Portfolio Project Problem algorithm description
My algorithm to solve this graph traversal problem uses a queue and a breadth-first search to find the shortest path from coordinates (a,b) to (x,y).

We make a queue that keeps track of the order of coordinates to process. In the queue are tuples that contain a coordinate to traverse(we will call this the traversal coordinate) and the previous coordinate on the path(we will call this the previous coordinate). For example, if (1,1) is to be traversed and its previous coordinate is (0,1), then the tuple to be put into the queue is ((1,1), (0,1)).

We also make a deep copy of the board which will be modified and will eventually be used to make a list of coordinates that form the shortest path.

To kick off the algorithm, a tuple with the starting coordinate as both the traversal and previous coordinates is added to the queue. Next, we enter a loop. In this loop, the first tuple in the queue is popped and is saved into a variable which we can call “Current”. In the board copy, the cell at the traversal coordinate of Current is initialized to hold the previous coordinate of Current. Then, each adjacent cell(any cell directly above, below, right, or left) to the current cell is added to the queue if the cell has not been traversed, is not blocked, and the cell value at the destination coordinate has not been changed. This loop emulates a breadth-first search. This loop continues until the queue is empty.

After exiting the loop, we can now make a list of coordinates forming the shortest path using the modified board copy. The previous coordinate values inside the cells of the board copy will tell us which way to go to form the shortest path if we first look at coordinate (x,y). Therefore, we first set coordinate (x,y) to be the coordinate of the current cell being traversed(let’s call this “Current2”). Then, inside a loop we append the coordinate of Current2 to our shortest path list and change the value of Current2 to be the previous coordinate value found inside the cell of Current2. This loop continues until we reach coordinate(a,b). When the loop ceases, we append coordinate (a,b) as our last value in the list.

Finally, we reverse the shortest path list in order to obtain the correct order of the shortest path from (a,b) to (x,y).

Portfolio Project Problem solution time complexity
The time complexity of my solution is O(m*n) where m is the number of columns and n is the number of rows in the given puzzle.