Adrian Gellert

Professor Maximillian Bender

CS231 Lecture A, Lab D

31 October 2022

Adrian's Project 5: Grid Search and Comparison of Breadth-First Search and Depth-First Search

ABSTRACT

In this project we used a new data structure, specifically that of queues, and compared its functionality with stacks. Specifically, we compared these two data structures by implementing them in breadth-first search and depth-first search methods. We then applied these algorithms to a simulation which allowed us to analyze how a queue keeps track of cells and travels through a grid of cells versus how stacks do this. The end goal was to better understand these two structures and their differences.

RESULTS

1. What is the relationship between the density of obstacles to probability of reaching the target?

Relationship between the density of obstacles to probability of reaching the target	
Chance that cell is an obstacle	Percentage that have a result (100 runs, used a counter to count number of false/true returns)
.2	98%
.4	43%
.6	2%
.8	0%

Given the table above, we can see that, as the density of obstacles increases, the probability of reaching the target decreases. It is an inverse relationship.

2. What is the relationship between the minimal length path from the start to the target to the number of vertices visited by BFS?

Average length of minimal length path (10 runs of BFS) = 41 + 41 + 25 + 33 + 26 + 41 + 28 + 37 + 14 + 28 = 31.4
Average number of vertices visited by BFS (10 runs of BFS) = 442 + 741 + 424 + 597 + 494 + 707 + 679 + 680 + 669 = 543.3

543/31 is about 17.5.

On average the number of vertices visited for BFS is 17.5x more than the length of the shortest path.

3. What is the relationship between the minimal length path from the start to the target to the number of vertices visited by DFS?

Average length of minimal length path (10 runs of BFS) = 41 + 41 + 25 + 33 + 26 + 41 + 28 + 37 + 14 + 28 = 31.4
Average number of vertices visited by DFS (10 runs of DFS) = 320 + 11 + 62 + 549 + 560 + 17 + 585 + 533 + 448 + 699 = 378.4

378.5/31.5 = about 12.

On average the number of vertices visited for DFS is 12x more than the length of the shortest path.

4. What is the relationship between the minimal length path from the start to the target to the length of the path found by DFS?

Average length of path BFS (10 runs) = 41 + 41 + 25 + 33 + 26 + 41 + 28 + 37 + 14 + 28 = 31.4
Average length of path DFS (10 runs) = 120 + 5 + 28 + 247 + 230 + 196 + 226 + 34 + 260 + 120 = 146.6

146.5/31.5 = 4.65

On average the length of the path for DFS is about 4.65 longer than the path for BFS.

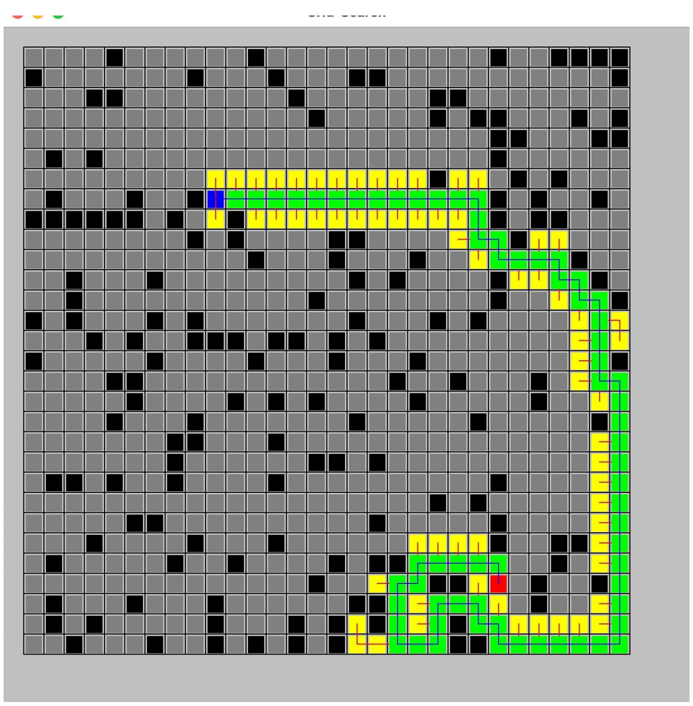

Figure 1: Depth First Search Example

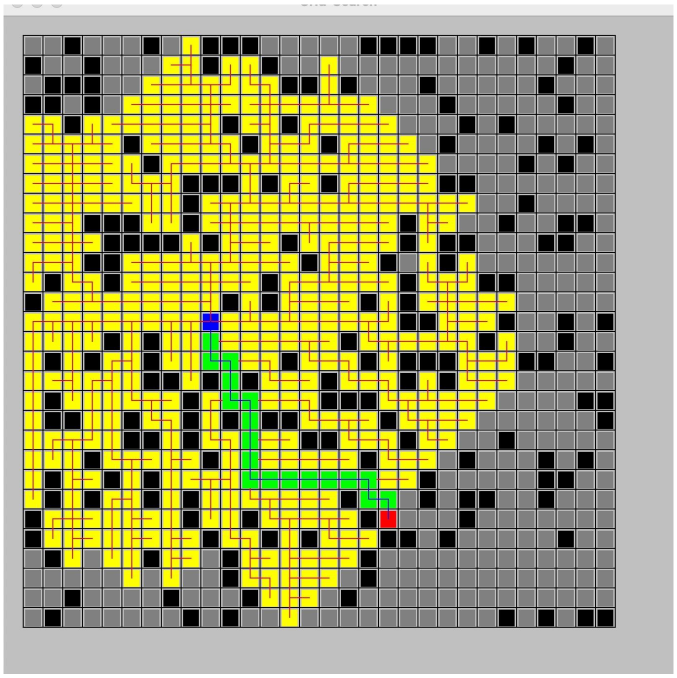

Figure 2: Breadth First Search Example

REFLECTION

A queue allows us to check the neighbors of each initial step to a next cell before going to a subsequent step. Thus, we are able to find the shortest possible path to the target. In contrast, in a stack, where the process of analyzation is first in last out rather than first in first out, we travel through the longest possible path before analyzing neighbors of the top of the stack. Thus, rather than finding all possible paths, which allows us to find the shortest path, we are left with finding a longer path.

COLLABORATION

I got help from Catherine (“Jaime”) Yockey for code that we had similar errors that she had fixed. I also used pseudocode discussed in Discord and lecture code.