

Homework 1

CS 199-173
September 14, 2021

Due: September 27

In this exercise, we will use Z3 to solve and then modify this logic puzzle: <https://primepuzzles.wordpress.com/2016/08/11/puzzle-40/> (Note: if there is a different puzzle you would like to work with instead, contact me and I will probably approve it if it is of roughly similar complexity and not a Sudoku.)

0. Watch the video introductions to Z3 (https://mediaspace.illinois.edu/media/t/1_4rdpxom/229438603) and to Skyscrapers puzzles (https://mediaspace.illinois.edu/media/t/1_vzw8m2rf/229438603)
1. We are going to reduce the puzzle to a satisfiability problem and solve it using Z3. Basically, our existential claim will be “This puzzle is solvable”, and then the “model” that Z3 produces to prove the claim will be the solution to the puzzle. To do this, we need to represent the puzzle as a set of variables and constraints.
 - (a) Create 16 variables representing the values of the 16 cells. Each variable should be named x_{ij} where i is rows from the top and j is columns from the left, e.g. x_{43} is the value for the cell containing an S. Create 4 additional variables representing the values of the four letters.
Note: copy/paste and find/replace can greatly speed up the task of writing 16+ very similar lines, both here and later in the assignment.
Note: If you have some significantly different scheme for modeling the problem that does not involve these 20 variables, contact me and I will probably allow it. But if you do want to follow my 20-variable scheme, then use my naming convention as well.
 - (b) Write a function named “clue” which takes in four numbers (representing the contents of a row or column), and returns the appropriate Skyscrapers clue for that row/column. For example, (clue 2 1 3 4) should equal 3.
 - (c) Using your “clue” function when necessary, write constraints encoding the ciphered Skyscraper rules and the givens of this puzzle, i.e. all the constraints that must be true for the 16 cell variables to form a valid solution.
 - (d) Get results from Z3. Check your work by confirming that the values Z3 assigns to the variables are indeed a valid solution to the puzzle.
2. Next we are going to use Z3 to make a harder version of the puzzle. Removing a given clue makes the puzzle harder (because now the person solving it has less information to work with), *but* removing a given may also make it possible for there to be multiple solutions, in which case the whole puzzle is considered invalid. So first we need a way to ensure there is only one solution.
 - (a) Add one more constraint representing that the solution must *not* be the one Z3 output for part 1. (*This will be a very large constraint - tedious to write by*

Homework 1

CS 199-173
September 14, 2021

hand, but fast if you use take the model from the previous step and edit it using find/replace.) When you run the code it should now say “unsat”, since in the original puzzle there aren’t any other solutions.

- (b) Comment out constraints to discover at least two of the 11 givens that can be safely simultaneously removed from the puzzle. *(It is actually possible to remove five at once! But you don’t need to find all five. If Z3 finds a solution to your reduced set of constraints, you need to pick different givens to remove, because your current version of the puzzle has multiple solutions: the one newly found, and the one being banned by part (a). If you’ve removed all instances of a letter from the puzzle, also remove it from your uniqueness check in part (a) - otherwise there may be many “distinct” solutions which differ only by what value they assign to the unused letter.)*

3. What to turn in (I will provide a place to turn these in in a few days):

- A text file named “part1.smt2” containing your answer to part 1. If I run that code, I should see the word “sat” followed by a model representing a solution to the original puzzle.
- A text file named “part2.smt2” containing your answer to part 2: it is the same as the first file except that it has one extra constraint banning the original solution (part 2a), and has at least two of the old constraints commented out, along with a comment as the first line of the file explaining which of the 11 givens you removed (part 2b). If I run that code, I should see the word “unsat”, followed by an error saying that no model is available.