F.7 Problem 202: Laser Beam
AJ Rasure
Programming for Problem Solving
April 25, 2019

Setup:

For the program we want to know how many ways there are for a laser beam to enter an equilateral triangle and have a specified number of reflections before it leaves out of the same vertex that it entered. I choose to look at the laser beam as a straight line that continues through a reflection of the triangle rather than being reflected across the normal line at the same angle it incidents upon the mirrored side. This removed the need to know at what angle the laser beam enters the triangle and where it strikes the mirror. Instead I am able to treat it as a line a grid.

Program

The program itself is rather simple and needs to determine a few things:

1. If that amount of reflections is possible, then what row does it exist on.
2. What is the first column that an exit through the starting vertex is feasible.
3. Many of the available possibilities are not duplicates or leave the triangle early.

Variables

reflec: a variable that is the number of reflections to occur within the triangle before exiting.
row: a constant that is determined from the amount of reflections.
param: a list (currently commented off) that holds all of the rows and
total: The total number of correct exits counted (begins with zero).
col: the column currently being checked for vertices that are determined to be correct exits.

Code

If reflec is equal to 1, then the total amount of possible reflections is 1. *The first if statement simply looks at if you only want a single reflection within the triangle. This is a special case where the laser enters the triangle perpendicular to the first mirror and directly exits.*
elsif the row being looked at is a multiple of 3, there are no possible appropriate vertices. *These are also special cases and none of them will reflect without first leaving through an incorrect vertex.*
Is row an integer?

> col is equal to 1 plus row(mod 3) *this is the first vertex to check*
>
> > As long as row+1 is greater than or equal to col, continue with the function
> >
> > > if the greatest common factor for the row and col is 1, increase the total by 1
> > >
> > > then look at the next possible col *it will be 3 greater than the previous*

Print the total times 2 *The program only looks for half of the available as the sum is duplicated above and below the axis*

While the code is rather simple, it is effective because of how the problem is being interpretted.

Justification

The crux of the code is that it only looks possible "lasers" that have not left the triangle early. To begin looking at this I created a diagram that is a series of triangles that are reflections of each other via the shared wall. I also labeled the vertices of the triangles to keep track of what where a laser was entering or leaving. To maintain with the convention established in the prompt, a laser is intended to enter and leave only through the **C** vertex.

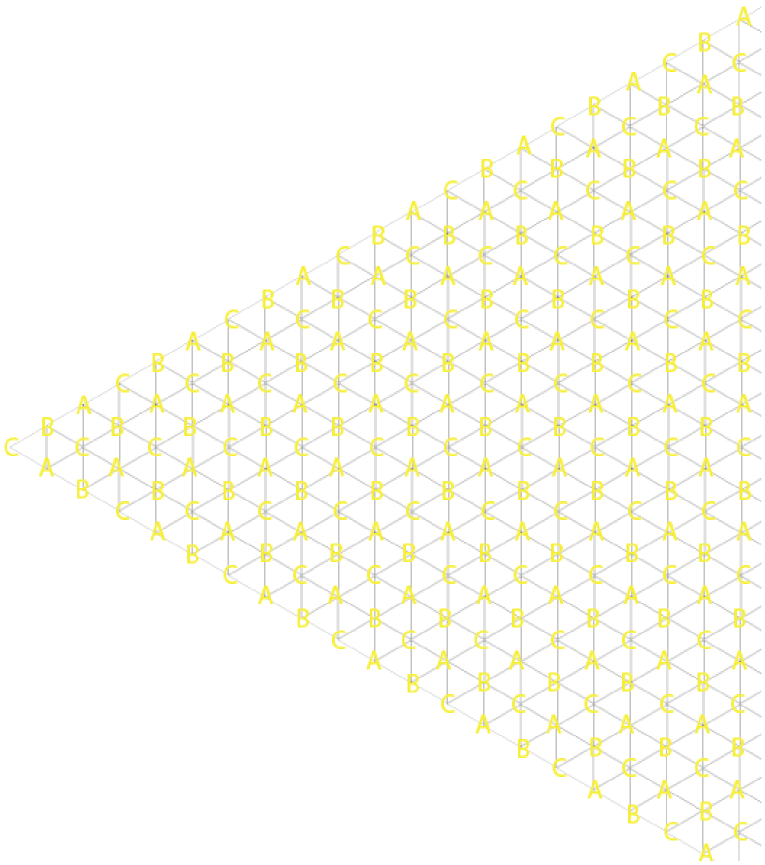

Figure 1: Triangular Grid

To see how a laser acts lets quickly look at what happens when a laser is reflected either 1 time or 7 times.

