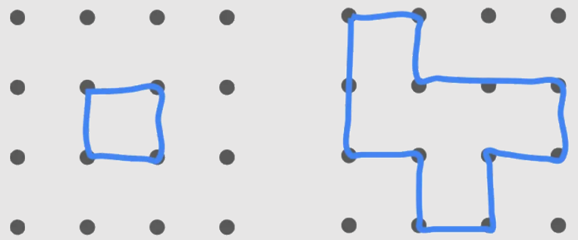
Slitherlink

This week's Fiddler on the Proof (9 February 2024) asks:

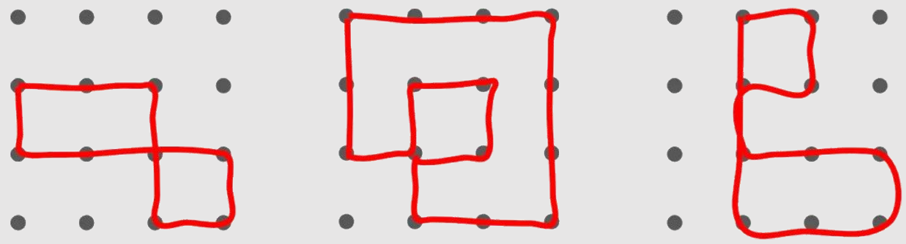
Nikoli the snake wants to slither along a loop through a four-by-four grid of points. To form a loop, Nikoli can connect any horizontally or vertically adjacent points with a line segment. However, Nikoli has certain standards when it comes to loop construction. In particular:

- The loop can never cross over itself.
- No two corners of the loop can meet at the same point.
- Once Nikoli has crossed the connection between two points, Nikoli can't cross it again (in either direction).

For example, the following two constructions are valid loops:



Meanwhile, the following three constructions are *not* valid. The one on the left crosses over itself, the one in the middle has two corners that meet at a single point, and the one on the right requires Nikoli to pass over the same line segment twice.



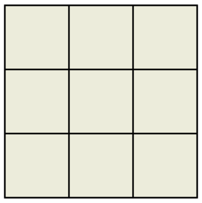
How many unique loops can Nikoli make on the four-by-four grid? (For any given loop, Nikoli can travel in two directions around it. However, these should still be counted as a *single* loop.)

Shapes Instead of Loops

I found it helpful to think of "shapes" instead of loops. Any properly formed loop encloses a shape, so counting the shapes is the same as counting loops.

You can think of a "shape" is a collection of one or more tiles that are connected to each other. (They are also called polyominos.) How do we find all the possible shapes?

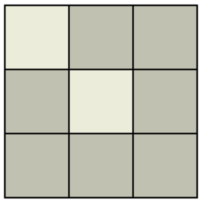
An easy way to do it is start with a 3-by-3 grid of cells, like this:



