Assignment 1

By: Chinmayee Prabhakar

Code and algorithm explanation

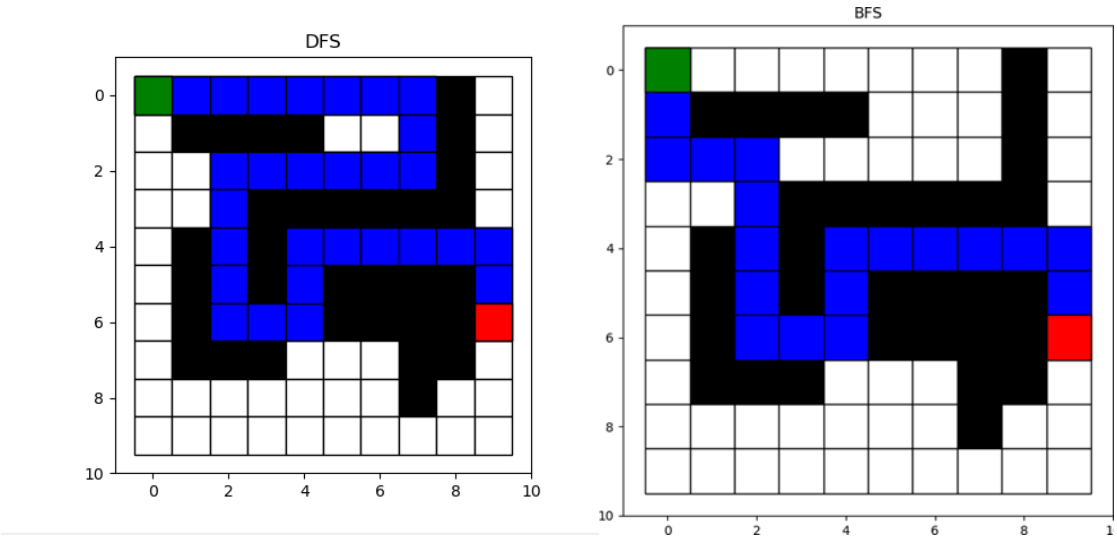
The algorithm takes two approaches Breadth First Search (BFS) and Depth First Search (DFS) to find the goal node in a 2 dimensional map initialized with the Start and Goal as well as obstacle nodes.

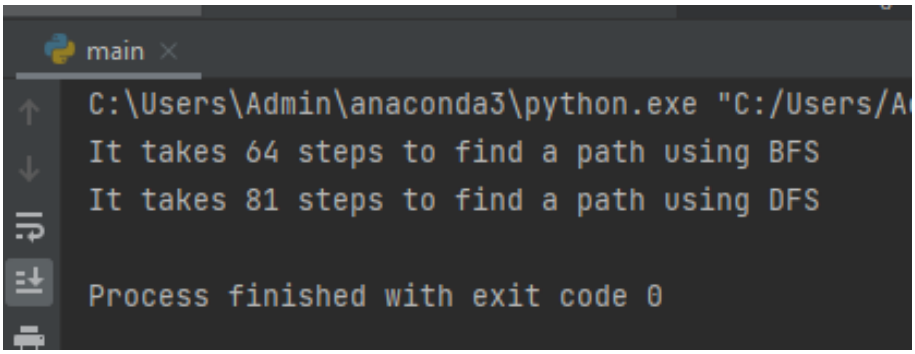
We initially split the grid elements to two sets of nodes : Lists with nodes containing Obstacles and Lists with nodes that are Obstacle Free.

When it comes to the Bread First Search implementation, the algorithm starts from the source node in the 'Q' list, finds the adjacent nodes that are obstacle free and not already a part of the 'Q' list and appends these nodes to 'Q'. This way , starting from 'start' node we move forward with its adjacent nodes and their adjacent nodes iteratively through the Q set until we reach the 'Goal' node. Upon reaching the 'Goal', the iteration going through each element in 'Q' using the 'For Loop' is broken out of and the collected parent nodes from each node during the BFS implementation is used for tracing the path from start to goal.

Similarly, when implementing Depth First Search we start with the 'Source' node and compute its adjacent nodes out of which only one that satisfies two conditions i.e. is not an obstacle node and is not already present in 'Q' list is selected and its adjacent nodes are further calculated and selected continuously unless and until we reach a node that doesn't have any viable adjacent nodes to trace through. Then the previous node in the 'Visited' list is selected and its adjacent nodes are further traversed through in the same way until we reach the 'Goal'. Then the parent nodes are traced and the path from start to goal is traced.

1. For the given test case (map.csv) in the assignment the results for BFS and DFS are as follows:





The results are very different as the approaches are different. In DFS the implementation takes longer or more number of steps to reach the goal whereas in BFS the number of steps are less.