Npuzzle Game

DR . IBRAHIM AL HASNONY

Name: عمر نجاح الوردانى غريب النحاس

ID: **1000214371**

Sec: **5**

Introduction:

The Compact N-Puzzle Game is a project aimed at developing a streamlined version of the classic N-Puzzle game, providing a quick and engaging gaming experience for users. The game revolves around rearranging shuffled tiles to form a complete image, challenging players' problem-solving skills and spatial reasoning abilities.

Abstract:

The Compact N-Puzzle Game project aims to create a user-friendly and minimalist rendition of the traditional N-Puzzle game. By focusing on essential features and a simple interface, the game seeks to cater to casual players seeking a quick and enjoyable puzzle-solving experience. The project

involves the development of a compact game with limited complexity while retaining the core essence of the N-Puzzle concept.

Problem Statement:

Traditional implementations of the N-Puzzle game often feature complex interfaces and extensive functionalities, which may overwhelm casual players or those seeking a quick gaming session. There is a need for a more concise and straightforward version of the game that can be easily accessible and enjoyable for users with varying levels of puzzle-solving expertise.

Objective:

The primary objective of the Compact N-Puzzle Game project is to design and develop a compact and intuitive N-Puzzle game that offers a brief yet satisfying gaming experience. The project aims to achieve the following goals:

1. Create a minimalist user interface that is easy to navigate and understand.
2. Implement essential game mechanics for tile movement and puzzle solving.
3. Provide customization options and settings to enhance user experience.
4. Ensure smooth performance and compatibility across different devices and platforms.

Rules:

1. The game consists of a grid of square tiles with one tile missing, arranged randomly at the start.

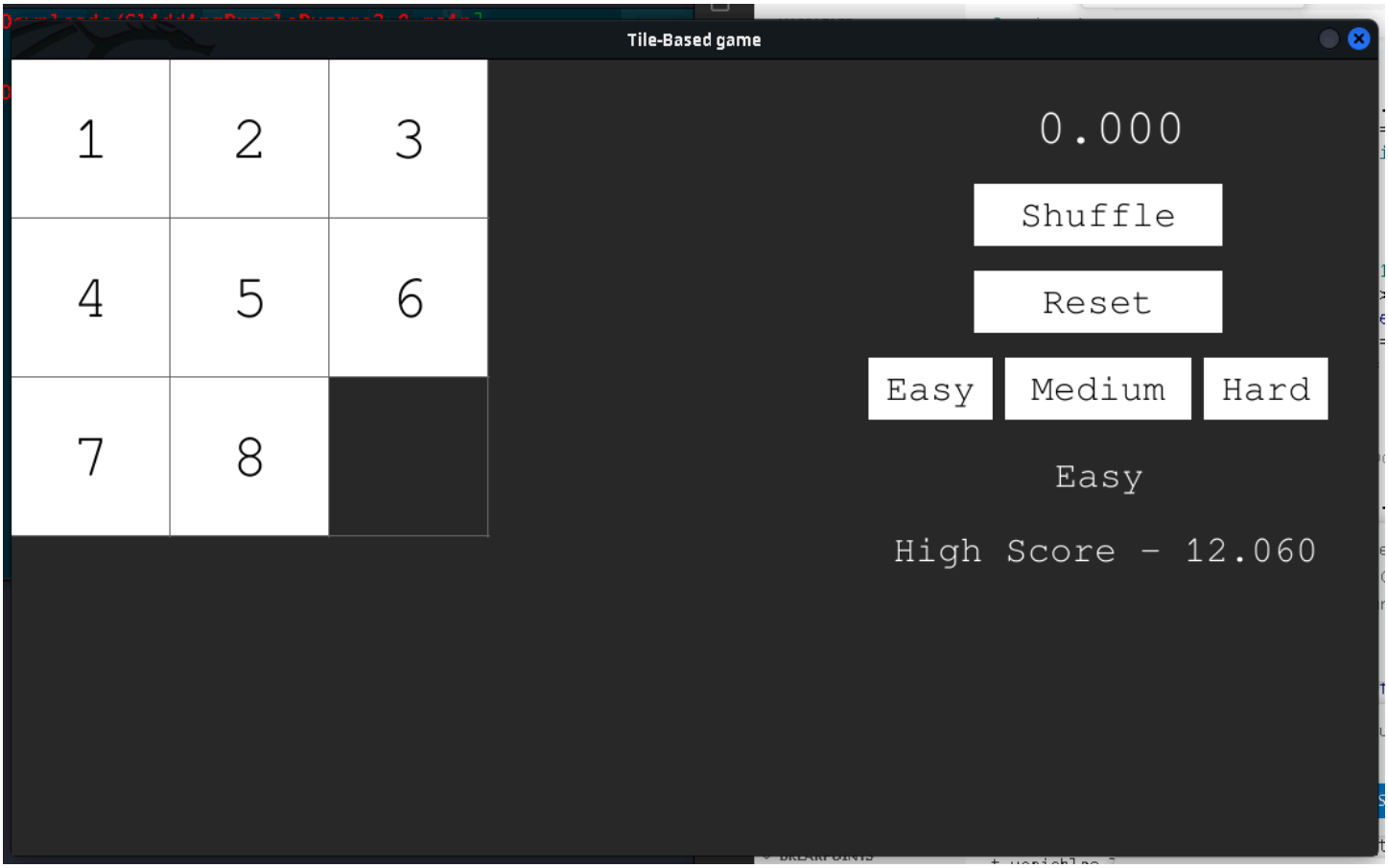
2. Players can move tiles adjacent to the empty space into that space, thereby rearranging the tiles.

