**The Maze**

**Introduction:**

* **Maze games** are puzzles where you are at (A) start position and navigate through a complex network of paths, walls, and dead-ends to reach the exit (B). The goal could be completing the maze as quickly as possible.
* **Maze Representation:**
  + Represent the maze as a 2D grid where each cell can be either a wall (‘#’), a space (’ '), the start position (‘A’), or the exit (‘B’).
* **Agent:**
  + Start from the entrance (‘A’).
  + Explore adjacent cells (up, right, down, left)
  + Repeat until you reach the exit (‘B’).

**Solution**:

1. **Stack**:last-in first-out data type: Add the nodes into the frontier. If the frontier contains a lot of nodes, the **latest** node added to the frontier will be removed and continue the loop until the solution is found.
2. **Queue**:first-in first-out data type: Add the nodes into the frontier. If the frontier contains a lot of nodes, the **first** node added to the frontier will be removed and continue the loop until the solution is found.

**Function**:

* Frontier: An array includes nodes
* Node: the places in the maze
* Solution: The array contains the nodes to show the way from A to B
* Maze: to print the maze: walls, spaces, A, B