Programming Sudoku Game

Alperen Koçoğlu

Supervisor
Dr. Mücahit K. Üner
Department of Electrical and Electronics Engineering, Hacettepe University

Introduction

❑ The "Programming Sudoku Game" project aims to create an interactive Sudoku game with Python.

❑ The program has multiple functions like
- Difficulty level selection from easy, medium and hard
- A solution option, if the user wants to see the solution of the puzzle
- A "note" function that lets the user take notes without finalizing their answer
- A clear function that clears the user's input while preserving the current Sudoku puzzle
- A preventive measure that highlights an illegal entry
- An Info page that gives a brief description on the objective of Sudoku and information regarding the purpose of the program.

❑ This Project was coded in Python and the graphical user interface was designed utilizing PyQt5.

Design Process

❑ There were many iterations of this program throughout the design process.

❑ I used Pygame, PyQt5 libraries in different iterations to design the graphical user interface.

❑ I utilized a backtracking algorithm to be able to create the solve functionality. The Initial numbers generated for each puzzle is fetched using a web scraping algorithm. Other than this, every other part of the code is self-sufficient.

Problems Faced and Solution Methodology

❑ There were many problems faced during the development process of the program.
- Design and the development of the graphical user interface (GUI).
- Generating valid Sudoku puzzles of different difficulties.
- Adding preventive measures to keep the user from making illegal moves such as inputting letters instead of numbers 1 through 9 or inputting duplicate numbers that already exist in a certain row, column or within a 3 by 3 matrix.
- Playtesting the program to discover bugs.

❑ For the development of the final GUI design, there were two choices: PyQt5 and Pygame. PyQt5 was chosen because it offered a more efficient way to implement the different labels, button, menus etc. that were in the program.

❑ The generation of Sudoku puzzle numbers was a choice between coding an algorithm that would generate it and using a web scraping algorithm that would get the values from the web.

❑ Adding preventive measures to keep the users from making illegal moves and debugging the features were problems that had to be dealt with. The debugging of features usually takes a long time for any code but for interactive programs like this it takes even longer since it had to be playtested to discover the bugs before being able to start debugging.

Application Areas

❑ This code can be run by any computer that has Python and also be converted into an exe file and distributed that way.
❑ The general application purpose of the project is to play a Sudoku game interactively.