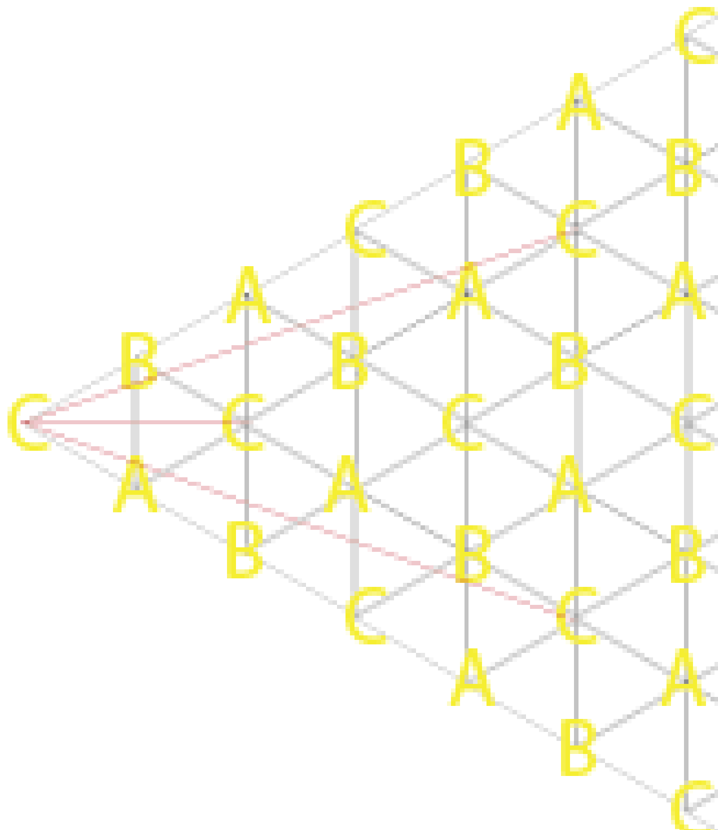

Figure 2: 1 and 7 reflections

First a laser was created starting from the origin vertex going to the first available exit **C** vertex. This results in a laser crossing a single mirror and exiting and is the same as a laser entering the triangle perpendicular to the opposing mirrored surface. The next two lasers each go to the next available **C** vertex that does not require them to exit a **B** or a **A** vertex prior, nor will it cross a **C** vertex a previous laser has already exited. This laser crosses 7 surfaces and correlates to 7 reflections.

If the laser was to travel directly down the sides of our grid, it would exit the first available vertex every time. This means that any **C** vertex that lies on the outside of the grid is not a potential appropriate exit. To see why the laser can not cross a previous **C** vertex or any other vertex inside the grid let us look at the laser as a line segment on a normal graph.

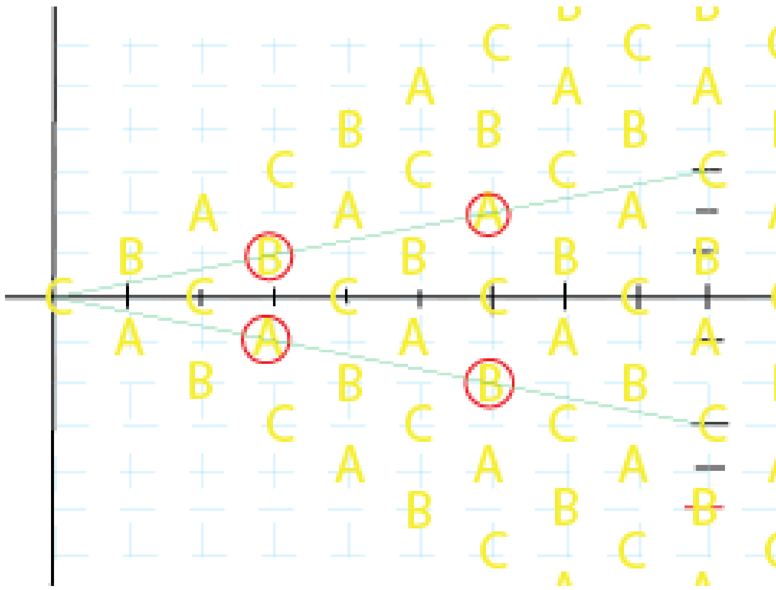

Figure 3: Rectangular Grid

This laser should have 15 reflections however it leaves early before ever reaching the appropriate **C** vertex. This is easy to see from the slope of the line. For this laser the slope is $\frac{3}{9}$. This fraction can be reduced down to $\frac{1}{3}$ and we can see that $\frac{2}{6}$ is also a multiple of the reduced slope. At both of these points the laser encounters a vertex other than the **C** it is trying to leave. From this we can determine that a laser will only reach the appropriate vertex if the slope created by the laser is irreducible, or that the greatest common factor for the row and column of the exit vertex is 1.

