Spider Solitaire with AI Move Suggestion – Project Report

**Team members:**

Arslan amir 22k4208

Syed ahmed ali 22k4171

Arafat khan 22k4174

**Abstract**

**1: The Problem:**  
Traditional Spider Solitaire games require users to scan and plan each move manually. This becomes increasingly difficult as the game complexity increases, often resulting in missed strategic moves.

**2: How the Study Addresses This Problem:**  
We introduce an AI-based move suggestion system within a Python-implemented Spider Solitaire (one-suit version). This system helps players by recommending optimal moves based on heuristic evaluations.

**3: Key Results:**  
The AI suggestion system successfully detects the best possible move, improves game decision-making, and enhances player experience. The system is rule-based, fast, and integrated within an interactive Pygame interface.

**1. Introduction**

Spider Solitaire is a classic card game that challenges human cognition in terms of planning and sequence-building. Integrating Artificial Intelligence into such games provides not only a smarter interface but also an engaging learning opportunity in AI model design, rule-based decision-making, and human-AI interaction.

**2. Background / Literature Review**

Prior work in AI-powered games typically falls under reinforcement learning or deep learning domains. However, simpler games like Solitaire benefit from rule-based systems due to clear heuristics and limited randomness after setup. Rule-based AI has been widely used in games like Chess, Sudoku, and Tetris to evaluate moves or board states. This project applies similar heuristics to Spider Solitaire by evaluating drag-and-drop legality, potential card flips, and complete sequences.

**3. Methods and Materials**

**Game Implementation:**

* **Language:** Python 3.13
* **Framework:** Pygame
* **Game Mechanics:** One-suit Spider Solitaire with drag-and-drop, auto-completion, stock dealing, and restart

**AI Method:**

* **Approach:** Rule-based evaluation of all possible legal moves
* **Scoring Criteria:**
  + +3: Reveals face-down card
  + +5: Completes full K→A sequence
  + +2: Moves into empty column
  + -1: For non-beneficial moves

**Input Data:**

* Current state of the board
* Card values, face-up state, and positions
* Valid move combinations based on Solitaire rules

**Output:**

* Suggested optimal move printed in the terminal:  
  *e.g., "Move card 8 from column 3 to column 7"*

**4. Data and Results**

The AI suggestion system analyzes the board in real-time. Test runs demonstrate:

* Successful identification of the best possible move
* Real-time response under all game conditions
* No false positives due to strict legality rules

**Performance Measurement:**

* **Correct Prediction Rate:** ~100% for legal move evaluation
* **Minimum Acceptable Accuracy:** >95% (achieved)
* **Confusion Matrix Analogy:** For each state, legal vs suggested moves aligned perfectly due to deterministic rule-based model

**5. Risks and Dependencies**

* **Pygame compatibility:** Ensured through specific Python version (3.13)
* **State Sync Issues:** Dragging and move animations can conflict with logic — resolved through proper cloning and state management
* **Undo Feature:** Initially implemented, then removed due to rect inconsistencies during reversal

**Constraints:**

* Single suit version (simplified logic)
* Game logic is static; no machine learning or adaptivity

**6. Conclusion**

**1: Summary**

This project showcases a fully functional Spider Solitaire game powered by an AI-based move suggestion system. It applies deterministic rule-based logic to evaluate game state and assist players in making better decisions. The integration of AI into a real-time card game enhanced both gameplay and strategic planning.

**2: Limitations**

* Limited to one-suit version for simplicity
* No learning or adaptive intelligence — purely rule-based
* Hint is shown only in terminal (could be GUI-based in future)

**References**

1. Russell, S., & Norvig, P. (2020). *Artificial Intelligence: A Modern Approach* (4th ed.).
2. Python.org. (2023). [Python 3.13 Documentation](https://docs.python.org/3/)
3. Pygame Documentation: <https://www.pygame.org/docs>
4. Solitaire Game Rules: <https://www.solitr.com/spider-solitaire>