Results and Discussion
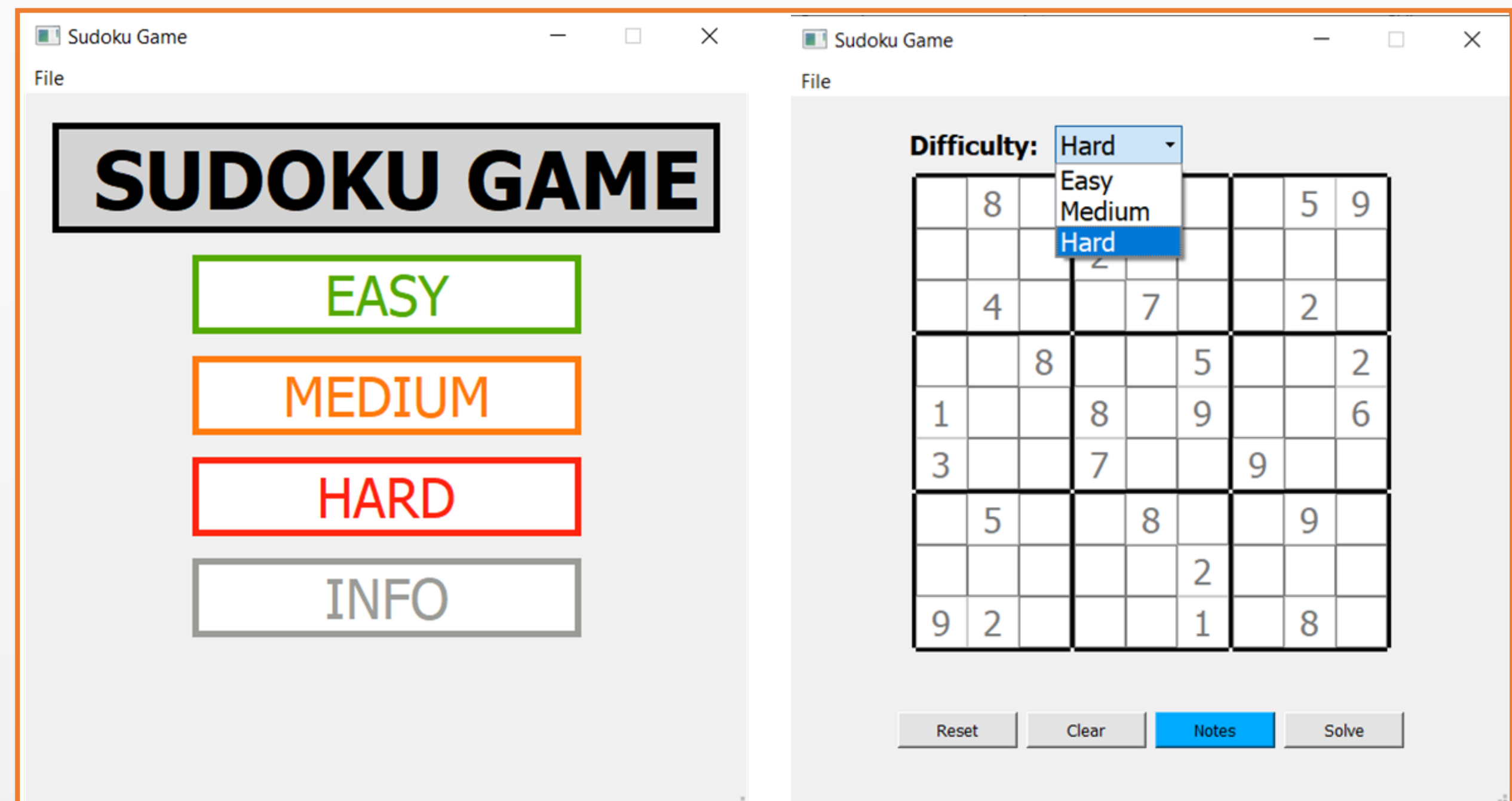
❑ The opening page of the program includes a label that lets the user know what game they're playing, three different buttons ranging from easy to hard levels and an info button that brings the user to another page which includes information about the Sudoku game and the project itself.
❑ Once the user picks a difficulty level, the second interface comes up (in the right picture shown above). This interface features a difficulty selection menu and four different buttons with different functionalities.
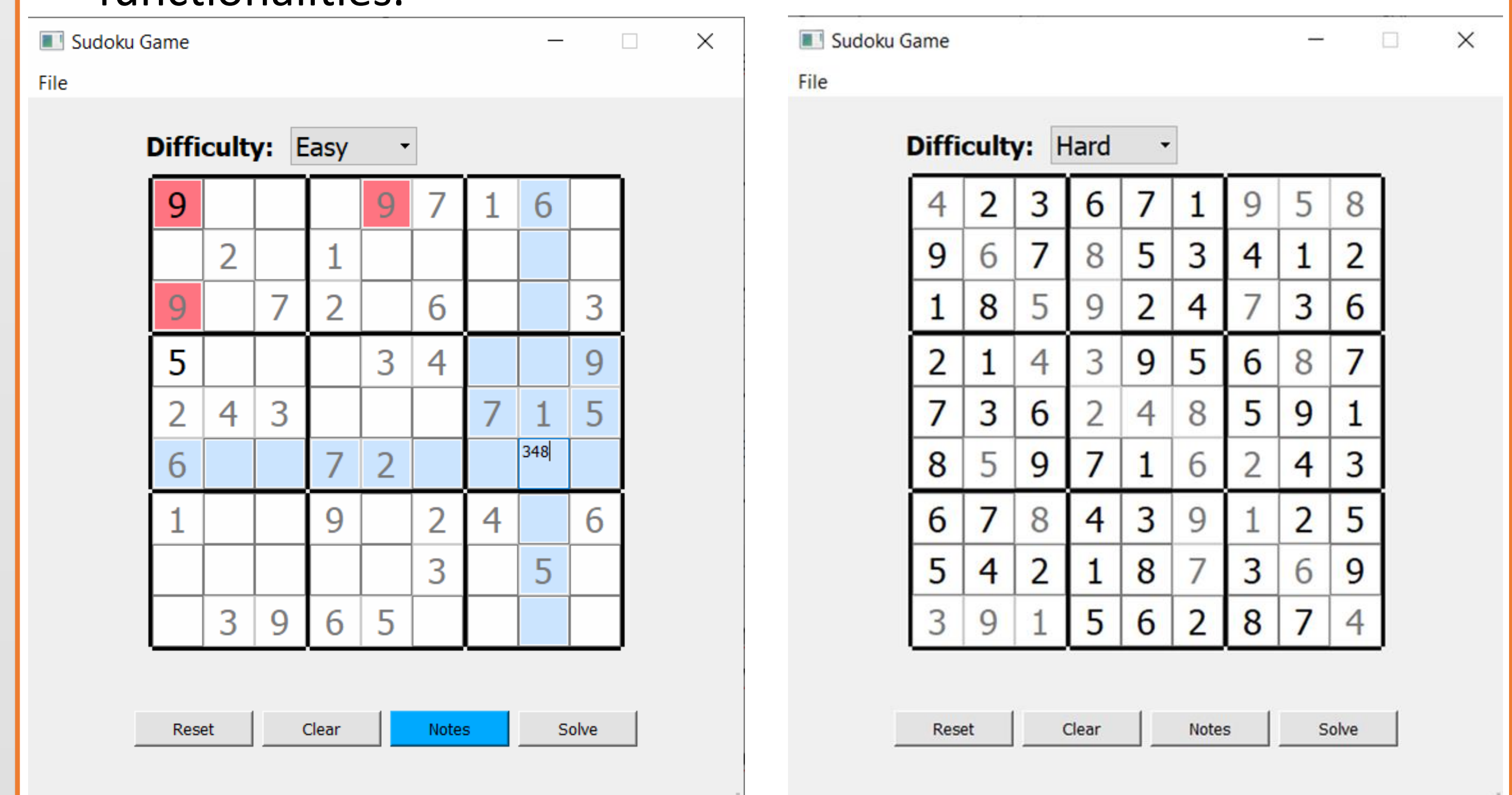
❑ Two demonstration screens with different features of the program are shown above.
❑ On the left picture, in case duplicate inputs happen they're highlighted red. The color difference between the Sudoku numbers and the numbers input by the user is also noticable. Also in the same picture the functionality of the "notes" button is shown.
❑ On the right picture, the outcome of using the "solve" button is shown.

Acknowledgements