1. **Represent the problem of finding the exit of a maze a search space.**
   • States: A state description specifies the location inside or outside the maze.
   • Initial state: Any state can be designated as the initial state. Most frequent state is at the entrance of the maze.
   • Actions: Move forward, move backwards, turn right, turn left.
   • Transition model: Given a state and action, this returns a new state, or well a new position inside the maze.
   • Goal test: Check if the current state is outside the maze.
   • Path cost: Each step costs 1, so the path cost is the number of steps in the path.
2. **Represent a search space (for search algorithms) from your favourite video game/novel/comic/sport, etc … Remember that the search space represents possible states, but it is different from a state machine or an automata.**

   Football
   • States: The state is determined by:
   -current score
   -current time
   -ball position in the field
   -player position in the field
   -current team with ball possesion
   -player position [role](i.e. forward, defender, goalkeeper)
   • Initial state:
   -current score: 0-0
   -current time: 0:00
   -ball position in the field: center
   -player position in the field: any position on it's team side of the field
   -current team with ball possesion: Team with the initial "ball-kick"
   -player position: previously determined
   • Actions:
   Attack:
   -shoot
   -pass
   -field ball
   -header
   Defend:
   -tackle
   -sliding tackle
   Neutral:
   -move forward
   -turn right
   -turn left
   -move backwards
   -sprint
   -jump
   Goalkeeper:
   -grab ball

-drop ball

- Transition model: Every second passed in the game will determine a new state.
- Goal test: Check if the current state has a winning score & game time is done.
- Path cost: This depends on the physical effort, injury ratio, sanction ratio & time spent.