CS357: Second Home Assignment

– First Order Logic–

This assignment is intended to be solved **individually**, but discussion via Piazza is encouraged. Submit your report via email to `zeljic@stanford.edu` with subject CS357 - Assignment 2. The deadline is **Tuesday November 5$^{\mathbf{th}}$**.

1. Consider the following signature $\Sigma = (D, P)$, with the domain set $D = \{1, 2, 3, 4\}$ and the relation $P = \{(1, 1), (2, 1), (2, 4), (3, 3), (4, 2), (4, 3)\}$. Which of the following Σ-sentences are true:

   (a) $\forall x \, \exists y \, Pxy$

   (b) $\forall x \, (Pxx \vee \exists y \, Pyx)$

   (c) $\exists x \, \forall y \, \neg Pyx$

   (d) $\forall x \, \exists y \, Pxy \wedge Pyx$

   Answer:

   (a) Yes, every element is related to another

   (b) Yes, every element is either related to itself or has an element related to itself

   (c) No, every element has an element related to itself (negation of (b))

   (d) Yes, for every element exists a pair such that they are both related to one another.

2. Suppose P is a binary predicate. Show that no one of the following sentences is logically implied by the other two. Do this by giving a model for each sentence in which the sentence is false but the other two sentences are true.

   (a) $\forall x \, Pxx$

   (b) $\forall x \, \forall y \, (Pxy \vee Pyx \vee x = y)$

   (c) $\exists x \, \forall y \, Pxy$

   Answer:

   Consider a model M with $(M) = \{a, b, c\}$.

(a),(b),¬(c): $P^M = \{(a,a),(b,b),(c,c),(a,b),(b,c),(c,a)\}$

(a),¬(b),(c): $P^M = \{(a,a),(b,b),(c,c),(a,b),(a,c)\}$

¬(a),(b),(c): $P^M = \{(a,a),(a,b),(a,c),(b,c)\}$

3. Consider a language with equality and a single binary predicate symbol P. For each set $\mathcal{M}$ of models below, write a first order sentence ϕ such that $\models_M \phi$ iff $M \in \mathcal{M}$.

   (a) $\mathcal{M} = \{M | P^M$ is a transitive relation $\}$.
   Answer: $\forall x \, \forall y \, \forall z \, ((Pxy \wedge Pyz) \rightarrow Pxz)$

   (b) $\mathcal{M} = \{M | P^M$ defines a function $\}$.
   Answer: $\forall x \, \forall y \, \forall z \, ((Pxy \wedge Pxz) \rightarrow y = z) \wedge \forall x \, \exists y \, Pxy$

   (c) $\mathcal{M} = \{M | P^M$ is a bijection (i.e. a function that is 1-1 and onto) $\}$.
   Answer: $\forall x \, \forall y \, \forall z \, ((Pxy \wedge Pxz) \rightarrow y = z) \wedge \forall x \, \exists y \, Pxy \wedge$
   $(\forall x \, \forall y \, \forall z \, ((Pxy \wedge Pzy) \rightarrow x = z) \wedge \forall y \, \exists x \, Pxy$

4. Consider a signature Σ with no constant symbols, no predicate symbols (except for equality), and a single binary function symbol, $+$. Let M be a Σ-model with domain (the natural numbers) which interprets $+$ in the standard way.

   Note that the only non-logical symbols you may use are $=$ and $+$.

   (a) Give a Σ-formula which defines the set $\{0\}$ in M.
   Answer:
   $v_1 + v_1 = v_1$

   (b) Give a Σ-formula which defines the set $\{1\}$ in M.
   Answer:
   $\forall v_2 \, (v_2 + v_2 = v_2 \rightarrow (v_2 \neq v_1 \wedge \forall v_3 \, (v_3 = v_2 \vee \exists v_4 \, (v_3 = v_1 + v_4))))$

   (c) Give a Σ-formula which defines the binary relation $\{\langle m, n \rangle \, | m < n\}$ in M.
   Answer:
   $\forall v_3 \, (v_3 + v_3 = v_3 \rightarrow \exists v_4 \, (v_1 + v_4 = v_2 \wedge v_3 \neq v_4))$

5. Mapping a monotonic function to a sorted array preserves sortedness. Encode this property into AUFLIA logic of SMT-LIB and check whether the property holds.

   Answer: See Fig. 1

6. (BONUS) Consider the following puzzle:

   A room has N light switches, numbered by the positive integers 1 through N. There are also N children, numbered by the positive integers 1 through N. Initially, the switches are all off. Each child k enters the room and changes the position of every light switch n such that n is a multiple of

```
(set-logic AUFLIA)
(set-option :produce-models true)
(declare-const a (Array Int Int))
(declare-fun f (Int) Int)
(declare-const mappedf_a (Array Int Int))

;; predicate for sorted array
(define-fun sorted ((arr (Array Int Int))) Bool
(forall ((i Int) (j Int)) (=> (<= i j) (<= (select arr i) (select arr j))))
)

;; assert that f is monotonic
(assert (forall ((x1 Int) (x2 Int)) (=> (<= x1 x2) (<= (f x1) (f x2)))))

;; assert that mappedf_a is f mapped over a
(assert (forall ((x1 Int)) (= (select mappedf_a x1) (f (select a x1)))))

;; the array starts sorted
(assert (sorted a))

;; is it possible for the mapped version to not be sorted?
(assert (not (sorted mappedf_a)))

(check-sat)
```

Figure 1: SMT-LIB code for problem 5

k. That is, child 1 changes all the switches, child 2 changes switches 2, 4, 6, 8, ..., child 3 changes switches 3, 6, 9, 12, ..., etc., and child N changes only light switch N. When all the children have gone through the room, how many of the light switches are on?

1 Write a program that encodes this puzzle for a given value of N in SMT-LIB format using the logic of quantifier-free bit-vectors. For the report, describe what kinds of constraints are generated.

2 Download and install CVC4 from `cvc4.github.io`. What is the largest N CVC4 can solve the problem for, within 10minutes.

3 Intuitively, what is the solution to the puzzle?

For the SMT-LIB problems the documentation can be found at `http://smtlib.cs.uiowa.edu`. It can be useful to use online interfaces to SMT-solvers to work through some examples — CVC4 interface and Z3 interface.