CS5700 - Project Assignment 3

Adversarial Search

March 27, 2024

Deadline: Wednesday, April 17th, by midnight

Submission Format: PDF file containing copy-pasted code

Late Submissions: No late submissions will be accepted

Accepted Programming Languages: Java, Python, C, or C++

Work Ethic: Academic integrity is of utmost importance. Each student is expected to complete this assignment independently, without seeking help from other students or external individuals, including engineers or professionals. Any violation of this code of honesty will be taken seriously and may result in penalties in accordance with the rules and regulations of the University of Central Missouri.

Description

Adversarial Maze is a two-player game played on a grid. Each player controls a character, and the goal of the game is to reach the other player's side of the grid. However, the players cannot move through walls.

On each turn, a player can move their character one square in any direction, as long as the square is not occupied by a wall or the other player's character. If a player moves their character onto the same square as the other player's character, they capture the other player's character. The captured character is then removed from the game and the capturer wins automatically.

The first player to reach the other player's side of the grid wins the game.

Exercise 1

Implement an adversarial search algorithm to play Adversarial Maze. Your program should be able to play against a notable opponent, and it should be able to win as often as possible. Please remember the minimax algorithm, and then try to implement that.

You should follow some rules while implementing the algorithm:

- Your implementation should work with the given topology
- Your implementation should calculate the best action that could be done by two of the parties.
- Include comments and clear documentation in your code for clarity.

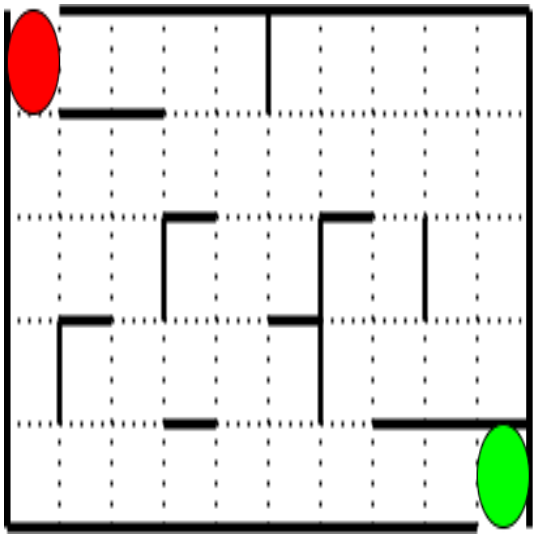

Figure 1: The maze topology.

Submission Guidelines

- Each student must submit their assignment individually.
- Submit your assignment as a PDF file.
- The PDF file should include BOTH Code for the Task.
- Ensure that your code and screenshots are clearly labeled and organized within the PDF.
- Submit your assignment by the specified deadline. Late submissions will not be accepted.
- Violations of academic integrity will not be tolerated and will result in serious consequences.

Evaluation Criteria

Your assignment will be evaluated based on the following criteria:

- Correctness of your implementations.
- Efficiency and clarity of your code.
- Proper documentation and comments.
- Adherence to the submission format and deadline.