Assignment 09 Analysis
Torin McDonald

1) Nathaniel Coleman was my programming partner. He submitted the source code.
2) Nathaniel was a great programming partner. He had a lot of great ideas for going above the basic code we had come up with to get the program to work.
3) It would depend on the density of the graph. In general, the straight line distance between the start and goal would not have an impact on runtime.
4) The straight line distance is the direct distance between the start and goal, while the actual path has to navigate walls. If you have a graph that had a very small density (very few walls), the straight line distance would be a relatively good indicator of runtime. For example, if you had a graph where the only walls were on the perimeter, the straight line distance would be a direct indicator of runtime because there wouldn't be any obstacles for the path to navigate. On the other hand, if you had a graph that was extremely dense, the straight line distance would be a very bad indicator of run time. Even if the straight line distance were extremely small (say two), the actual path would be extremely long. The best indicator of runtime is the distance of the actual path.
5) The worst case performance of this situation would be O(N^2) because you would have to traverse N^2 number of vertices. If you consider each space to be a vertex however, the complexity is O(2N) which becomes O(N). The worst case performance would occur with an extremely dense graph. If the graph isn't dense, then the complexity becomes O(N) plus the number of edges.
6) I spend around 10 hours on this assignment.