

BIGTREE: Big Search Trees

题目描述

大厨的花园里有各种各样的树。有黑柿子树，有谈像素，当然还有二叉搜索树。

大厨的**二叉搜索树**是以一种独特的方式生长的。二叉搜索树会通过向其中插入节点的方式来生长。第一个被插入的节点叫做**树根**。与大厨的其他树不一样，二叉树是向下生长的，树根其实是树的最顶端。

每个节点都是由**边**与最多两个在其下方的节点相连的，这两个节点分别被称为左儿子和右儿子。这些节点可以为“空”，也即不存在这样的节点。最后，每个节点上都有一个数字，为其**标记**。

要让一棵二叉搜索树生长，你必须向其中插入新的节点。插入的规则由下面的伪代码给出：

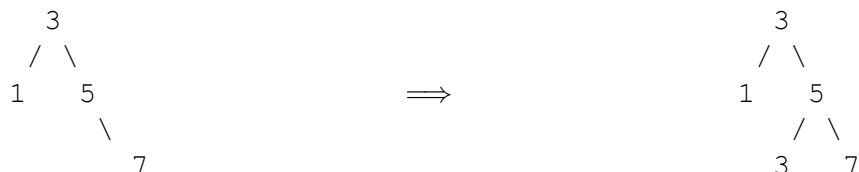
```
def insert(x, y):
    if x == null:
        return y

    if y.label < x.label:
        x.left = insert(x.left, y)
    else:
        x.right = insert(x.right, y)

    return x
```

要在树根为 x 的树种插入一个新节点 y ，调用 `insert(x, y)` 即可。返回的值为新树的树根。

下图左是一棵二叉树，其根的标记为 3。在插入一个标记同样为 3 的新节点后，其形态变为了下图右中的样子。



一个节点的**深度**是从其到根的路径上的边数。比如，上图中标记为 7 的节点的深度为 2，而标记为 1 的节点的深度为 1。

大厨的花园中一共有 T 棵二叉搜索树。每棵树用四个数字 a 、 b 、 m 和 N 描述，意味着这棵树一共有 N 个节点，第 k 个插入的节点的标记为 $(a + bk) \pmod{m}$ 。

请帮大厨算出每棵二叉搜索树中最后一个插入的节点的深度。

输入格式

输入的第一行包含一个整数 T ，代表花园中二叉搜索树的棵数。

接下来 T 行，每行包含四个整数 a 、 b 、 m 和 N ，描述一棵树。

输出格式

对于每个询问，输出一行，包含一个整数，代表树中最后一个插入的节点的深度。

数据范围

- $1 \leq T \leq 5 \cdot 10^4$
- $1 \leq N \leq 10^{16}$
- $0 \leq a, b < m \leq 10^8$

样例数据

输入

```
5
1 2 8 1
1 2 8 2
1 2 8 3
1 2 8 4
1 2 8 5
```

输出

```
0
1
2
1
2
```

样例解释

样例即为题目描述中的例子。

时间限制

2 秒