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 5th.

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 $\Sigma$-sentences are true:

(a) $\forall x \exists y Pxy$
(b) $\forall x (Pxx \lor \exists y Pyx)$
(c) $\exists x \forall y \neg Pyx$
(d) $\forall x \exists y Pxy \land Pyx$

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) $\forall x \forall y (Pxy \lor Pyx \lor x = y)$
(c) $\exists x \forall y Pxy$

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 $\phi$ such that $\models_{\mathcal{M}} \phi$ iff $\mathcal{M} \in \mathcal{M}$.

(a) $\mathcal{M} = \{M|P^M \text{ is a transitive relation}\}$.
(b) $\mathcal{M} = \{M|P^M \text{ defines a function}\}$.
(c) $\mathcal{M} = \{M|P^M \text{ is a bijection (i.e. a function that is 1-1 and onto)}\}$.

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

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

1
(a) Give a $\Sigma$-formula which defines the set $\{0\}$ in $M$.

(b) Give a $\Sigma$-formula which defines the set $\{1\}$ in $M$.

(c) Give a $\Sigma$-formula which defines the binary relation $\{(m, n) \mid m < n\}$ in $M$.

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.

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 $k$. 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 10 minutes.

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.