**Question 1.1: Describe how you would frame the maze solver as a search problem.**

A maze solver can be described as a search problem by representing the maze a connection of nodes; these nodes represent each possible position of a player. Each node will have neighbourhood of nodes that it is connected to. We can specify what the start and end nodes are and begin searching from the start node. At each node we will check if it is the goal, if yes then goal found, if no we move the one of the nodes in the current node’s neighbourhood. Our path will be each node that we have visited.

**Question 1.2: Solve the maze using depth-first search.**

1. **Briefly outline the depth-first algorithm.**

The depth first algorithm consists of traversing a given path a far as it can possibly go, we use a stack data structure to store the nodes that will be traversed, coupled with a visited list containing all nodes that have been visited; this is to avoid infinite loops in the search. The search starts using our