SUDOKU PUZZLE SOLVER WITH DYNAMIC GUI VALIDATION

Anwar Basha

INTRODUCTION

- The Sudoku Solver project is a Python application with a Tkinter-based GUI that allows users to input and solve Sudoku puzzles.
- It validates inputs in real-time, ensuring no duplicates in rows, columns, and 3x3 grids, and that all numbers are between 1 and 9.
- Upon valid input, the program solves the puzzle and displays the solution, while invalid inputs trigger an error message.
- Users can solve additional puzzles or exit the application through intuitive buttons.

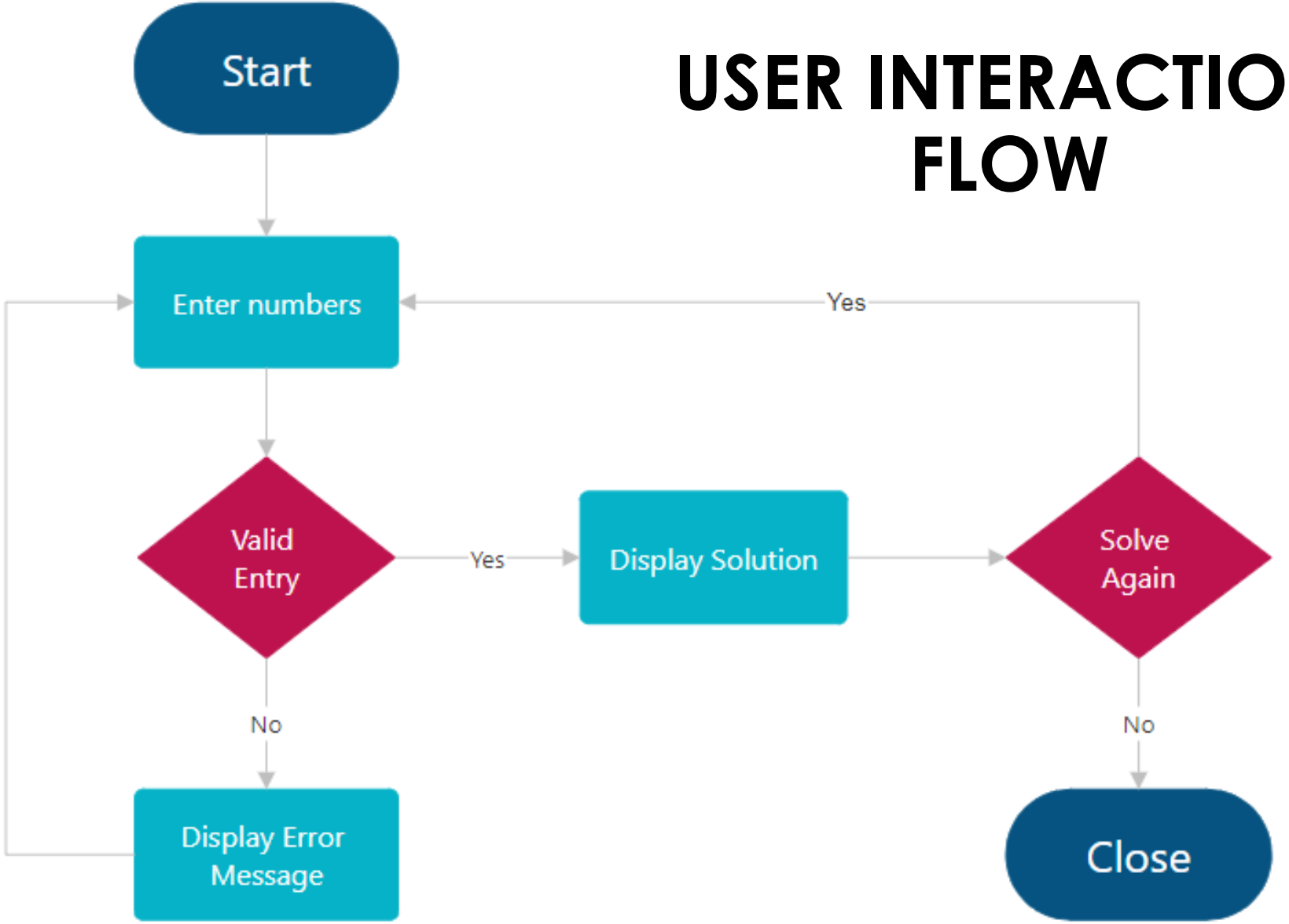
GUI DESIGN AND FEATURES

- Implemented a user-friendly GUI for inputting Sudoku puzzles.
- Program validates input to ensure numbers are between 1-9 without duplicates.
- Displays the Sudoku solution if the input is valid, otherwise shows an error message.
- After solving, the GUI provides options to either close the application or solve a new puzzle.
