五指山

description

1 西游记中孙吾空大闹天宫，如来佛祖前来降伏他，说道：“我与你打个赌赛；你若有本事，一筋斗打出我这右手掌中，算你赢，再不用动刀兵苦争战，就请玉帝到西方居住，把天宫让你；若不能打出手掌，你还下界为妖，再修几劫，却来争吵。”

那大圣闻言，暗笑道：“这如来十分好呆！我老孙一筋斗去十万八千里。他那手掌，方圆不满一尺，如何跳不出去？”急发声道：“既如此说，你可做得主张？”佛祖道：“做得！做得！”伸开右手，却似个荷叶大小。那大圣收了如意棒，抖擞神威，将身一纵，站在佛祖手心里，却道声：“我出去也！”你看他一路云光，无影无形去了。大圣行时，忽见有五根肉红柱子，撑着一股青气。他道：“此间乃尽头路了。这番回去，如来作证，灵霄殿定是我坐也。”翻转筋斗云，径回本处，站在如来掌：“我已去，今来了。你教玉帝让天宫与我。”

如来骂道：“你正好不曾离了我掌哩！”大圣道：“你是不知。我去到天尽头，见五根肉红柱，撑着一股青气，我留个记在那里，你敢和我同去看么？”如来道：“不消去，你只自低头看看。”那大圣睁圆火眼金睛，低头看时，原来佛祖右手中指写着“齐天大圣，到此一游。”大圣大吃了一惊道：“有这等事！有这等事！我将此字写在撑天柱子上，如何却在他手指上？莫非有个未卜先知的法术？我决不信！不信！等我再去来！”

好大圣，急纵身又要跳出，被佛祖翻掌一扑，把这猴王推出西天门外，将五指化作金、木、水、火、土五座联山，唤名“五行山”，轻轻的把他压住。

　　我们假设佛祖的手掌是一个圆圈（所以任凭大圣一个筋斗云十万八千里也是飞不出其手掌心），圆圈的长为 n，逆时针记为：0，1，2，…，n-1，而大圣每次飞的距离为 d. 现在大圣所在的位置记为 x，而大圣想去的地方在 y。现在要你告诉大圣至少要多少筋斗云才能到达目的地。

input

有多组测试数据。

第一行是一个正整数 T，表示测试数据的组数。

每组测试数据包括一行，四个非负整数，n（$2 < n < 10^9$），表示如来手掌圆圈的长度；d($0 < d < n$)，筋斗所能飞的距离；x($0 <= x < n$)，大圣的初始位置；y($0 <= y < n$)，大圣想去的地方。

　注意孙悟空的筋斗云只沿着逆时针方向翻。

output

对于每组测试数据，输出一行，给出大圣最少要翻多少个筋斗云才能到达目的地。

如果无论翻多少个筋斗云也不能到达，输出“Impossible”.

sample_input

2
3 2 0 2
3 2 0 1

sample_output

1
2

hint

source

拓展欧几里得

设大圣翻的筋斗云的个数为 res, 起始点和目标点的距离为 x， 则

本题即为求同余方程 $res * d \equiv x \pmod n$ 中 res 的最小非负整数解。。。

下面详讲如何解同余方程 $ax \equiv c \pmod b$

根据同余的定义有 $ax \equiv c \pmod b$ 相当于 $ax + by = c$

所以， 求同余方程的整数解可转换为求不定方程的整数解

显然,不定方程 $ax + by = gcd(a, b)$有整数解（具体证明不给出）

对于不定方程 $ax + by = c$, 当且仅当 $gcd(a, b) \mid c$ 时，

方程有整数解（可以思考一下为什么）

当 $gcd(a, b) \mid c$ 时， 设 $g = gcd(a, b), a' = a / g, b' = b / g, c' = c / g$

我们可以用 extend_gcd(a’ , b’)求出不定方程 $a' x' + b' y' = 1$ 的整数解

那么 $a' c' x' + b' c' y' = c'$

$a' g c' x' + b' g c' y' = c' g$

即 $a c' x' + b c' y' = c$

所以 $x = c' x' , y = c' y'$

注意：此时求出的 x , y 只是一组解

该不定方程的通解为 $x = x_0 + b / gcd(a, b) * t, y = y_0 - a / gcd(a, b) * t$

所以, x 是模 b’ 同余的一个剩余类

让我们回到 $ax \equiv c \pmod b$,

x 的最小非负整数解 即为 不定方程 $ax + by = c$ 的某一个特殊解 模 x 的结果

下面讲一下 extend_gcd(int, int)

我们在求两数 a, b 的最大公因数的同时, 求出 $ax + by = gcd(a, b)$的一组特殊解

显然， 当 $b = 0$ 时， $gcd(a, b) = a$

所以， $ax + by = gcd(a, b)$ 有一组解为 $x = 1, y = 0$

当 b != 0 时， 我们递归求 gcd(b, a % b)

现在我们已经求出 bx’ + a % b y’ = gcd(b, a%b) = gcd(a, b)的解

那么 bx’ + (a - a / b * b) y’ = gcd(a, b) （此处 / 为整除）

所以 ay’ + b (x’ - a / b y’) = gcd(a, b)

所以 x = y’ , y = x’ - a / b y’

```
int extend_gcd(const int &a, const int &b, int &x, int &y) {
    if (!b) {
        x = 1, y = 0;
        return a;
    }
    else {
        int tmp;
        extend_gcd(b, a % b, tmp, x);
        y = tmp - x * (a / b);
    }
}
```