EPFL Formal Verification Course Exam, October 2022

The exam is from 15:15 to 18:00. Do not open the exam until we tell you to.
Place your CAMIPRO card on your desk.
Put all electronic devices in a bag away from bench.
Write using permanent pen (no graphite nor heat-disappearing pen).

Write answers to different problems (1–6) on disjoint sheets of white paper that we supply.
Write your name, SCIPER and question number on top of each sheet you return.
Do not write the solutions that you want us to grade on the sheets with exam questions; please take these printed exam sheets with you after the exam.

Each subquestion is scored independently. Wrong answers do not penalize other parts.

We advise you to first solve questions that you find easier. If you expect you are running out of time on a particular problem, try to convince us that you know the right strategy to solve it.

You are allowed to use every true statement (e.g. theorem, equality) from the lecture slides provided that you clearly repeat it and refer to it as "seen in the lecture" (preferably with slide title or lecture name).

The exam is open book in the sense that you are allowed to take with you any printed material. Please do not take slides that are hand-annotated but only original ones or printed annotations.

The maximal number of points on the exam is 40.

Small reminder: the following are the names of some basic properties of a relation r:

- reflexive (R): $\forall x.\ (x, x) \in r$
- antisymmetric (A): $\forall x. \forall y.\ (x, y) \in r \wedge (y, x) \in r \rightarrow x = y$
- symmetric (S): $\forall x. \forall y.\ (x, y) \in r \rightarrow (y, x) \in r$
- transitive (T): $\forall x. \forall y. \forall z.\ (x, y) \in r \wedge (y, z) \in r \rightarrow (x, z) \in r$

Equivalence relation is a relation that is reflexive, symmetric, and transitive.
Partial ordering relation is a relation that is reflexive, antisymmetric, and transitive.

1 Weakest Precondition (3pt)

Let $r \subseteq S \times S$ and $Q \subseteq S$. Give an expression defining weakest precondition $\mathsf{wp}(r, Q)$ using operations of inverse of a relation, $(\_^{-1})$, set difference $(\setminus)$, and image of a relation under a set, $(\_[\_])$. Prove that your expression is correct by expanding the definitions of wp as well as of relational and set operations.

2 Iterating a Relation (7pt)

Let $M = (S, I, r, A)$ be a transition system and $\bar{r} = \{(s, s') \mid (s, a, s') \in r\}$, as usual. Let $\Delta = \{(x, x) \mid x \in S\}$.

Let $\bar{r}^k$ denote the usual composition of relation $\bar{r}$ with itself k times.

Define the sequence of relations r_n, for all non-negative integers n, as follows:

- $r_0 = \Delta \cup \bar{r}$
- $r_{n+1} = r_n \circ r_n$

A) (2pt) Prove that $r_n \subseteq r_{n+1}$ for every n.

B) (3pt) Prove that for every n and every k where $0 \leq k \leq 2^n$ we have $\bar{r}^k \subseteq r_n$.

C) (2pt) Suppose that S is finite. Find a bound B as a function of $|S|$ such that

$$\mathsf{Reach}(M) \subseteq r_B[I]$$

Aim to find as small bound as possible.

3 Propositional Logic with If (8pt)

Consider the ternary propositional operation $\mathsf{ite}(\mathsf{x}, \mathsf{y}, \mathsf{z})$ (if x then y else z) defined by the following truth table:

x	y	z	$\mathsf{ite}(\mathsf{x}, \mathsf{y}, \mathsf{z})$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

We consider expressions build only from variables and ite, without constants 0 and 1. We call them pure ite expressions.

A) (1pt) Express $x \wedge y$ as a pure ite expression;

B) (1pt) Express $x \vee y$ as a pure ite expression;

C) (1pt) Let e denote an ite expression whose set of free variables is $V = \{x_1, \ldots, x_n\}$. Let v be an assignment assigning all variables in V to 0 (false). What is the truth value of e under v?

D) (1pt) Given an example of a function $f : \{0,1\}^n \to \{0,1\}$ that is not expressible as a pure ite expression.

E) (4pt) List (in the separate sheet) the answer numbers next to all true completions of the statement:

The functions $f : \{0,1\}^n \to \{0,1\}$ that can be expressed as pure ite expressions using variables $x_1, \ldots, x_n$ are precisely ...