- Ensured seamless user experience by resetting the process for solving additional puzzles.

VALIDATION CRITERIA

- Numbers are considered invalid if any row, column, or 3x3 grid contains duplicate numbers.
- Valid inputs must be between 1 and 9.
- The input is invalid if it contains any alphabets, special characters, or numbers outside the valid range (1-9).

USER INTERACTION FLOW



CHALLENGES

Data Validation: Ensuring that user input consists of numbers only, to prevent invalid data entry.

Duplicate Detection: Identifying duplicate values in rows, columns, and grids to maintain data integrity.

User Interface: Designing an intuitive user interface using Tkinter to accept user input through text boxes and implementing interactive action buttons.

Solution Display: Displaying multiple solutions effectively using Tkinter labels.

SOLUTION

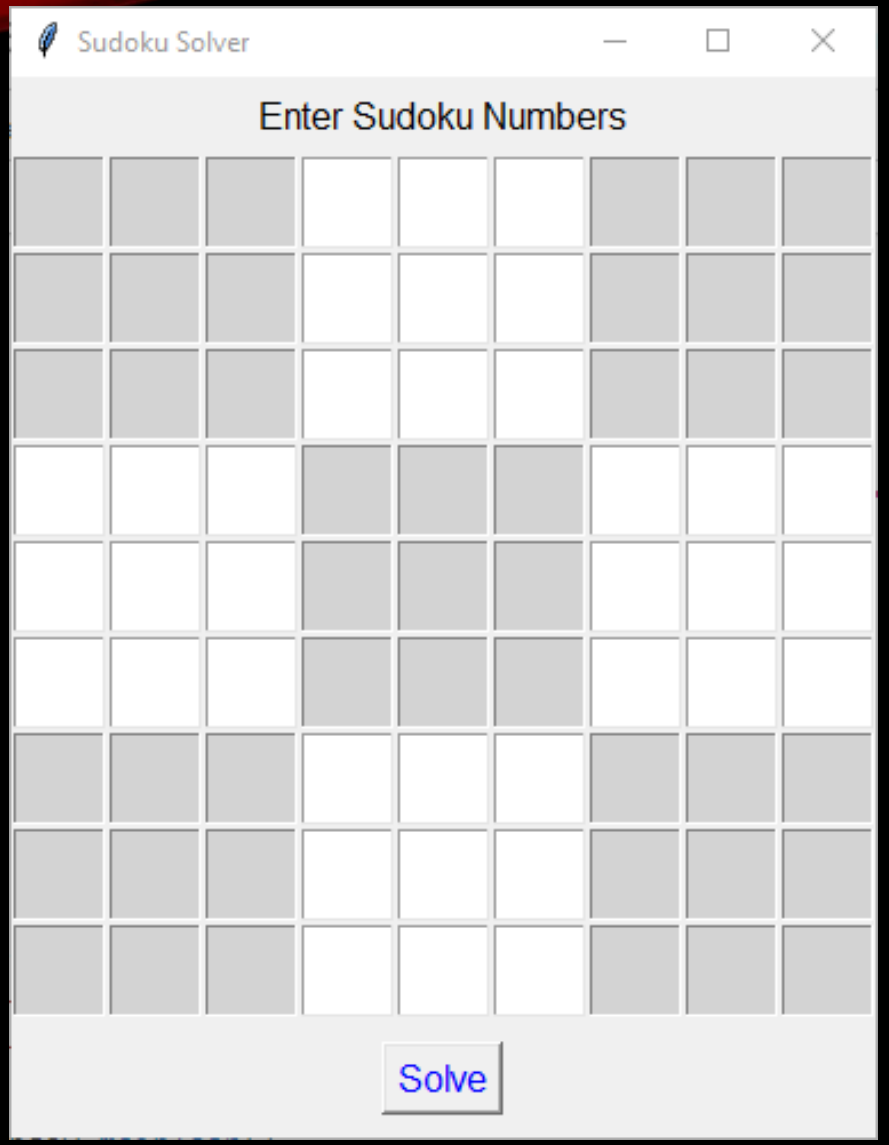
Data Validation: Implemented input validation using python functions to ensure only numerical data is accepted.

