CodeForces Round 1037 Div. 2E / Div. 1C

omeganot

July 19, 2025

The airplane getting reflected when it hits a boundary can instead be thought of as continuing with the same trajectory but reflecting the shape instead. Consider the grid below.

Rather than being reflected when it hits a boundary, we can think of the airplane as traveling on an infinite grid as shown above, as in both cases the plane will be in the same spot relative to its respective triangle. This is because crossing any edge is just entering the same shape but flipped. As such, if a plane hits one of the vertices of the triangle, it will hit a lattice point of this special grid.

Thus, we would like to know how many boundaries the airplane crosses before it hits a lattice point on the grid given its initial position in the bottom left triangle and the velocity vector, which now doesn't change as a result of our transformation. Intuitively, we probably want to know at which point on the grid does the plane exit the triangle. Thus, we are looking for x_f, y_f such that

$$x_f = v_x t + x$$

$$y_f = v_y t + y$$

and x_f and y_f are integers divisible by n, while minimizing t. Intuitively, if v_x and v_y are coprime, then t must be an integer if it exists. If v_x and v_y share

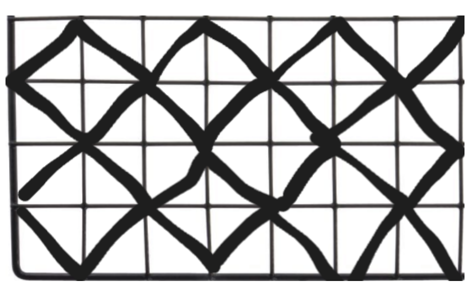

Figure 1: Infinite grid

a common divisor, then t might not be an integer. Thus, we can divide v_x and v_y by $\gcd(v_x, v_y)$, and from now on, we will treat these as the new values of v_x and v_y, as we still maintain the same velocity vector. The benefit is that now we can reformulate the previous equations into modular arithmetic:

$$v_x t + x \equiv 0 \pmod{n}$$

$$v_y t + y \equiv 0 \pmod{n}$$

If we try to solve these equivalences, we would get:

$$t \equiv -x \cdot v_x^{-1} \pmod{n}$$

$$t \equiv -y \cdot v_y^{-1} \pmod{n}$$

The issue with solving this directly is that the modular inverse of our velocity vector might not exist modulo n. This occurs when $\gcd(v, n) \neq 1$. If $\gcd(v, n) \neq 1$ and $x \bmod \gcd(v, n) \neq 0$, then trivially a solution doesn't exist. Thus, x must be divisible by $\gcd(v, n)$, and we can reformulate our equivalence statements into

$$t \equiv \frac{-x}{\gcd(v_x, n)} \cdot \left(\frac{v}{\gcd(v_x, n)}\right)^{-1} \quad (\mathrm{mod}\ \frac{n}{\gcd(v_x, n)})$$

$$t \equiv \frac{-y}{\gcd(v_y, n)} \cdot \left(\frac{v}{\gcd(v_y, n)}\right)^{-1} \quad (\mathrm{mod}\ \frac{n}{\gcd(v_y, n)})$$

The issue now is that our modulo might not be the same, so we will have to use the Chinese Remainder Theorem. My inexperience with modular arithmetic led me to being unable to construct a solution purely using the Extended Euclidean algorithm. However, it would be much easier as there are only two statements. Regardless, since the Chinese Remainder Theorem finds exactly one solution modulo some m, we know that will be our minimum t.

Using a modular inverse algorithm that uses the Extended Euclidean algorithm we can derive two modular equivalence statements that our Chinese Remainder Theorem needs to satisfy. Since our modulo are not necessarily coprime, I obtain the prime factors of $\gcd\left(\frac{n}{\gcd(v_x,n)}, \frac{n}{\gcd(v_y,n)}\right)$, and for each prime factor, I remove it from the congruence statement whose modulo has the lower power of that prime. For example, if the modulo are 6 and 4, and my prime factor is 2, I remove 2 from the statement modulo 6 and turn it into a statement modulo 3. Of course, we must first ensure that both statements are valid modulo the GCD. In other words, if we have statements like $t \equiv 1 \pmod{4}$ and $t \equiv 4 \pmod{6}$ then there is no solution. This is because $t \bmod \gcd(4, 6)$

is not consistent.

This makes our solution $O(\sqrt{n}+\log v_x+\log v_y)$ rather than $O(\log n+\log v_x+\log v_y)$ like the more elegant solutions using only the Extended Euclidean algorithm. After we find t using the Chinese Remainder Theorem, we can easily find the lattice point on the grid where the plane exits the triangle by multiplying by t by v and adding our initial position. Since n doesn't matter from now on, we can divide all latice points on the grid by n. In other words, $n = 6$ with an exit point of $(6, 12)$ is the same as $n = 1$ with an exit point of $(1, 2)$. Let our calculated exit point be (x_f, y_f). Now we must figure out how many grid lines our plane crosses.

We can think of the plane's path as a function $f(t)$ which is continuous on the interval (x, x_f). Furthermore, it is an increasing function with range (y, y_f). More intuitively, we just want to draw the line that our plane travels and see which grid lines we intersect. We can separate the grid lines into vertical, horizontal, main diagonal, and antidiagonal. The vertical lines are easiest: we will intersect the vertical line $x_f - 1$ times. Similarly, for the horizontal lines we intersect them $y_f - 1$ times.

The diagonal lines are trickier. For the main diagonal, we can observe that the set of main diagonal grid lines consists of all lines of the form $y = x + b$ where b is an odd integer. We can rewrite this as $y - x = b$. We can define this parametrically as $y(t) - x(t) = b$ if we let $x(t)$ and $y(t)$ be the respective parametric equations for our airplane. We can then derive:

$$y(t) - x(t) = b$$

$$y + v_y t - x - v_x t = b$$

$$y - x + (v_y - v_x)t = b$$

We would like to know how many odd integers b satisfy this on our curve. Since $x, y < 1$ (remember that we shrunk the grid where each cell is $n \times n$ to 1×1), we can eventually deduce that there are $\lfloor \frac{y_f - x_f}{2} \rfloor$ valid b.

Similarly, for antidiagonal lines, they all come in the form $y = b - x$ where b is an odd integer, which can then be turned into $y + x = b$. If we parameterize again we may derive

$$y + x + (v_y + v_x)t = b$$

This gets us that there are $\lfloor \frac{y_f + x_f}{2} \rfloor$ antidiagonal lines that we cross. I didn't prove this any more rigorously, but perhaps it is easy to convince yourself by inspecting some example cases.

Thus, we find our answer as $x_f + y_f + \lfloor \frac{y_f - x_f}{2} \rfloor + \lfloor \frac{y_f + x_f}{2} \rfloor - 2$. We solve the problem in $O(\sqrt{n} + \log v_x + \log v_y)$.