Yossatorn Phithakjiraroj 6280942
Kanokluk Supthamrong 6280563

Project Proposal: Parallel Sudoku Generator

Overview:

Sudoku solving is a time-consuming process, particularly for larger puzzles. As the puzzle size and difficulty increase, the time required to solve them grows exponentially, making it impractical for many applications. To reduce the time required to solve complex puzzles, we propose to apply parallelism to the Sudoku generator.

Scope:

We will divide the puzzle into smaller sub-puzzles and use Rust programming language to implement the parallel Sudoku generator. The parallel approach can be used for generators and solvers. We will measure the efficiency of our parallel Sudoku generator using performance metrics such as time taken, CPU cores used, and speedup achieved. We will also validate the quality of the generated puzzles to ensure they have unique solutions and are solvable by backtracking algorithm. The graphical user interface will allow users to customize puzzle size and difficulty, generate random puzzles, or input their own puzzles. Automated tests and manual testing will validate the correctness of our algorithm and user interface's functionality.

Conclusion:

Our project aims to apply parallelism to the Sudoku generator to reduce the time of solving large and complex puzzles. The parallel approach can reduce the time of solving complex Sudoku puzzles and can also be used for generators and solvers. This project has the potential to improve the performance of Sudoku generators and solvers and has many practical applications.