

LonPro Solver

Abstract:

The LonPro solver is a program designed to solve puzzles presented in the LonPro board game. LonPro is the Taiwanese version of the game Quadrillion. Quadrillion is a puzzle game where players are given several different puzzle pieces of different shapes and colors. These puzzle pieces have to all fit onto a game board in order for the player to win.

Introduction:

The original idea of this program was for it to be a Quadrillion game solver. The Quadrillion game presents users with different puzzles and then provides them with the answers, similar to a Sudoku booklet or a word search puzzle where the answers can be found in the book. I am more familiar with the Taiwanese version of the game. I would play it when I was younger. LonPro has a different board which is triangle shaped. As far as the puzzles go, LonPro has a somewhat different approach and starts off with simple puzzles that have answers that the player can look up, but then as the player progresses through the booklet the puzzles get more and more difficult. LonPro has a last section of high level puzzles and does not give the player solutions. The LonPro Research Working House even held contests in the past to see if people could figure out these puzzles and submit the answers to the company for chances to win various prizes and badges. While this contest is no longer still going on, I thought it would be interesting to write a program that can solve these higher level puzzles. This document describes the layout of the program and how it works.

Detailed System Description:

The system works by printing the board and then filling it with the different puzzle pieces until they are all put in successfully. Matrices are used for the board, game pieces, and puzzle inputs. The system then uses a series of loops to try to put all the pieces in using brute force. The pieces are tried through different orientations and positions until a fit is found. This program follows the same method that a person would use when trying to solve the puzzle, only the program can do it a lot faster.

Two classes, GameBoard and GamePiece will be defined as illustrated in the UML below.

UML:

GameBoard
+GameBoard() +draw(): void

Note: GameBoard is a set size, so it does not have any data fields.

GamePiece
+id: int +name: String +color: String +possibleOrientation: int +position: int

The initial version of this program have the puzzle input built-in as part of the code.

In the final version, the puzzles will be in separate input files to the program that can be run from the terminal.

Requirements:

Puzzles are inputted as matrices. The LonPro puzzle game package did not provide solution to players. This LonPro program can solve the puzzle quickly and reliably.

Literature Survey:

User Manual:

In the current version of the LonPro solver, the puzzles are inputted into the code and then the code is run to solve the puzzles. In the final version, the puzzles will be in separate files that can be run from the terminal.

Conclusion:

Original proposal stated that mouse events were going to be used to input the puzzles, but due to time constraints this is no longer part of the program. Instead the puzzles are in separate files.

References/Bibliography:

LonPro User Manual by LonPro Research Working House 2002