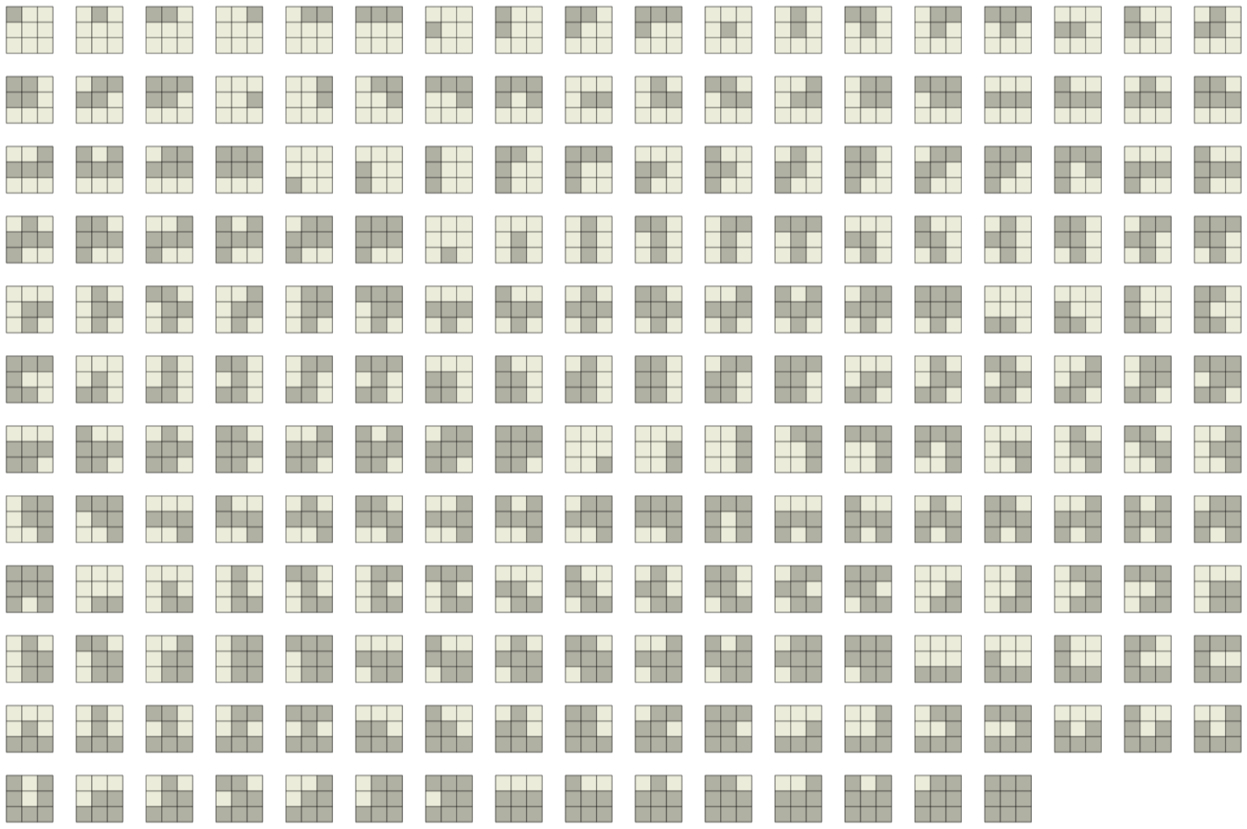
Each cell of the grid can either be part of a particular shape or not part of that shape. Since there are 9 cells, there are $2^9 = 512$ such combinations (or 511 if you don't include the empty set).

I checked each combination to see if all of its cells are connected to each other edge-to-edge. If not, then that combination doesn't work as a shape.

The other thing I checked for was whether the shape has a "hole" in it. A shape like the one below has to be excluded, in the case, because its boundary loop intersects with itself.



Here are all the combinations I found that qualify as valid shapes:

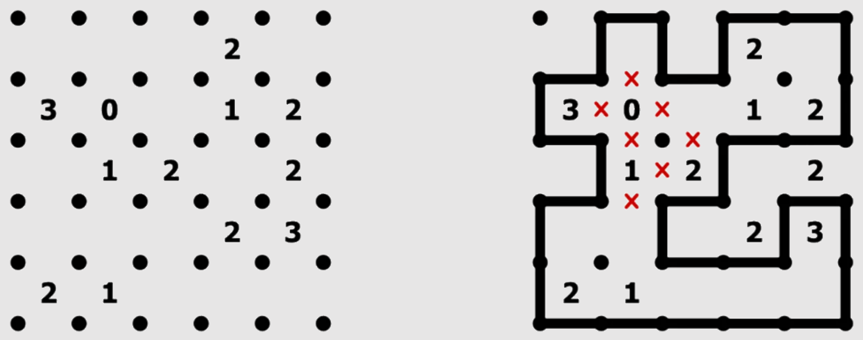


There are 213 of them.

Extra Credit

In the game Slitherlink, you connect adjacent points in a grid to form a loop that does not self-intersect, as described above. But what makes a Slitherlink a *puzzle* is that numbers are provided in some of the spaces between four grid points. These numbers specify how many of the four surrounding edges are present in the desired loop. Then it's entirely up to you to draw the loop.

For example, here's a puzzle I encountered on one of the previously linked sites, as well as the solution. (I marked red x's where I knew there couldn't be any edges—a helpful strategy when solving such puzzles.)

