Mondrialayag

Problem Solving Technique: Solve an easier version first

It can be easier to reckon with, and might give you insight to the full version of the problem.

Always do the subtasks first. Even then, don't feel limited to needing subtasks—make up your own!

Subtask 1 asks us to output any Mondrialayagic Squares. But all magic squares are also Mondrialayagic squares. Get any magic squares from e.g. Wikipedia and hard code them into your program.

Standard Toolbox: Solution families

In math, a common technique is to ask:

- If I have *one* solution to a problem, can I use that to generate *more* solutions?

If you do, then starting from one "primitive" solution, you can produce an entire *family* of other solutions.

In the Wikipedia article on magic squares, under "Transformations that preserve the magic property", you would see the following properties:

- If you have a magic square, then adding a positive integer b to all entries yields another magic square.
- If you have a magic square, then multiplying a positive integer a to all entries yields another magic square.

To solve subtask 2: Take your hard-coded magic square. Suppose that the value of y is pre-filled, but in your primitive solution, there is x on that spot instead. Then, just add $y - x$ to all cells. Do this modulo 101, and you will still have a valid Mondrialayagic square.

Remember to replace 0 with 101 or you will get a Wrong Answer verdict!

To solve subtask 3: This time, suppose the values y_1 and y_2 are pre-filled, but your primitive solution has x_1 and x_2 in those spots instead. Then, we use our second degree of freedom. We want to solve this simultaneous system of equations:

$$\begin{cases} ax_1 + b = y_1 \\ ax_2 + b = y_2 \end{cases}$$

and by familiar HS algebra, this has the unique solution:

$$\begin{aligned} a &= \frac{1}{x_1 - x_2} \left(y_1 - y_2 \right), \\ b &= \frac{1}{x_1 - x_2} \left(x_1 y_2 - x_2 y_1 \right). \end{aligned}$$

(If you know a little bit of matrix stuff, now would be a good time to use it.)

To generate the Mondrialayagic square, we compute the multiplicative inverses modulo the prime 101, which are guaranteed to exist (as long as $x_1 \neq x_2$, which should be the case); furthermore, since 101 is prime, we are assured that distinct values will still remain distinct after the transformation.

Alternatively, you can just use **brute force** to search for the correct values of a and b. Just test all possible until you find the right pair.

Standard Toolbox: Complete Search

Can't figure out the right value? Just try everything.

- **Pros:** Always works
- **Cons:** Might be too slow (do the analysis)

The above computation is important in *proving* that a solution *always* exists... but "proof by AC" is an option in competitive programming as well.

In any case, apply $x \mapsto ax + b$ to all cells x in your hard-coded magic square, and you will have a Mondrialayagic square that solves the problem.