Patrick Ekel cs2420

Assignment09 Analysis

1: My programming partner was Kyle Price (u1018878). Kyle submitted the source code.

2: Kyle was a good programming partner. We spoke about the assignment and planned out how we were going to approach and solve the assignment. Allocating time to work on the assignment together was not easy since both our schedules were busy this week, but we were able to get a lot done of the weekend which proved helpful. We communicated well via emails and texts but we were not able to meet as much in person the two days before submitting the assignment, so in the future I would try and allocate a bit more time to communicate and work on the assignment together in person.

3) The straight-line distance to the goal can be a good indicator in some situations and a bad indicator in other situations, making it an unreliable indicator overall. If there are few walls, or walls that do not greatly affects the path to the goal, the straight-line distance to the goal from the start is a good indicator of the running time of the algorithm, but there could be walls which block off access to goal until many other nodes are traversed. In this case the straight-line distance is a poor indicator of the running time of the algorithm.

4) The straight-line distance is the number of blocks away from the solution the goal is from the start, assuming walls can be ignored. The actual solution path length is the number of blocks the path spans since the path can't traverse over walls. An example maze of this was the "turnMaze". The goal block was only two blocks away from the start, but there was a long wall separating the two blocks. Because of this wall the path had to go all the way down to the bottom and before traversing back up towards the goal. The actual path length is a more accurate representation of actual run time for it is a better indicator of how many nodes had to be checked in order to reach the goal. The straight-line distance could potentially be a good indicator as well, but it also has the potential to be a very poor indicator of run time performance, making it a very unreliable indicator.

5) If the maze is square and the side length is considered to be N, then there would be N^2 spaces in the maze. The worst-case scenario would be N^2 if each node needed to be checked. This could happen even on a maze that has very few mazes. If the start is in one corner (let's say top left) and the goal is in the opposing corner (let's say bottom right), then the entire board would need to be check, resulting in a performance of O(N^2). Even if the board was densely filled with walls, the worst-case performance would still be O(N^2) since in the worst case every space would need to be checked. After the goal is found the algorithm traverses back to the starting location. This could take up to N^2 traversals, but this occurs after the searching so the worst-case complexity is N^2 + N^2 which simplifies to O(N^2).

6: I spend about 20 hours on this assignment.