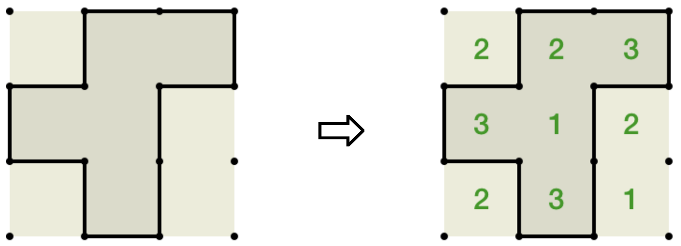


Got all that? Okay, so the Extra Credit is this: How many distinct Slitherlink *puzzles* can you create on a four-by-four grid of points? Each puzzle consists of a placement of numbers (from 0 to 4) between grid points, and must result in exactly one loop. Note that multiple distinct puzzles can result in the same loop, but again, each puzzle itself can have only one loop solution.

Note: What I'm calling a "four-by-four" grid is technically a "three-by-three" Slitherlink, since the latter describes the array for the *numbers*, rather than the *points*. But to keep things consistent between the puzzles this week, I'm steadfastly calling this a "four-by-four" grid.

We are already well on the way to solving this. We know the solutions (the shapes discovered above); we just need to figure out what puzzles lead to those solutions.

I worked on it one shape at a time. Here, for example, the r-pentomino shape. It is pretty straightforward to calculate how many surrounding edges each cell has:

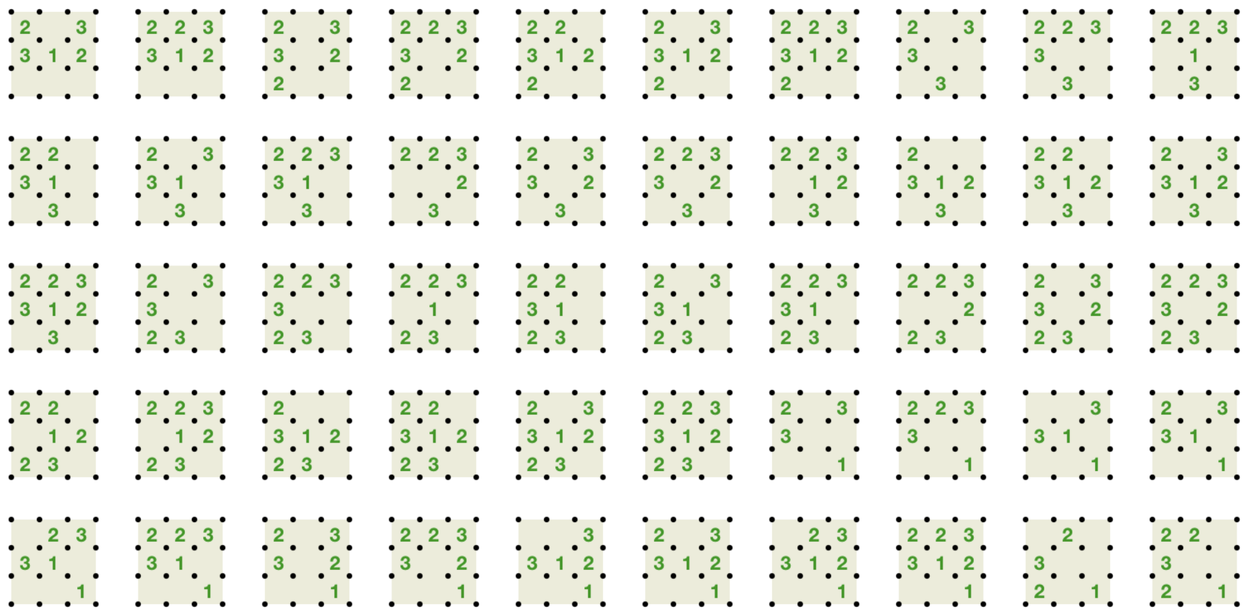


It turns out that these 9 numbers always lead to the same solution. I know this because I compared the numbers for this shape with the numbers for all other 212 shapes that are possible solutions, and none of them were the same. (This isn't always the case. See "Shapes with No Puzzles" below.)