1. ... the functions that are not constant, i.e., not equal to either the function $f_1(x_1, \ldots, x_n) = 1$ nor to the function $f_0(x_1, \ldots, x_n) = 0$
2. ... all functions if $n \geq 3$; functions expressible using $\wedge, \vee$ in cases $n = 1, 2, 3$
3. ... the functions that could be expressed using an expression with only $\wedge$ and $\vee$
4. ... the functions f such that $f(0, \ldots, 0) = 0$
5. ... the functions f such that $f(0, \ldots, 0) = 0$ and $f(1, \ldots, 1) = 1$
6. ... the functions expressible using $\neg(x \wedge y)$ as a binary operation
7. ... the functions such that $f(x, \ldots, x) = x$ for every x
8. ... the functions such that $f(x, \ldots, x, f(x, \ldots, x)) = x$ for every x

For those answers that you chose, explain briefly why they are correct.

4 Resolution with Congruence (6pt)

Consider a set D with a binary relation $\equiv$ on D and a binary function $\sqcup : D^2 \to D$ which satisfy the following properties:

$$\begin{array}{ll} \mathsf{assoc}: & x \sqcup (y \sqcup z) \equiv (x \sqcup y) \sqcup z \\ \mathsf{transE}: & (x \equiv y) \wedge (y \equiv z) \to (x \equiv z) \\ \mathsf{sym}: & (x \equiv y) \to (y \equiv x) \\ \mathsf{congL}: & (x_1 \equiv x_2) \to (x_1 \sqcup y \equiv x_2 \sqcup y) \\ \mathsf{congR}: & (y_1 \equiv y_2) \to (x \sqcup y_1 \equiv x \sqcup y_2) \end{array}$$

Define another binary relation $\sqsubseteq$ on D by:

$$x \sqsubseteq y \quad \leftrightarrow \quad x \sqcup y \equiv y \qquad (\mathsf{DefLE})$$

We claim that it follows that $\sqsubseteq$ is a transitive relation, that is:

$$x \sqsubseteq y \wedge y \sqsubseteq z \;\rightarrow\; x \sqsubseteq z \qquad \text{(TransLE)}$$

In other words, we wish to prove that the following holds in pure first-order logic, where all formulas are considered universally quantified:

$$\{\mathsf{assoc}, \mathsf{transE}, \mathsf{sym}, \mathsf{congL}, \mathsf{congR}, \mathsf{DefLE}\} \models \mathsf{TransLE} \qquad (*)$$

Note that, when representing the problem in first order logic, $t_1 \equiv t_2$ is just a notation for $E(t_1, t_2)$ where E is some binary predicate symbol, $t_1 \sqsubseteq t_2$ is a notation for $L(t_1, t_2)$ where L is another binary predicate symbol, and $t_1 \sqcup t_2$ is a notation for $f(t_1, t_2)$ where f is some binary function symbol. You can choose to either use notation $\equiv, \sqsubseteq, \sqcup$ or E, L, f, but use one or the other consistently.

Our goal is to use resolution for first-order logic to derive a formal proof of $(*)$ by deriving a contradiction.

A) (2pt) To begin the proof, write down a numbered sequence of *clauses* that you will need in your proof, corresponding to $\mathsf{assoc}, \mathsf{transE}, \ldots$ (whatever the initial set of clauses should be to prove $(*)$ by contradiction).

1. $\{x \sqcup (y \sqcup z) \equiv (x \sqcup y) \sqcup z\}$ (from assoc)
2. $\{\neg(x \equiv y), \neg(y \equiv z), (x \equiv z)\}$ (from transE)
3. ...

You may need more than one clause to encode some of the formulas.

B) (4pt) Continue to write proof steps where each step follows from previous ones. For each step indicate from which previous steps it follows and by which substitution of variables. The last step should be the empty clause $\emptyset$. To save you time in finding the resolution proof, we provide the following fragment of a Stainless file which we used to check this fact; even if this is not a resolution proof, the invocations of functions in the proof provides hints about the substitutions that you may want to use in your proof.

```
def sym(x: D, y: D) = {...}.ensuring( !(x ≡ y) || y ≡ x )
... // define the other axioms

def transitive(x: D, y: D, z: D) = {
  require(x ⊑ y)
  require(y ⊑ z)
  sym(y ⊔ z, z)
  congR(x, z, y ⊔ z)
  assoc(x, y, z)
  trans(x ⊔ z, x ⊔ (y ⊔ z), (x ⊔ y) ⊔ z)
  congL(x ⊔ y, y, z)
  transE(x ⊔ z, (x ⊔ y) ⊔ z, y ⊔ z)
  transE(x ⊔ z, y ⊔ z, z)
}.ensuring(x ⊑ z)
```

5 Approximating Relations (7pt)

Consider a guarded command language whose meanings are binary relations on the set states U.

