Project Proposal

Even-Odd Sudoku

Instructor

Miss Mehak Mazhar

AI-LAB BCY6A

Usman Islam K224708

Muhammad Bilal K224779

Ismail Ahmed K224792

1. Project Overview

- **Project Topic:** This project is an enhanced version of the classic Sudoku game called "Even-Odd Sudoku." In this version, an additional constraint is added: no two consecutive numbers in any row, column, or 3x3 grid can have the same parity (even or odd). The user interface of the game is developed using the Pygame library in Python.
- **Objective:** The objective of this project is to create an interactive Sudoku game with a new challenging constraint, focusing on logical problem-solving. The project will also develop a graphical user interface (GUI), with puzzle generation, hints, and solving capabilities.

2. Game Description

- **Original Game Background:** Sudoku is a well-known logic-based puzzle game where players fill a 9x9 grid with digits, ensuring that each digit appears exactly once in each row, column, and 3x3 sub-grid.
- **Innovations Introduced:**
    - **Even-Odd Constraints:** In addition to traditional Sudoku rules, no two consecutive numbers within a row, column, or 3x3 grid can have the same parity (even or odd).
    - **Gameplay Impact:** This adds an extra layer of complexity, requiring players to consider parity constraints while solving the puzzle, significantly increasing the difficulty.

3. AI Approach and Methodology

- **AI Techniques:** While the game does not feature AI opponents, the puzzle-solving will be achieved using a backtracking algorithm that is designed to consider both the standard Sudoku rules and the even-odd constraints.
- **Heuristic Design:** The backtracking algorithm will ensure the even-odd constraints are respected while solving the puzzle.

4. Game Rules and Mechanics

- **Modified Rules:** In addition to the usual Sudoku rules, players must ensure that consecutive numbers in any row, column, or 3x3 grid do not share the same parity (either both odd or both even).
- **Winning Conditions:** The player wins once the grid is fully populated with valid numbers that respect both traditional Sudoku and the added parity constraint.

- **Turn Sequence:** Players fill the grid, and the game checks whether the entered number satisfies the even-odd constraint. Hints and automatic solving functionality will also be available.

5. Implementation Plan

- **Programming Language:** Python
- **Libraries and Tools:**
  - **Pygame** for developing the GUI.
  - **Random** for puzzle generation and hint suggestions.
- **Milestones and Timeline:**
  - **Week 1-2:** Finalize game design and rules.
  - **Week 3-4:** Implement puzzle generation, even-odd logic, and backtracking solver.
  - **Week 5-6:** Develop and test the game interface.
  - **Week 7:** Refine game logic, test solving functionality, and add hints.
  - **Week 8:** Final testing and project report preparation.