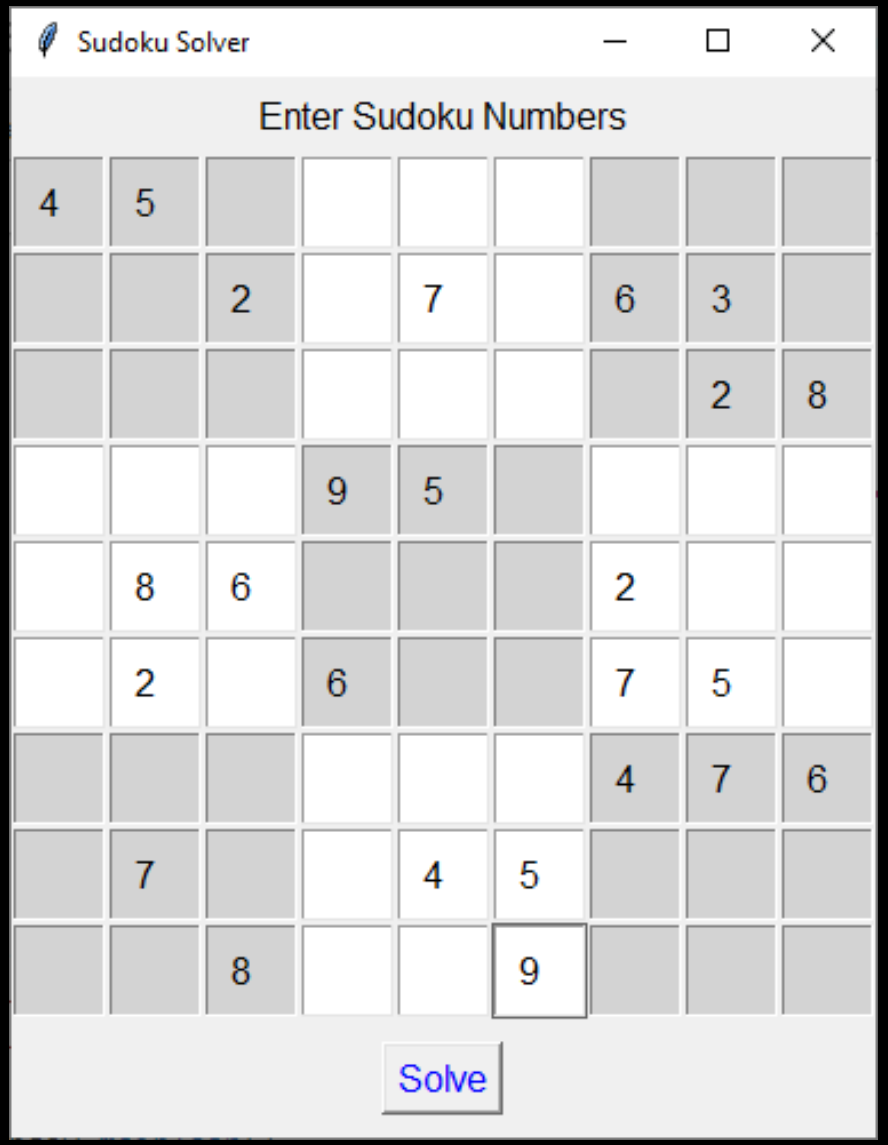
Duplicate Detection: Developed functions to detect duplicates in rows, columns, and grids, utilizing sets and lists to optimize performance.

