Discrete Mathematics

Homework 1

One More Puzzle with SMT Solver

but not with your team members this time

Shin Hong

3 Oct, 2019

Assignment Overview

- Write a C program that finds a solution of a given game of the Number Cross puzzle using the Z3 SMT solver
- Submission
  - deadline: 9 Oct (Wed), 11:59 PM
  - deliverables
    - program (C source code files, build script, README)
    - write up (1 page)
- You are not allowed to collaborate with any one in or out of the class in doing this homework
  - it is allowed to reuse the PA 1 results of your team

Number Cross

- Number Cross consists of a $M \times N$ grid where every square, every row, and every column and is labeled with a positive integer

- The goal of the game is to determine the color of each square as either Black or White, such that
  - the label (number) of each column is the same as the sum of the numbers in the Black squares of the column
  - the label of each row is the same as the sum of the numbers in the White squares of the column
  - the number of a square label is greater than or equal to 1, and not greater than 9
  - initially, no information on the coloring is given

37	17	16	16	18	25	17	12	30	
8	2	1	8	1	3	5	7	6	18
9	1	4	2	5	6	3	1	7	28
3	5	1	4	9	1	3	9	1	8
8	6	6	3	5	1	1	4	1	4
8	6	6	2	6	8	3	3	9	31
8	7	8	8	4	5	2	1	1	18
4	8	3	5	5	2	1	2	8	24
1	2	8	8	8	3	7	2	7	22
8	3	9	5	9	2	1	4	9	35

37	17	16	16	18	25	17	12	30	
8	2	1	8	1	3	5	7	6	18
9	1	4	2	5	6	3	1	7	28
3	5	1	4	9	1	3	9	1	8
8	6	6	3	5	1	1	4	1	4
8	6	6	2	6	8	3	3	9	31
8	7	8	8	4	5	2	1	1	18
4	8	3	5	5	2	1	2	8	24
1	2	8	8	8	3	7	2	7	22
8	3	9	5	9	2	1	4	9	35

Requirements

- Your program must use the Quantifier-free LIA logic to model this game (not propositional logic)
- Input
  - input is given via the standard input
  - the first line consists of *N* tokens which are a sequence of the column labels
  - each of the following *M* lines has *N* + 1 tokens; first *N* tokens are a sequence of the square labels in the row, and the last label for the row label
  - M is not greater than 50, and so is N.
- Output
  - print the coloring of the grid to the standard output
    - 1 : Black
    - 0 : White
  - print "No solution" if there's no solution

```
37 17 16 16 18 25 17 12 30
8 2 1 8 1 3 5 7 6 18
9 1 4 2 5 6 3 1 7 28
3 5 1 4 9 1 3 9 1 8
8 6 6 3 5 1 1 4 1 4
8 6 6 2 6 8 3 3 9 31
8 7 8 8 4 5 2 1 1 18
4 8 3 5 5 2 1 2 8 24
1 2 8 8 8 3 7 2 7 22
8 3 9 5 9 2 1 4 9 35
```

<Input example>

```
1 1 0 1 0 0 1 0 0
0 1 1 0 1 0 0 0 0
1 1 1 0 1 0 0 1 1
1 1 1 1 1 1 1 0 1
0 0 0 1 1 0 0 1 1
0 1 1 1 0 0 1 1 0
0 0 1 0 0 0 1 1 1
1 1 1 1 0 1 0 1 0
0 0 0 0 1 1 0 1 0
```

<Output example>

Program Structure

- Write a C program running on UNIX/LINUX
  - Each program receives input from the standard input and produces the ouput to the standard output
  - Tests will be conducted on Peace
- The program must be built as a single executable
  - Programs can execute Z3 in a middle of execution through `popen` (see an example of `nqueen-LIA.c`)
- You must submit a build script and README together with source code files
  - buid script: Bash script, Makefile, Ant, Maven, etc.
  - REAME: instruction/manual on how to build and run your program

Submission

- Deadline: 9 Oct (Wed), 11:59 PM
  - no late submission will be accepted

- Submit the program and one write-up (report)
  - Program: source code files, build script and README
  - Write up: must not exceed 1 page (in A4; either in single or double column)

- Submit all deliverables via Hisnet homework submission repository

- Your submissions may be open to the class and public

Evalution Criteria

- Write up (60 points)
  - Description (50 points)
    - check whether you found all constraints of a solution
    - check whether each constraint is correctly represented as a logic formula
    - check whether you demonstrate the correctness of your programs in a convincing way (e.g., by tests)
    - check whether all descriptions are clear and consistent
  - Discussion (10 points)
    - detailed analysis of results, interesting observations, lessons learned, suggestions, new ideas, etc.

- Tests (40 points)
  - Run each program with several inputs to see whether the results are correct
  - Note that tests will be conducted on the Peace server