The Sudoku Project
WEEK 2: 09/08/2021 to 15/08/2021

Ishika De and Yashvi Donga

August 2021

Agenda

- Brief Overview
- Current Status
- Toolchain
- Difficulties
- Learnings

Brief Overview

The goal of this project is to investigate a variety of algorithms (backtracking, brute force, stochastic search, depth first search) that are capable of solving standard Sudoku puzzles, of ranging difficulties, in order to learn more about Sudoku solving techniques.

We also wanted to create the sudoku solver using OpenCV that will read a puzzle from an image and solve it. We plan on using OpenCV for multiple programming languages.

Current Status

- Implemented Brute Force Algorithm to solve a Sudoku in Java and Python.
- Implementing Stochastic Simulated Annealing Algorithm in Python. We are very close to completing it.
- Reasearched about OpenCV library of multiple languages like Java, C++ and Haskell.

Toolchain

- Languages: Python, Haskell, Elixir, C++, Java.
- Open CV - Possible in Python, C++ and Java.

Difficulties

- The backtracking algorithm took some time to implement because we had few challenges implementing the recursive function.
- Removing Spaces from the Solved Sudoku for Generating Sudoku is a challenge.
- Implementing Brute Force in Java needed extra conditions to give the correct output.
- Understanding Stochastic Simulated Annealing was difficult. The resources for this algorithm were quite less.
- Explaning Algorithms to each other was also a challenge.

Learnings

- Learning about the backtracking Algorithm - Generating and Solving a Sudoku in Python.
- Collaboration and understanding git commands, discovered about VSCode live share.
- Implementing the backtracking Algorithm in Java and C++.
- Explaining our code, thought process and ideas to each other.
- Implementing Brute Force Algorithm in Java and Python.
- Understanding the concepts implemented in Stochastic Simulated Annealing.