MariHacks 2025 Math challenge : Billiards

2-dimensional Billiards are studied in abstract mathematics, but their visual representation require the use of computer tools which you will develop in this challenge. These tools are a useful support to understand the math and make conjectures about the long-term behaviour of billiards.

Part 1: The warmup challenge

As a warmup, here is a very specific problem that you should solve. Part of your code should be reinvested in Part 2.

Imagine a square billiard table, where the billiard is frictionless and the ball travels at constant speed. For simplicity, the ball starts travelling from the southernmost edge (at some undefined point) with initial velocity pointing towards the eastmost edge of the table (at an undefined angle less than 45 degrees) and the physics are not the ones you know: namely, the angle between the incoming ball and the outgoing ball is always a right angle.

- What can you say about the long-term behaviour of the ball's trajectory?
- Can you compute some of the relevant math with pen and paper before you start coding?
- How does your answer depend on the initial conditions of the problem?
- Can you draw a useful visual figure in Python?
- Once you can draw it, can you animate it?

Restrictions:

Do not use fancy libraries and software, billiards have already been coded by mathematicians, but the point of this challenge is to redo all this work yourselves, almost from scratch. Use Python and only the following libraries: numpy, math, and matplotlib, that's it. For animations: you'll need "matplotlib import animation", specifically, "from matplotlib.animation import FuncAnimation" then look up the documentation on matplotlib's website.

Needless to say, using ChatGPT or something similar defeats the purpose of a coding challenge. Don't use it!

Do not write a program that does everything, write multiple programs. Once one of them works, do not change it, make a copy (or branching if you know what that means); recall the engineer's motto: "if it ain't broken, don't fix it!"

MariHacks 2025 Math challenge : Billiards

Part 2: the actual challenge

Move on to physically realistic billiards, yet still frictionless: now the ball reflects just like light does. What can you say about the long-term behaviour of the billiard depending on initial conditions and the shape of the table? Do not animate things first, draw a static version at time T, then animate it in a separate program.

Suggestions for investigation:

- What happens if the table is not a square but a pentagon, or something else?
- Can you generalize to three dimensions and animate it?
- Can you introduce multiple balls that bounce off each other?
- What else can you think of?
- Push the limits as much as you can once you have a functioning program. Write multiple programs, one for each iteration / generalization.

Restrictions: Same as before, code mostly from scratch: use Python, numpy and matplotlib, nothing fancier than that.

Judging criteria

Your project will be judged based on the following criteria and ability to answer certain questions:

- Adaptability / flexibility of the code: can it handle different situations based on user input?
- Performance: does your code compute things fast when pushed to extreme cases?
- Can you explain the structure of your code and the key points of the algorithms / functions you created?
- Can you explain the underlying math notions? Can you show some pen and paper computations you had to do to achieve the results you got?
- Based on your program from part 1, what can you say about the long-term behavior of this billiard? Same question for part 2.