Problem Statement

Implement Conway's Game of Life in 64-bit signed integer space.

Imagine a 2D grid - each cell (coordinate) can be either "alive" or "dead". Every generation of the simulation, the system ticks forward. Each cell's value changes according to the following:

- If an "alive" cell had less than 2 or more than 3 alive neighbors (in any of the 8 surrounding cells), it becomes dead.
- If a "dead" cell had *exactly* 3 alive neighbors, it becomes alive.

Your input is a list of integer coordinates for live cells in the Life 1.06 format. They could be anywhere in the signed 64-bit range. This means the board could be very large!

Sample input:

```
#Life 1.06
0 1
1 2
2 0
2 1
2 2
-2000000000000 -2000000000000
-2000000000001 -2000000000001
-2000000000000 -2000000000001
```

Your program should read the state of the simulation from standard input, run 10 iterations of the Game of Life, and print the result to standard output in Life 1.06 format.

Please don’t spend more than 3 hours on your solution. Feel free to allocate that time in a manner that works best for your schedule. You may work in any language you prefer.

We're most interested in both the technical aspects of how you deal with very large integers and how you go about solving the problem. At the onsite, be prepared to discuss your solution, including the choices and tradeoffs you made. Please send your solution to your recruiting partner in advance of the onsite.