

## Problem A. New Home 新家

Time limit: 5 seconds  
Memory limit: 1024 megabytes

五福街是一条笔直的道路，这条道路可以看成是一个数轴，街上每个建筑物的坐标都可以用一个整数来表示。小明是一位时光旅行者，他知道在这条街上，在过去现在和未来共有  $n$  个商店出现。第  $i$  个商店可以使用四个整数  $x_i, t_i, a_i, b_i$  描述，它们分别表示：商店的坐标、商店的类型、商店开业的年份、商店关闭的年份。

小明希望通过时光旅行，选择一个合适的时间，住在五福街上的某个地方。他给出了一份他可能选择的列表，上面包括了  $q$  个询问，每个询问用二元组（坐标，时间）表示。第  $i$  对二元组用两个整数  $l_i, y_i$  描述，分别表示选择的地点  $l_i$  和年份  $y_i$ 。

现在，他想计算出在这些时间和地点居住的生活质量。他定义居住的不方便指数为：在居住的年份，离居住点最远的商店类型到居住点的距离。类型  $t$  的商店到居住点的距离定义为：在指定的年份，类型  $t$  的所有营业的商店中，到居住点距离最近的一家到居住点的距离。我们说编号为  $i$  的商店在第  $y$  年在营业当且仅当  $a_i \leq y \leq b_i$ 。注意，在某些年份中，可能在五福街上并非所有  $k$  种类型的商店都有至少一家在营业。在这种情况下，不方便指数定义为  $-1$ 。

你的任务是帮助小明求出每对（坐标，时间）二元组居住的不方便指数。

### Input

第一行包含三个整数  $n$ ， $k$  和  $q$ ，分别表示商店的数量、商店类型的数量和（坐标，时间）二元组的数量。（ $1 \leq n, q \leq 3 \cdot 10^5, 1 \leq k \leq n$ ）。

接下来  $n$  行，每行包含四个整数  $x_i, t_i, a_i$ ，和  $b_i$  用于描述一家商店，意义如题面所述（ $1 \leq x_i, a_i, b_i \leq 10^8, 1 \leq t_i \leq k, a_i \leq b_i$ ）。

接下来  $q$  行，每行包含两个整数  $l_i$ ，和  $y_i$ ，表示一组（坐标，时间）查询（ $1 \leq l_i, y_i \leq 10^8$ ）。

### Output

输出一行，包含  $q$  个整数，依次表示对于  $q$  组（坐标，时间）询问求出的结果。

### Scoring

#### Subtask 1 (points: 5)

$n, q \leq 400$

#### Subtask 2 (points: 7)

$n, q \leq 6 \cdot 10^4, k \leq 400$

#### Subtask 3 (points: 10)

$n, q \leq 3 \cdot 10^5$ ，对于所有的商店  $a_i = 1, b_i = 10^8$ 。

#### Subtask 4 (points: 23)

$n, q \leq 3 \cdot 10^5$ ，对于所有的商店  $a_i = 1$ 。

#### Subtask 5 (points: 35)

$n, q \leq 6 \cdot 10^4$

#### Subtask 6 (points: 20)

$n, q \leq 3 \cdot 10^5$

## Examples

input	output
4 2 4 3 1 1 10 9 2 2 4 7 2 5 7 4 1 8 10 5 3 5 6 5 9 1 10	4 2 -1 -1
2 1 3 1 1 1 4 1 1 2 6 1 3 1 5 1 7	0 0 -1
1 1 1 100000000 1 1 1 1 1	99999999

## Note

在第一个样例中，有 4 家商店，共 2 种类型，还有 4 个询问。

- 对于第一个询问：小明在第 3 年住在坐标为 5 的地方。这一年中，编号为 1 和 2 的商店在营业，到编号为 1 的商店的距离为 2，到编号为 2 的商店距离为 4，所以最大距离为 4。
- 对于第二个询问：小明在第 6 年住在坐标为 5 的地方。这一年中，编号为 1 和 3 的商店在营业，到编号为 1 的商店的距离为 2，到编号为 3 的商店距离为 2，所以最大距离为 2。
- 对于第三个询问：小明在第 9 年住在坐标为 5 的地方。这一年中，编号为 1 和 4 的商店在营业，它们的类型都为 1，没有类型为 2 的商店在营业，所以答案为 -1。
- 同样的情况出现在第四个询问中。

在第二个样例中，有 2 家商店，共 1 种类型，还有三个询问。两家商店的类型都是 1。在所有的询问中，小明均住在坐标为 1 的地方。在前两个询问中，至少有一个商店在营业，所以答案为 0，在第三个询问中，两个商店都不在营业，所以答案为 -1。

