Constraint Solving | SS 2024 | LVA 703304

Test-Exam | June 21, 2024

> This is an old exam that consisted of five exercises. The available points for each item are written in the margin. You needed at least 50 points to pass. Exercise 3(b) was a **bonus** exercise. Explain your answers!
> **Exercises 1 and 2(b) cannot be solved with knowledge of this course, as these topics have not been treated!**

[5] $\boxed{1}$ (a) ~~Compute the ZDD representation B of the boolean function $f(a,b,c) = ab + \bar{a}c$ with variable ordering $[a,b,c]$.~~

[5] (b) ~~What is $\mathsf{count}(B)$ for the ZDD B of part (a)?~~

[10] (c) ~~List all binary boolean functions that have no $\boxed{0}$ node in their ZDD representation.~~

[10] $\boxed{2}$ (a) Use Cooper's method to transform the QLIA formula

$$\neg\forall x.\,(3y - 1 \geqslant 3x \;\vee\; y = 2x - 6)$$

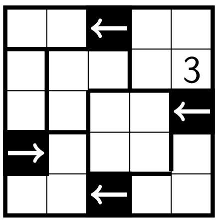
into an equivalent quantifier-free formula.

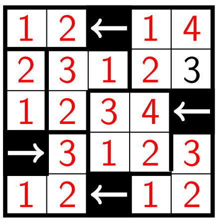
[10] (b) ~~Find all solutions of the (integer) divisibility constraints $4 \mid 3a + 2b \wedge 2 \mid a - b + 1$ with $0 < a, b < 5$.~~

$\boxed{3}$ In this exercise we consider the logic puzzle *Makaro* published by Nikoli. The grid is divided into rooms and the objective is to fill the empty cells in the rooms with numbers, subject to the following constraints:

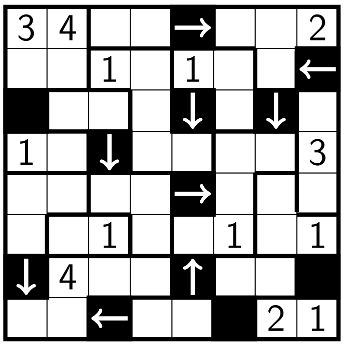
- A room with n cells is filled with the numbers 1 to n.
- Neighbouring cells in adjacent rooms may not have the same number.
- The number in a cell pointed to by an arrow must be larger than the number in every other neighbouring cell of the arrow.

For example, the puzzle on the left has the (unique) solution on the right:





[20] (a) Construct a formula φ that is satisfiable if and only if the puzzle



has a solution. Specify the underlying (SMT) theory and provide sufficiently many details (allowing the instructor to construct the full encoding).

[10] (b) Solve the puzzle of part (a). (This is a bonus exercise.)

4 Linear Arithmetic

[11] (a) Consider the following constraints:

$$2y \geq -4x - 3 \tag{1}$$
$$4y + 8x \leq -5 \tag{2}$$

Apply the simplex algorithm to find a rational solution to the constraints. Here, you should use s as name of the slack variable that is introduced for constraint (1) and t as name of the slack variable for (2). Use Bland's selection rule with the variable order $x < y < s < t$. No preprocessing is allowed (e.g., one that would completely eliminate one of the four variables). Provide the initial tableau and bounds as well as intermediate results after each pivoting step and after each update of the assignment.

[9] (b) The Bellman–Ford algorithm from the lecture works on weighted graphs $G = (V, E, w)$. It contains a loop with $|V| - 1$ iterations of distance-updates, where $|V|$ is the number of nodes (not counting the fresh starting node s). Consider a modified algorithm with only $|V| - 2$ iterations. Figure out which of the following three problematic situations can occur. For each situation either provide a concrete witness with four nodes ($V = \{a, b, c, d\}$, just the graph suffices), or provide a brief justification why the situation cannot arise.

(1) It can happen that G does not contain a negative cycle, but the modified algorithm reports a negative cycle.

(2) It can happen that G does not contain a negative cycle, the modified algorithm returns a distance array, but at least some computed distance is not correct.

(3) It can happen that G contains a negative cycle, but the modified algorithm returns a distance array.

5 Arrays

Consider the following program to compute Fibonacci numbers for arbitrary $N \geq 0$:

```
int a[N+1];    // entries a[0], ..., a[N]
a[0] = 0;
a[1] = 1;
int i = 1;
while (i < N) {
  a[i+1] = a[i] + a[i-1];
  i = i+1;
}
return a[N];
```

[4] (a) Think of a suitable invariant which permits to prove that all array accesses within the loop-body are within bounds. Write down the invariant and all formulas that have to be validated.

[10] (b) In order to prove soundness of the program, the following formula $\varphi(a, i)$ might serve as an invariant:

$$\varphi(a, i) := (\forall k.\ 0 \leq k \leq i \longrightarrow a[k] = \mathit{fib}(k))$$

Then the formula ψ expresses that the invariant is maintained after a loop iteration:

$$\psi := \varphi(a, i) \wedge i < N \wedge b = a\{i + 1 \leftarrow a[i] + a[i - 1]\} \wedge j = i + 1 \longrightarrow \varphi(b, j)$$

Transform the negated formula $\neg\psi$ into an equisatisfiable quantifier-free formula χ which does not contain any array-operations. Provide intermediate formulas.

[6] (c) Try to prove unsatisfiability of χ. To this end you can of course use the specification of Fibonacci-numbers, in particular the equation:

$$\forall n.\ n \geq 1 \longrightarrow \mathit{fib}(n + 1) = \mathit{fib}(n) + \mathit{fib}(n - 1)$$

Either complete the proof or indicate why your attempt got stuck and how the invariant might be adapted.