In fact, you don't even need all 9 numbers. Some subsets of these numbers unambiguously lead to the same solution. To determine which subsets work and don't work, I looked at every combination of each number being present or not. There are $2^9 = 512$ possibilities.

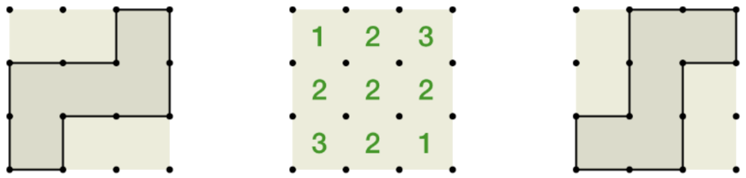
With each combination, I checked to see if the same combination of numbers appeared in the numbers for any other shape. If they do, then that combination has too few numbers in it to work as a puzzle.

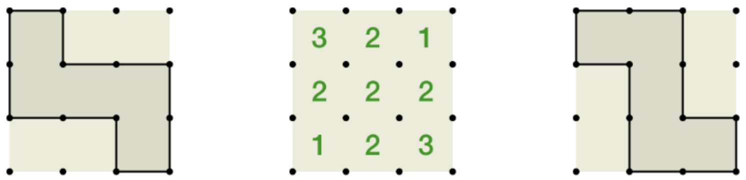
For the r-pentomino shape, it turns out that 150 different combinations of numbers that lead to the proper solution. Here are the first 50 of those puzzles:

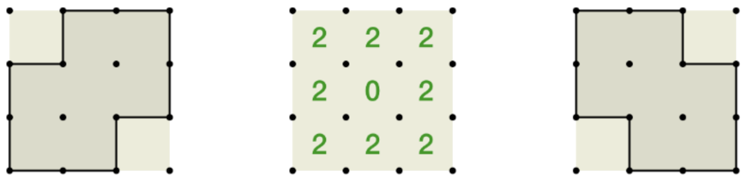


Shapes with No Puzzles

Some shapes have *no* puzzles associated with them! The reason for that is that they share a common numbering with another shape. Here are three pairs of shapes that have the same numbering as each other:

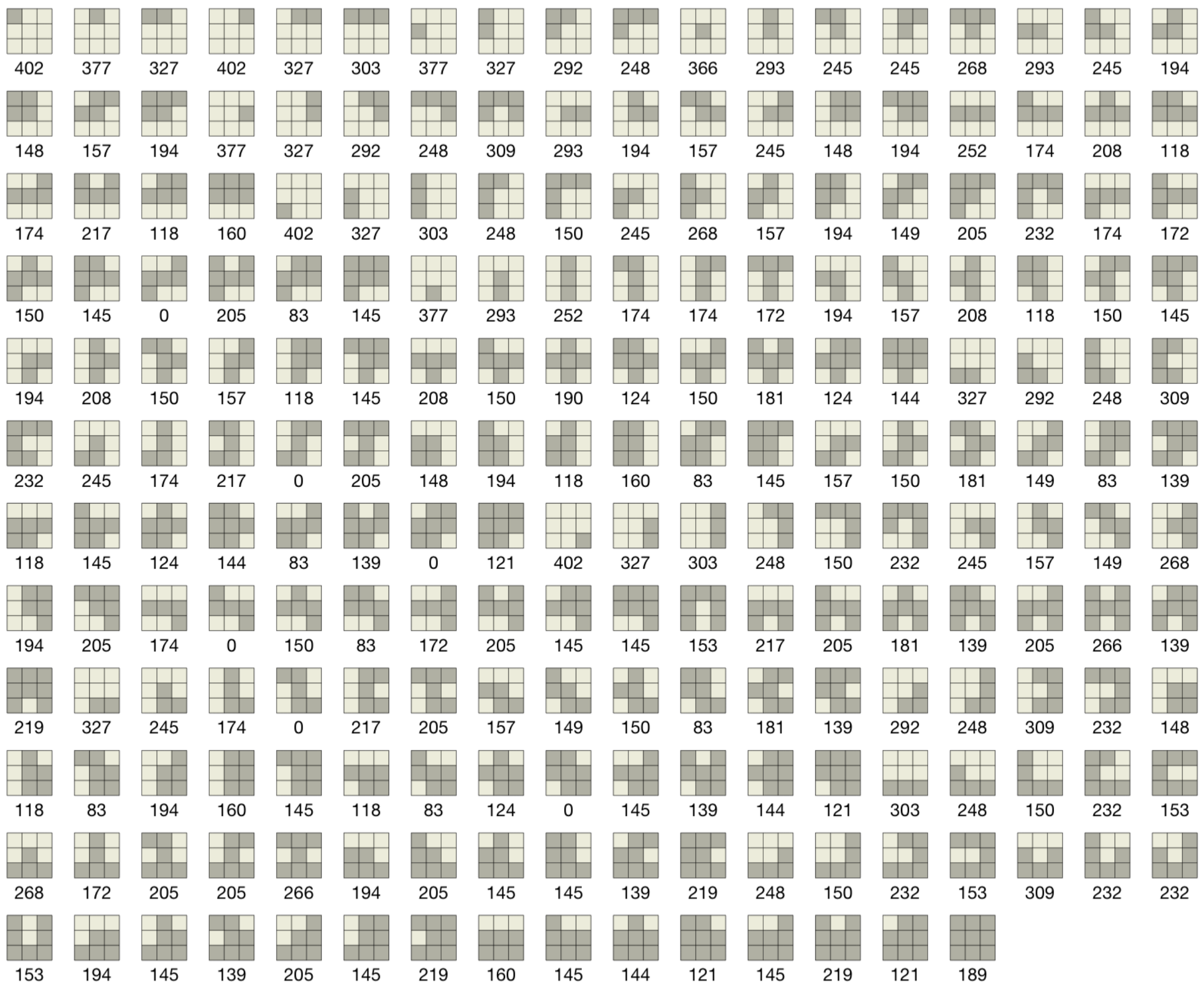






Adding Up the Puzzles

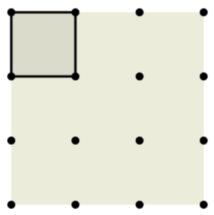
Okay, time to add up all the puzzles. I went shape-by-shape and counted the combinations of numbers that worked as a puzzle for each shape. Here are the totals I got:



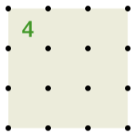
In all, there are 41,433 different puzzles.

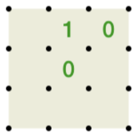
Minimal Puzzles

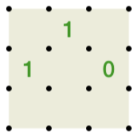
Most of the 41,433 puzzles above provide more numbers than needed to solve the puzzle. What if we counted only *minimal* puzzles — puzzles which have no redundant numbers in them? To see what I mean, consider this 1-by-1 shape:

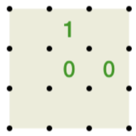


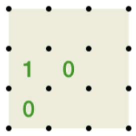
It has a total of 402 puzzles associated with it, but most of those puzzles contain redundant numbers. If we disallow redundant numbers, then this shape has just the 16 puzzles listed here:

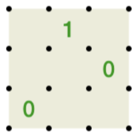


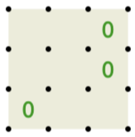


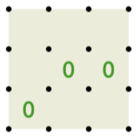


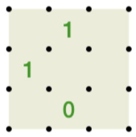


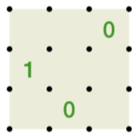


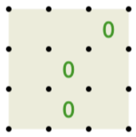


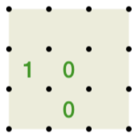


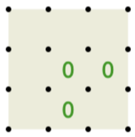


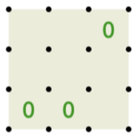


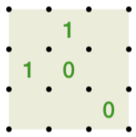


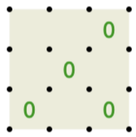












Adding up the number of minimal puzzles for all 207 shapes, I got a total of 5,299.