Sudoku Solver project proposal- Andrew Biehl and Calypso Leonard

We plan to create a Sudoku puzzle solver, treating the solver as a single player game with an AI which would either solve a given puzzle or help a user to solve a puzzle by telling them if they'd filled a box in incorrectly or by giving them hints when prompted.

Our first step would be a creating a very simple text based solver that takes in a string of numbers and spaces representing the Sudoku board and returns a string with the correctly filled in numbers. This first step would involve having the program check each number with respect to the row, column, and square that it would be in if there were a board, and recording all the numbers which could possibly be entered there. If only one number could be in that position, then that position will be assigned that number. This program will then use recursion to go through the string until there are no blank spaces left. We would also build a graphical interface that takes the string entered into the text solver, displays the unsolved puzzle, and then displays the solved puzzle.

Our next step would be making our text solver more efficient and effective by also checking for numbers that are only possible in one box of a row, column, or square, and adding more interactive features to our interface, allowing the user to fill in numbers on the board and have the solver check them, or asking for a hint and having the solver fill in a random number for them. Once our solver is working consistently, we can also add a program that generates solvable boards.

Our stretch goal would be to develop our solver further, so that it can be more and more flexible, this could involve making it more efficient so it can solve large boards, or could involve it handling letters as well as numbers in the input string, or it could involve generating and solving some of the less traditional boards. Which of these options we go with depends on where we think our program needs the most development once we’ve got everything running.