User Interface: Designed a user-friendly interface with Tkinter, featuring text boxes for input, action buttons for interaction, and labels for displaying solutions.

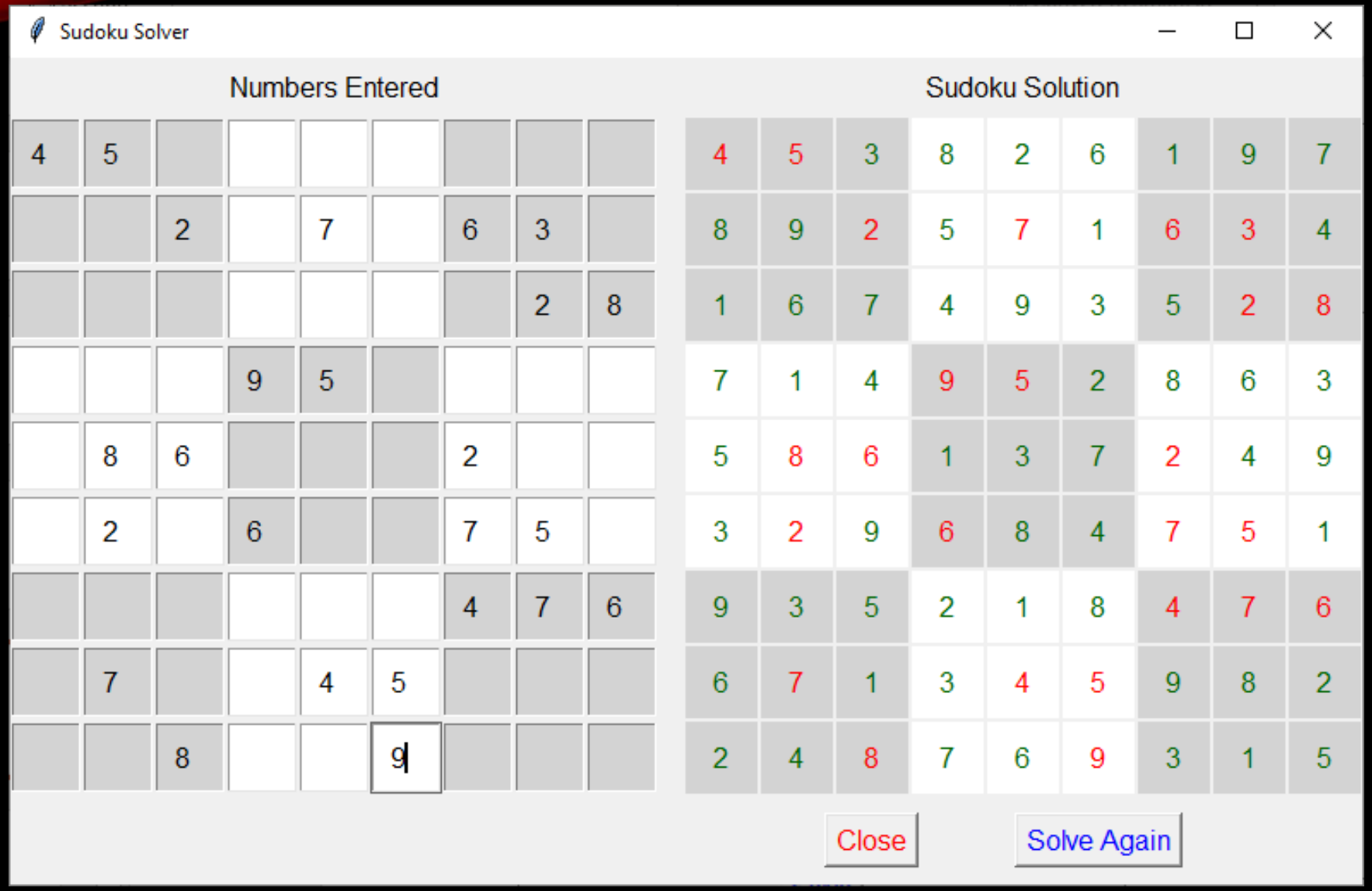
Solution Display: Utilized Tkinter labels to display multiple solutions, ensuring clear and concise presentation of results.