If the laser is treated as a line starting from the origin $(0, 0)$ and ending at the coordinates (x, y) where $x, y \in \mathbb{Z}$ then the slope of the line is equal to $\frac{y}{x}$ as long as the greatest common denominator is 1. If the greatest common denominator is not 1 then the slope of the line will not be $\frac{y}{x}$. Consider $x = ra$ and $y = rb$ where $r, b, a \in \mathbb{Z}$ and $\neq 1, 0$. In this case the slope of the line would be $\frac{rb}{ra} = \frac{b}{a}$ which allows the laser to lave after b columns and a rows, that is prior to the appropriate exit.

Next we look at when the what is the column number we should start looking to see if it as an appropriate exit. We start by saying that the first **C** vertex is our first column and we continue to travel down the lower angle of our triangular grid. The next column starts with an **A** vertex and then we get a **B** to start the column. This pattern repeats itself indefinitely. We give each column a starting value based on row (mod 3). The row is determined by counting down from the vertex to the column you are starting at. For example, the laser that we saw earlier with 7 reflections starts at row 5, then the column value for the first **C** vertex is 3. Now as you move vertically through the triangular grid you increase the column value by 3 to reach all vertices. Because the lasers are reflected the same above and below the mid line, we only need to count half of the number of columns that are available for a given row. As it is an equal lateral triangle, if there are 5 rows, there are 5 columns and we start by saying **C** is in the third column.

How do we know which row to start on?

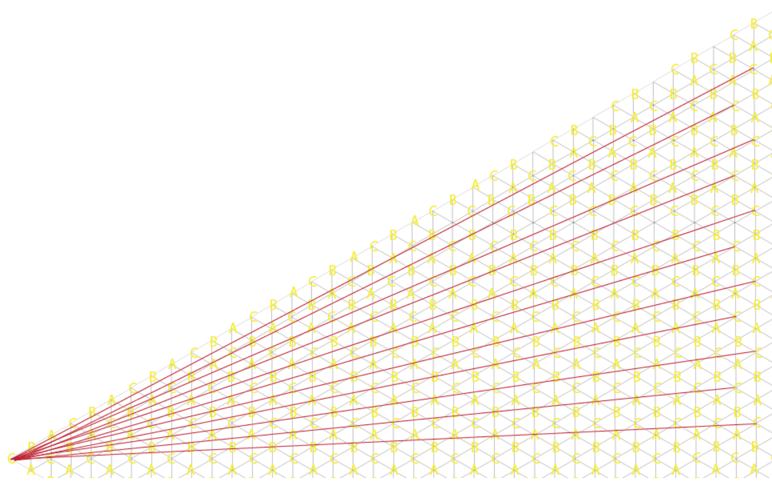

Figure 4: 69 and 71 reflections

As I was drawing out the lines of the lasers to see if I could spot a pattern, I noticed that a laser with a specified row exiting any of the **C** vertices had the same number of reflections, whether that **C** vertex was actually possible due to the slope of the line. The relation between the two was that the number of reflections equaled the row times two minus 3 ($reflec = row * 2 - 3$). Turning this around to determine what row to start on was simple algebra, ($row = (reflec + 3)/2$) and in such a manor I could know what row to look at for a given number of reflections. Now this works even if the **C** vertices in the row are not appropriate. If you look at Figure 4 you will see drawn the top half of lasers for 69 and 71 reflections. All of the 71 reflections are exit a **C** vertex and not early, however all of the 69 reflections exit early. This is due to the slopes being reducible.

I tried to figure out a rigorous method for proving why this was so but any method I tried either was self referential in a way that made it not valid, such as how I could use it to predict the next number of reflections for the next row, but this did not prove that that many reflections could only exist for that row.

Conclusion

This project took many turns from start to finish and I began thinking about it in a way that, now, I do not believe would have been feasible. I am glad I was able to envision an different way of seeing the problem and will continue to try and prove that the equations that make this work rigorously. The final answer for how many ways a laser can enter an equilateral triangle and bounce 12017639147 times before departing the same vertex it entered is 1209002624 different ways. This was verified on Project Euler.