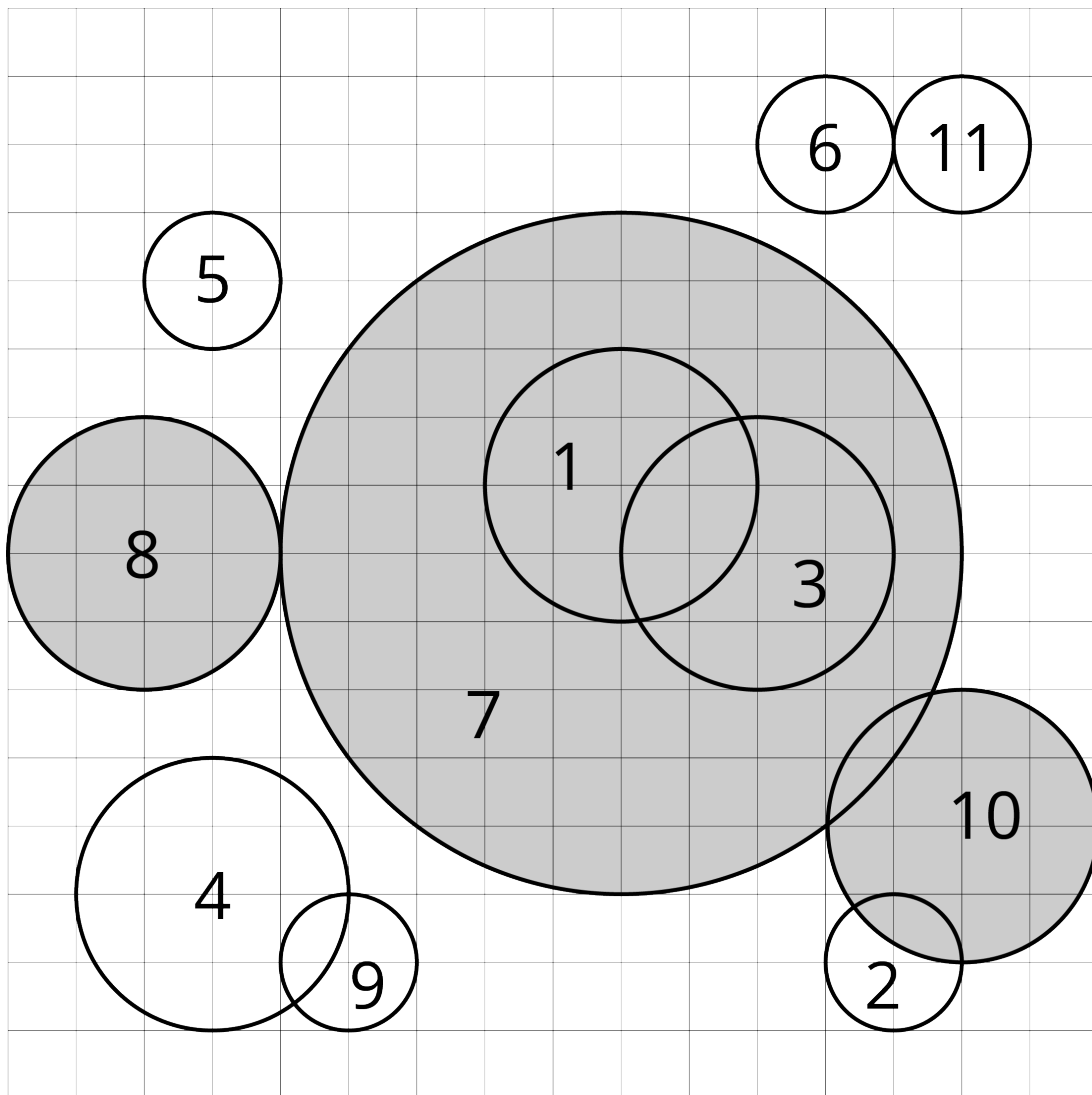
在第三个样例中，有 1 家商店和 1 个询问，两者之间的距离是 99999999。

## Problem B. Circle selection 选圆圈

Time limit: 3 seconds  
Memory limit: 1024 megabytes

在平面上，有  $n$  个圆，记为  $c_1, c_2, \dots, c_n$ 。我们尝试对这些圆运行这个算法：

1. 找到这些圆中半径最大的。如果有多个半径最大的圆，选择编号最小的。记为  $c_i$ 。
2. 删除  $c_i$  及与其有交集的所有圆。两个圆有交集当且仅当平面上存在一个点，这个点同时在这两个圆的圆周上或圆内。（原文直译：如果平面上存在一个点被这两个圆所包含，我们称这两个圆有交集。一个点被一个圆包含，当且仅当它位于圆内或圆周上。）
3. 重复上面两个步骤直到所有的圆都被删除。



当  $c_i$  被删除时，若循环中第1步选择的圆是  $c_j$ ，我们说  $c_i$  被  $c_j$  删除。对于每个圆，求出它是被哪一个圆删除的。

### Input

第一行包含一个整数  $n$ ，表示开始时平面上圆的数量 ( $1 \leq n \leq 3 \cdot 10^5$ )。

接下来  $n$  行，每行包含三个整数  $x_i, y_i, r_i$  依次描述圆  $c_i$  圆心的x坐标、y坐标和它的半径 ( $-10^9 \leq x_i, y_i \leq 10^9, 1 \leq r_i \leq 10^9$ )。

## Output

输出一行，包含  $n$  个整数  $a_1, a_2, \dots, a_n$ ，其中  $a_i$  表示圆  $c_i$  是被圆  $c_{a_i}$  删除的。

