Project 3 Report

Tarzan and Mojo

Question 1:

Graph will be simple (no multiple edges or self loops).

Graph will be Directed.

Graph will be weighted, where weights represent distance moved from vertex to vertex.

Vertices correspond to a single space in the maze.

Edges correspond to path from vertex to vertex and have weights that represent the distance.

A vertex is represented as a pair of integers.

I created a struct to represent edges where each edge has a pair of integers that corresponds to the ending vertex, each edge also has an integer w to represent the weight.

I also created a graph class that has a map mapping a pair of integers (vertex) to a vector of Edges. Representing an adjacency list graph. The class also has a set that stores all the vertices. And a function to add edges between two vertices.

Question 2:

After reading from the file and storing the maze in a 2D vector. I create a graph by adding edges between all of the vertices according to the direction of each vertex connecting the third and fourth vertices from it. I then mark all vertices as UNDISCOVERED. I then call DFS on the vertex representing start position. I perform normal DFS on the graph. From the way DFS whenever I explore the vertex where jojo is I break out of the loop and stop making any further calls to DFS and I start adding to a stack the vertex and the edge weight. After all the recursive calls return, the stack has the reverse path from jojo to tarzan, I empty out the stack to a vector which then stores the path from tarzan to jojo in the vector. I then print out the path to the output file.