ABSTRACT

Aristianto (08220030032)

Solving Logic Grid Puzzle Using
Recursive Constraint Satisfaction Algorithm

Logic puzzle is a kind of problem that gives us a set of premises and then we are asked to draw a conclusion using logical deduction or reasoning. Some types of logic puzzles include Knights and Knaves, Sudoku, Nonogram, and "logic grid" puzzle. If the puzzle is simple and contains only a small number of premises, it can be solved manually in a short time. But if the puzzle is complex and contains many premises we will need a systematic approach.

This final project proposes a systematic approach to solve "logic grid" puzzles. This approach can be used to solve all "logic grid" puzzles. The puzzle is translated into a specification language which will be processed by the software to generate a solution. To convert the specification into a solution we will use recursive function calling. How we specify the puzzle will affect the performance of the program.

The aim of this final project is to create a specification language that is easy to understand and flexible enough to accommodate all "logic grid" puzzles. An interpreter to process the language is also included. It does not deal with any optimization issues, complicated syntax checking, etc. More features can be added to the language in the future should the need arise. The name of the specification language is FasyLogic.