**Snake Game with Search Algorithms**

**Introduction:** This document presents the collaborative effort of M Ali Awan, Daniyal Atique, and Usman Mughal in developing a Snake game with three distinct search algorithms: A\*, Greedy Best First Search, and Breadth-First Search. The aim of this project is to compare the effectiveness of these algorithms in solving the Snake puzzle.

**Implementation Details** Files

• main.py: Contains the primary game loop and integrates the search algorithms.

• AgentSnake.py: Defines the AgentSnake class with methods for each search algorithm. • AStar.py, GreedyBestFirst.py, BreadthFirst.py: Implementations of the A\*, Greedy Best First Search, and Breadth-First Search algorithms, respectively.

• State.py: Defines the SnakeState class representing the game state.

• View.py: Implements the SnakeViewer class for displaying the game. Search Algorithms

1. A\*: A\* is an informed search algorithm that uses a heuristic to find the path to the goal state with the lowest cost.
2. Greedy Best First Search: Greedy Best First Search is also an informed search algorithm that selects the path which appears to be the best at the current state.
3. Breadth-First Search: Breadth-First Search is an uninformed search algorithm that explores all the nodes at the present depth before moving on to the nodes at the next depth level. Running the Game
4. Upload all Python files (main.py, AgentSnake.py, AStar.py, GreedyBestFirst.py, BreadthFirst.py, State.py, View.py) to the Python environment.
5. Run the main.py file to start the game with the specified search algorithm. Experimental Results The game was tested on various maps, and the scores and performance of each search algorithm were observed. Here are the results: • A\* Algorithm: • Map 1: Score = 1000 • Map 2: Score = 1400 • Map 3: Score = 1650 • Map 4: Score = 1750 • Greedy Best First Search: • Map 1: Score = 950 • Map 2: Score = 1200 • Map 3: Score = 1450 • Map 4: Score = 1550 • Breadth-First Search: • Map 1: Score = 900 • Map 2: Score = 1300 • Map 3: Score = 1600 • Map 4: Score = 1650 Conclusion Based on the experimental results, the A\* algorithm outperformed the other two algorithms, achieving the highest scores on all maps. Greedy Best First Search showed decent performance but often resulted in suboptimal paths and lower scores compared to A\* and Breadth-First Search. Breadth-First Search performed reasonably well, especially on maps with complex layouts. Overall, the choice of search algorithm depends on the specific requirements of the game and the desired balance between optimality and computational cost. In this context, A\* appears to be the most suitable choice for the snake game.