3. The objective is to rearrange the tiles to form a complete image by sliding them into the correct order.

4. Tiles can only be moved vertically or horizontally, not diagonally.

5. Players can track the number of moves taken to solve the puzzle and strive to achieve the lowest possible number of moves.

By adhering to these rules, players can engage in a challenging yet accessible puzzle-solving experience, aiming to complete the puzzle with the fewest moves possible.



Implementation:

PASE

1. **Performance (P):** Define performance metrics to evaluate the efficiency of the PEAS agent's puzzle-solving strategies, such as the number of moves taken to complete the puzzle or the time required to generate a solution.
2. **Actuators (A):** Implement actuators to generate solution recommendations, such as suggesting the next move or providing hints to guide players.
3. **Sensors (S):** Develop sensors to perceive the current state of the puzzle, including the positions of puzzle tiles and the empty space.
4. **Environment (E):** Design the puzzle environment to facilitate interaction between the player and the PEAS agent, ensuring seamless integration within the N-Puzzle game interface.

ODESA

- **O:** Optimized Fully observable
- **D:** Digital deterministic
- **E:** Explorer sequential
- **S:** Solver static
- **A:** Agent single agent

program formulation

1. **Initial State:**
   - The initial state represents the starting configuration of the N-Puzzle.
   - It consists of a square grid with dimensions N×NN \times NN×N, where each cell contains a unique number from 1 to N2−1N^2 - 1N2−1 and one empty cell.
   - The empty cell represents the space where adjacent tiles can be moved.
   - The initial state can be represented as a 2D array or a list of lists, where the numbers and the empty cell are arranged according to the puzzle configuration.
2. **Successor Function:**
   - The successor function defines the possible actions or moves that can be applied to transition from one state to another.
   - In the case of the N-Puzzle, the successor function generates valid successor states by moving a tile into the empty cell.
   - Each successor state is obtained by swapping the empty cell with one of its neighboring tiles (up, down, left, or right).
   - The successor function returns a list of valid successor states from the current state.
3. **Goal Test:**
   - The goal test determines whether a given state is a goal state, i.e., whether the puzzle is solved.
   - In the N-Puzzle problem, the goal state is reached when all tiles are arranged in ascending order, with the

empty cell in the bottom-right corner.
    - The goal test checks if the numbers in the grid are sorted in ascending order, with the empty cell in the bottom-right corner.
4. **Path Cost:**
    - The path cost represents the cost or distance associated with transitioning from one state to another.
    - In the context of the N-Puzzle, the path cost can be defined as the number of moves required to reach a particular state from the initial state.
    - Each move (transition from one state to another) has a uniform cost of 1.

Conclusion

Overall, the N-Puzzle problem provides a stimulating platform for honing cognitive abilities, fostering creativity, and enjoying a satisfying puzzle-solving experience. Whether as a casual pastime or a challenging intellectual pursuit, the N-Puzzle continues to captivate players of all ages and interests.