Let $E(a_1, \ldots, a_n)$ denote an expression built from some atomic relations $a_1, \ldots, a_n$, as well as diagonal relations

$$\Delta_P = \{(x, x) \mid x \in P\}$$

for various sets $P \subseteq U$. The expression E is built from these relations using union (to model non-deterministic choice, relation composition (to represent sequential composition) and transitive closure (to represent loops).

Let us call a relation $s \subseteq U \times U$ an *effect* if it is reflexive (R) and transitive (T).

A) (2pt) Prove that if s is an effect and $a_i \subseteq s$ for all $1 \leq i \leq n$, then

$$E(a_1, \ldots, a_n) \subseteq s$$

B) (2pt) Let $U = \mathbb{Z}^2$ denote pairs of integers, denoted by integer variables x, y. Let s be a specification relation given by the formula:

$$s = \{((x, y), (x', y')) \mid y \geq 0 \rightarrow (x' \leq x \wedge y' \geq 0)\}$$

Show that s is an effect.

C) (3pt) For the effect s in the previous point, prove that $\rho(p) \subseteq s$ where p is the following program (the initial values of variables can be arbitrary):

```
while (y ≥ 0) {
  x = x − y;
  if (x % 2 == 0)
    y = y / 2
  else
    y = 3*y + 1
}
```

Notation $\rho(p)$ denotes the relation corresponding to the program p.

6 Logic of Partial Functions (9pt)

We wish to use first-order logic for our verification task, which requires proving validity of formulas. We use a first-order language (signature) with a relational symbol E, which we would like to represent equality and satisfy the laws of reflexivity, symmetry, and transitivity, as well as congruence laws (analogous to congL and congR in Problem **??**): equal elements are related in the same way by all other relation symbols. We also need a way to represent a *partial* function $\bar{p}(x, y)$ of two arguments: the function can have at most one result, but

it can be undefined for certain pairs of elements. Applying $\bar{p}$ when argument is undefined gives undefined result. We will analyze two possibilities for encoding such partial functions in first-order logic, with questions arising in each of them.

Encoding 1. We use a constant b to represent undefined element and a binary function symbol $p(x, y)$ to represent the function $\bar{p}$ (note that we use p to represent the function symbol, whereas $\bar{p}$ represents the function that interprets it). The language of formulas in this part has only symbols $\{E, p, b\}$.

A) (1pt) Write down, as universally quantified first-order logic formulas, the congruence properties of p with respect to E, namely: if in the expression $p(x, y)$ we change x to an E-related element or change y to an E-related element, the new result is E-related to $p(x, y)$.

B) (1pt) Write down, as a universally quantified first-order logic formula, the property that applying p when one of the arguments is undefined results in an undefined value.

C) (1pt) Describe Herbrand universe (the set of ground terms) in this language. Is it finite or infinite?

Encoding 2. We use a ternary relation symbol $P(x, y, z)$ to represent the fact $\bar{p}(x, y) = z$ when all values are defined. To represent that the function is not defined for a given x and y, relation interpreting P would simply not contain any (interpretation of) z such that $P(x, y, z)$ is true. Let a be a constant denoting an arbitrary element of the universe. The language of formulas in this part has only symbols $\{E, P, a\}$.

D) (1pt) Write down, as universally quantified first-order logic formulas, the congruence properties of P with respect to E, namely: if elements x, y, z are related by P, then elements E-related to them are also related by P, as expected by a relation with properties of equality.

E) (1pt) Write down as a universally quantified formula, the property that P should represent a functional relation modulo E: for given pair of elements x, y, the result of $\bar{p}$, if it exists, is unique up to E.

F) (1pt) Consider the result of applying Skolemization of the properties in D) and E). How many new Skolem functions are introduced? Describe the Herbrand universe (the set of ground terms) in this language. Is it finite or infinite?

G) (3pt) Is there a terminating algorithm for checking, given two formulas F_1, F_2 in prenex form with only universal quantifiers using symbols from $\{E, P, a\}$, whether $\mathsf{Ax} \cup \{F_1\} \models F_2$ holds, where Ax are the Skolemized versions of properties introduced in D) and E). If yes, sketch the algorithm. If no, argue why the problem is undecidable.