## *Logical reasoning and programming*, lab session 5

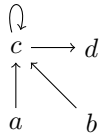
**(October 23, 2023)**

The following exercises require an SMT solver. For simplicity, you can use

- an online version of Z3, or
- an online version of CVC5[1],

or both. Even better, you can install Z3 or CVC5 yourself. Another option is to use pySMT, a convenient way how to experiment with various SMT solvers in Python. If you want to learn a bit more about the Z3 prover, you should start with this Z3 Guide. Moreover, if you want to play with the Z3 prover in Python, check Z3 API in Python and Programming Z3. However, if you want to experiment with various SMT solvers in Python, you should try pySMT.

**5.1** We have a language that contains only one binary predicate symbol $\in$ and we have an interpretation $\mathcal{M} = (D, i)$ such that $D = \{a, b, c, d\}$ and $i(\in)$ is given by the following diagram:




Meaning that $x \in y$ iff there is an arrow from $x$ to $y$. Decide whether the following formulae are valid in $\mathcal{M}$:

(a) $\exists X \forall Y(\neg(Y \in X))$,

(b) $\exists X \forall Y(Y \in X)$,

(c) $\exists X \forall Y(Y \in X \leftrightarrow Y \in Y)$,

(d) $\exists X \forall Y(Y \in X \leftrightarrow \neg(Y \in Y))$.

**5.2** Decide whether it is satisfiable in the theory of uninterpreted functions that

$$x = f(f(f(f(x)))) \wedge x = f(f(f(x))) \wedge x \neq f(x).$$

**5.3** Is it possible to decide whether $\forall X(f(f(X)) = g(X)) \wedge f(g(a)) \neq g(f(a))$ is satisfiable by our congruence closure algorithm?

**5.4** Prove that the algorithm to extract a solution for Difference logic, if there is no cycle in the graph, always works.

**5.5** Try all the examples in the SMT-LIB Examples.

**5.6** Show that $x - y > 0$ iff $x > y$ holds for integers, but does not hold for bit-vectors with a fixed length.

**5.7** Let $x$ be a 32 bit-vector. You want to verify that if you produce $y$ by $x \gg_s 31$ (arithmetic right shift is `bvashr`) followed by one of the following

[1] There is also an older version called CVC4 available.

- $(x \oplus y) - y$, or
- $(x + y) \oplus y$, or
- $x - ((x + x)\&y)$,

then you get the absolute value of $x$. For further details, check this webpage.

**5.8** How hard is it to check whether two programs are equivalent?

**5.9** Try CBMC, using MiniSAT and Z3, on `f11`, `f12`, `f13`, and `f14` from this example. For details, see these lecture notes.

**5.10** Check the Static Single Assignment (SSA) example in these slides.

**5.11** You can find many examples in Dennis Yurichev's SAT/SMT by Example.