BPP Final Project

Name: Tommy Tutone
Student Number: 8675309

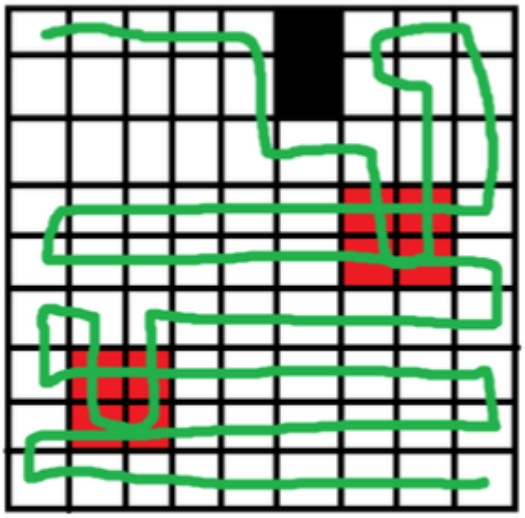

Figure 1: Movement example of my bot. Red squares are stains, black squares are obstacles.

1 Vacuum Bot Logic

My Bot (a.k.a. noStainNoGain) traverses each map from top to bottom, left to right. It prioritizes movement according to the list below:

- If a stain is in vision, clean that stain immediately
- If an obstacle is in the bot's current direction, move around it by moving downwards
- If not stains or obstacles are in vision, move rightwards on even rows and leftwards on odd rows

See Figure 1 for reference.

2 Performance

I tested my bot on 50 maps, which were generated by `map_generator.py`, code that I wrote to randomly generate maps to test on. My bot was capable of successfully solving 6-grade, 7-grade, 8-grade, and 9-grade maps. Labyrinth

(10-grade) maps were impossible, because my bot only moves to cells it has not visited before, hence is unable to backtrack if it reaches a dead end. My bot always performs better or equal than BruteBot.

In Table 1, I present a summary of its performance.

Map Size	**My Bot Performance**	**Brute Bot Performance**
6	.76	.5
7	.76	.1
8	.57	0
9	.56	0
10	0	0

Table 1: Comparison of bot performance. Performance is measured as the ratio of remaining energy to total energy. The average for each map grade level is provided. 10 maps per grade level were tested.

3 Discussion and potential improvements

Before reaching the above results, I tried various approaches. I first tried an implementation where the bot would prioritize traversing the entire map, meanwhile keeping stain locations in memory (to clean up afterwards). That approach was found to be significantly worse, performance-wise.

In order to solve 10-grade maps, my bot should be able to backtrack out of dead ends. I unfortunately did not have enough time to implement that successfully, even though I tried. I caused my sister's computer to catch fire last week, therefore I deleted that part of the code. In order to make this bot even more efficient, I should have made it skip two lines and only go up or down when necessary (to clean stains).

I would like to thank Github Copilot for generating the code used in this assignment; I basically copy-pasted everything and I hope I won't get caught by the plagiarism check. This report was generated by ChatGPT.

References

[1] Al Sweigart. *Automate the boring stuff with Python: practical programming for total beginners*. No Starch Press, 2019.