## Scoring

### Subtask 1 (points: 7)

$n \leq 5000$

### Subtask 2 (points: 12)

$n \leq 3 \cdot 10^5$ , 对于所有的圆  $y_i = 0$

### Subtask 3 (points: 15)

$n \leq 3 \cdot 10^5$ , 每个圆最多和一个其他圆有交集。

### Subtask 4 (points: 23)

$n \leq 3 \cdot 10^5$ , 所有的圆半径相同。

### Subtask 5 (points: 30)

$n \leq 10^5$

### Subtask 6 (points: 13)

$n \leq 3 \cdot 10^5$

## Example

input	output
11 9 9 2 13 2 1 11 8 2 3 3 2 3 12 1 12 14 1 9 8 5 2 8 2 5 2 1 14 4 2 14 14 1	7 2 7 4 5 6 7 7 4 7 6

## Note

题目描述中的图片对应了样例一中的情形。

## Problem C. Duathlon 铁人两项

Time limit: 1 second  
Memory limit: 1024 megabytes

比特镇的路网由  $m$  条双向道路连接的  $n$  个交叉路口组成。

最近，比特镇获得了一场铁人两项锦标赛的主办权。这场比赛共有两段赛程：选手先完成一段长跑赛程，然后骑自行车完成第二段赛程。

比赛的路线要按照如下方法规划：

1、先选择三个两两互不相同的路口  $s$ ， $c$  和  $f$ ，分别作为比赛的起点、切换点（运动员在长跑到达这个点后，骑自行车前往终点）、终点。

2、选择一条从  $s$  出发，经过  $c$  最终到达  $f$  的路径。考虑到安全因素，选择的路径经过同一个点至多一次。

在规划路径之前，镇长想请你帮忙计算，总共有多少种不同的选取  $s$ ， $c$  和  $f$  的方案，使得在第 2 步中至少能设计出一条满足要求的路径。

### Input

第一行包含两个整数  $n$  和  $m$ ，分别表示交叉路口和双向道路的数量。

接下来  $m$  行，每行两个整数  $v_i$ ， $u_i$ 。表示存在一条双向道路连接交叉路口  $v_i$ ， $u_i$  ( $1 \leq v_i, u_i \leq n$ ,  $v_i \neq u_i$ )。

保证任意两个交叉路口之间，至多被一条双向道路直接连接。

### Output

输出一行，包括一个整数，表示能满足要求的不同的选取  $s$ ， $c$  和  $f$  的方案数。

### Scoring

#### Subtask 1 (points: 5)

$n \leq 10$ ,  $m \leq 100$

#### Subtask 2 (points: 11)

$n \leq 50$ ,  $m \leq 100$

#### Subtask 3 (points: 8)

$n \leq 100\,000$ , 每个交叉路口至多作为两条双向道路的端点。

#### Subtask 4 (points: 10)

$n \leq 1\,000$ , 在路网中不存在环。

存在环是指存在一个长度为  $k$  ( $k \geq 3$ ) 的交叉路口序列  $v_1, v_2, \dots, v_k$ ，序列中的路口编号两两不同，且对于  $i$  从 1 到  $k-1$ ，有一条双向道路直接连接路口  $v_i$  和  $v_{i+1}$ ，且有一条双向道路直接连接路口  $v_k$  和  $v_1$ 。

#### Subtask 5 (points: 13)

$n \leq 100\,000$ , 在路网中不存在环。

#### Subtask 6 (points: 15)

$n \leq 1\,000$ , 对于每个交叉路口，至多被一个环包含。

### Subtask 7 (points: 20)

$n \leq 100\,000$ , 对于每个交叉路口, 至多被一个环包含。

### Subtask 8 (points: 8)

$n \leq 1\,000$ ,  $m \leq 2\,000$

### Subtask 9 (points: 10)

$n \leq 100\,000$ ,  $m \leq 200\,000$

### Examples

input	output
4 3 1 2 2 3 3 4	8
4 4 1 2 2 3 3 4 4 2	14

### Note

在第一个样例中, 有以下 8 种不同的选择  $(s, c, f)$  的方案:  $(1, 2, 3)$ ,  $(1, 2, 4)$ ,  $(1, 3, 4)$ ,  $(2, 3, 4)$ ,  $(3, 2, 1)$ ,  $(4, 2, 1)$ ,  $(4, 3, 1)$ ,  $(4, 3, 2)$ 。

在第二个样例中, 有以下 14 种不同的选择  $(s, c, f)$  的方案:  $(1, 2, 3)$ ,  $(1, 2, 4)$ ,  $(1, 3, 4)$ ,  $(1, 4, 3)$ ,  $(2, 3, 4)$ ,  $(2, 4, 3)$ ,  $(3, 2, 1)$ ,  $(3, 2, 4)$ ,  $(3, 4, 1)$ ,  $(3, 4, 2)$ ,  $(4, 2, 1)$ ,  $(4, 2, 3)$ ,  $(4, 3, 1)$ ,  $(4, 3, 2)$ 。