

# The Sudoku Project

## WEEK 4: 16/08/2021 to 21/08/2021

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 Backtracking and Brute Force Algorithm to solve a Sudoku in C++, Java and Python.
- Tried implementing Stochastic Simulated Annealing Algorithm in Python. (Not working for all cases)
- Tried implementing Crook's Algorithm in Python. (Not working for all cases)
- Started writing the code to solve sudoku in Haskell.
- Researched about OpenCV library of multiple languages.

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

- The backtracking algorithm took some time to implement because we had few challenges implementing the recursive function.
- Understanding Stochastic Simulated Annealing was difficult. The resources for this algorithm were quite less.
- For stochastic and crook's algorithm approach, the sudoku solver does not work for all cases. This requires a little bit of work.
- Explaining each other's code to each other took a while.

- Learning about the implementation of different algorithms to solve a Sudoku.
- Collaboration and understanding git commands.
- Explaining our code, thought process and ideas to each other.
- Understanding the concepts implemented in Stochastic algorithm using Simulated Annealing.