DEMO AND RESULTS 1

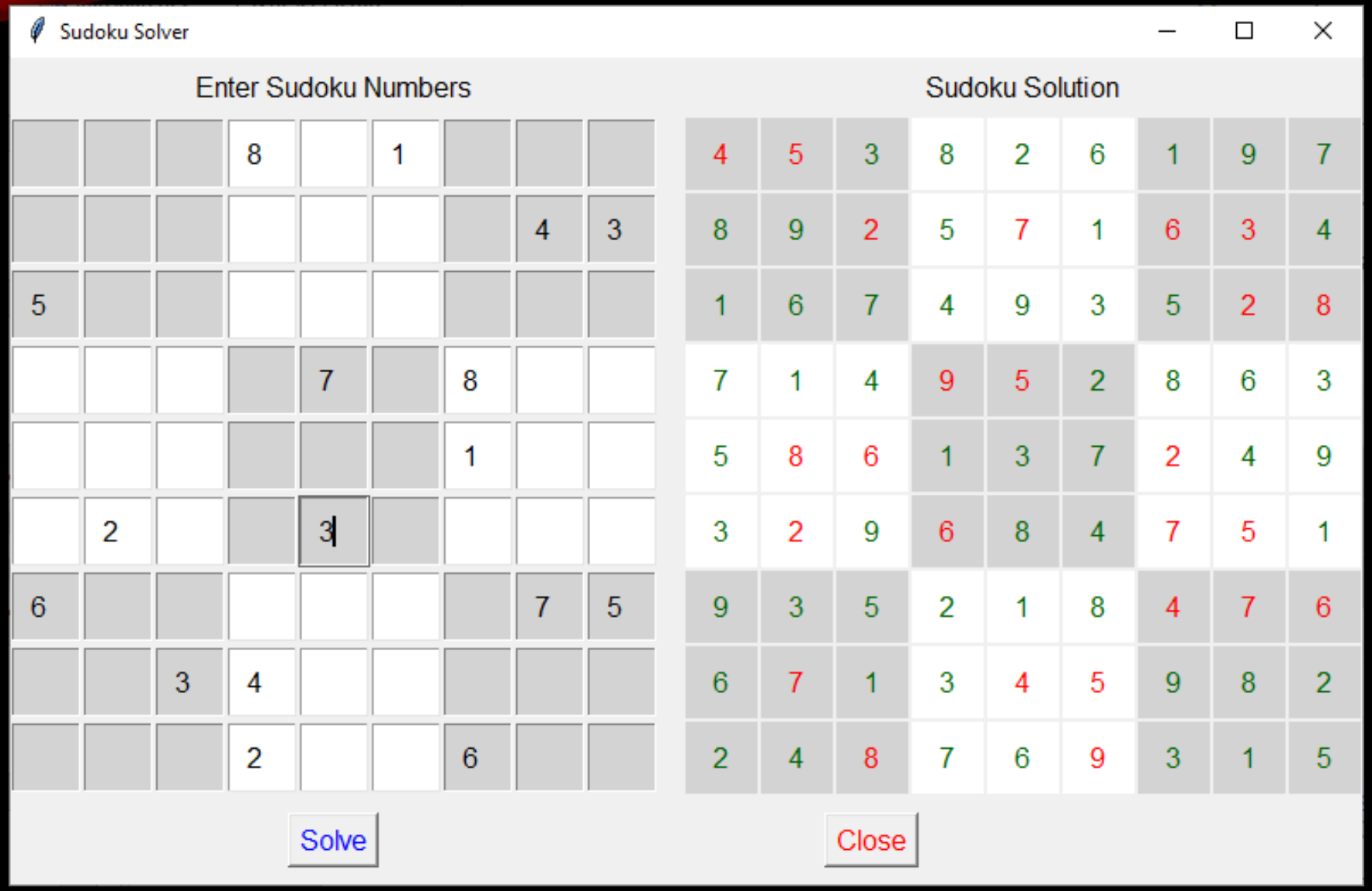




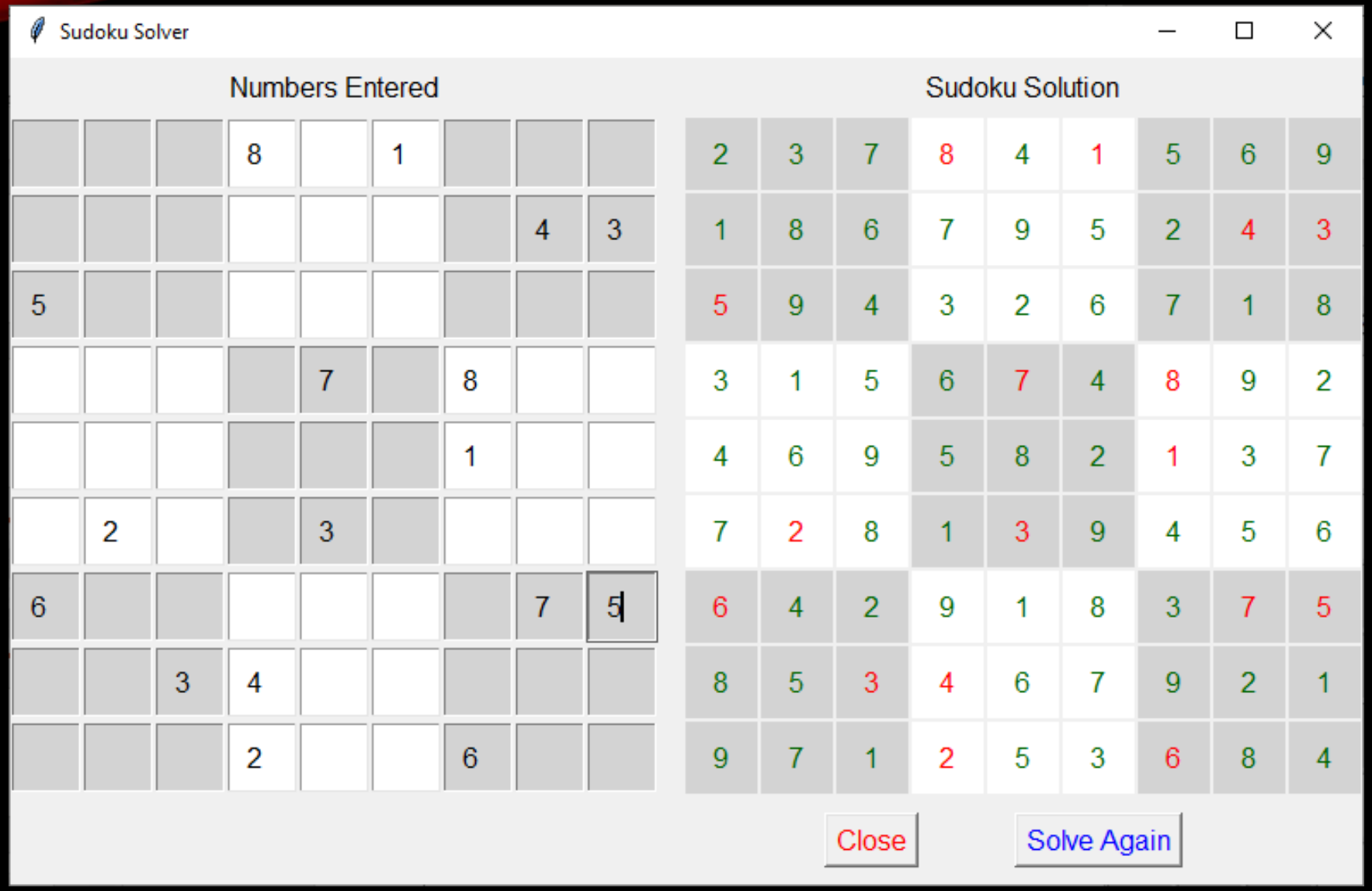
DEMO AND RESULTS 2



DEMO AND RESULTS 3



DEMO AND RESULTS 4



THANK YOU