 = max(0, b - (k - 1))
23        c1 = min(m - 1, b + (k - 1))
24
25        for i in range(r0, r1 + 1):
```

Отправить

Осталось 100 попыток

< Предыдущая

Следующая >

[Соглашение](#) · [Документация](#) · [Поддержка](#)