KKJT > ∞

Sudoku Solver & Generator

KATHLEEN NEAR
TANYA OLIVAS
JARED PETERSON
KURTLYND MCLANE

Sudoku Solver & Generator

- Generates a Sudoku puzzle to be solved
- Puzzle follows standard rules – numbers one to nine in every row, column, and section without duplicating
- Option to solve puzzle for current puzzle or custom made
- Option to increase or decrease difficulty (easy, medium, hard)
- Option to shrink from standard 9x9 puzzle to 4x4 puzzle

Deciding Architecture

- Basic program with no communication between clients
- Heavily based on user input to produce final product
- Graphical user interface provides easy accessibility
- Program algorithm filters numerous amounts of input

Design Choices

- Pipes-and-Filter
  - Input from user processed by algorithm for comparison against answers and then feedback
- Layering
  - Program designed in levels where GUI module accesses main algorithm and other modules

Decision to Use Layering

- Our program naturally utilizes a layered structure; modules neighbor each other and use each others’ services.
- Layering allows for our program’s modules to communicate effectively, back and fourth.
- This style allows us to easily make more modifications – something necessary due to the need for low dependency between our modules.

Summary

- Our Sudoku Solver & Generator generates and solves custom and pre-made puzzles with many options to improve the gaming experience.
- Driving factors to consider for architectural design included a GUI interacting with an algorithm and frequent input provided by the user
- Pipe-and-Filter and Layering were the two styles we considered.
- Our program naturally works with Layering in addition to the styles' own practicality for this project