Prof. Dr. Jürgen Giesl — Florian Frohn

Notes:

- Please solve these exercises in **groups of two**!
- The solutions must be handed in **directly before (very latest: at the beginning of)** the exercise course on Wednesday, November 13th, 2015, in lecture hall **AH 1**. Alternatively you can drop your solutions into a box which is located right next to Prof. Giesl's office (until the exercise course starts).
- Please write the **names** and **immatriculation numbers** of all (two) students on your solution. Please staple the individual sheets!

---

## Exercise 1 (An Application of Congruence Closure): (2 + 3 + 3 = 8 points)

Consider the following code fragment of an imperative program:

```
a = c;
d = f[f[c]];
f[b] = f[c];
if ( b == f[a] ) {
   (*)
}
```

The fragment has been translated to the following set of term equalities $\mathcal{E}$.

$$\begin{aligned} \mathsf{a} &\equiv \mathsf{c} \\ \mathsf{d} &\equiv \mathsf{f}(\mathsf{f}(\mathsf{c})) \\ \mathsf{f}(\mathsf{b}) &\equiv \mathsf{f}(\mathsf{c}) \\ \mathsf{b} &\equiv \mathsf{f}(\mathsf{a}) \end{aligned}$$

**a)** Show via $\leftrightarrow^*_{\mathcal{E}}$ that $\mathsf{d} \equiv_{\mathcal{E}} \mathsf{f}(\mathsf{c})$ holds.

**b)** Show via congruence closure that $\mathsf{d} \equiv_{\mathcal{E}} \mathsf{f}(\mathsf{c})$ holds.

**c)** Give initial values for the variables `a, b, c, d` and for the array `f`, such that at the position `(*)` the value of `d` is not equal to that of `f[c]`. What is the problem?

## Exercise 2 (The Algorithm CONGRUENCE_CLOSURE): (4 points)

Consider the set of term equalities $\mathcal{E}$ consisting of the following ground identities:

$$\begin{aligned} \mathsf{a} &\equiv \mathsf{b} \\ \mathsf{c} &\equiv \mathsf{f}(\mathsf{c}) \\ \mathsf{f}(\mathsf{b}) &\equiv \mathsf{g}(\mathsf{b}) \\ \mathsf{d} &\equiv \mathsf{c} \\ \mathsf{f}(\mathsf{g}(\mathsf{b})) &\equiv \mathsf{f}(\mathsf{d}) \end{aligned}$$

Decide $\mathsf{g}(\mathsf{c}) \equiv_{\mathcal{E}} \mathsf{f}(\mathsf{f}(\mathsf{a}))$ using the Algorithm CONGRUENCE_CLOSURE from the lecture. Give the set $S$ and as intermediate results also the sets $L$ during each iteration of Step 4. You can omit singleton sets in $L$.

## Exercise 3 (Congruence Closure for Satisfiability): (6 + 2 + 1 = 9 points)

The goal of this exercise is to develop a decision procedure for the decidability of propositional logic with equalities and uninterpreted functions. Given a signature $\Sigma$ and a set of variables $\mathcal{V}$, the set of all formulas in this logic $\mathcal{F}$ is the smallest set such that:

- if $s, t \in \mathcal{T}(\Sigma, \mathcal{V})$, then $s \equiv t \in \mathcal{F}$
- if $\varphi, \psi \in \mathcal{F}$, then $\varphi \wedge \psi \in \mathcal{F}$ and $\varphi \vee \psi \in \mathcal{F}$
- if $\varphi \in \mathcal{F}$, then $\neg\varphi \in \mathcal{F}$

So we have, e.g., $\neg(\neg(x_1 \equiv x_2) \vee \neg(x_2 \equiv x_3)) \wedge x_4 \equiv x_5 \wedge \neg(\mathsf{f}(x_1) \equiv \mathsf{f}(x_2)) \wedge \neg(x_5 \equiv x_1) \in \mathcal{F}$.
Given a formula $\varphi \in \mathcal{F}$, our goal is to prove or disprove that there is an interpretation $I = (\mathcal{A}, \alpha, \beta)$ such that $I \models \varphi$. Here, the relation $\models$ is defined as follows:

$$\begin{array}{rcl} I \models s \equiv t & \iff & I(s) = I(t) \\ I \models \varphi \wedge \psi & \iff & I \models \varphi \text{ and } I \models \psi \\ I \models \varphi \vee \psi & \iff & I \models \varphi \text{ or } I \models \psi \\ I \models \neg\varphi & \iff & I \not\models \varphi \end{array}$$

**a)** Use the congruence closure procedure to develop a decision procedure for propositional logic with equalities and uninterpreted functions. So given $\varphi \in \mathcal{F}$, the procedure has to decide whether there is an interpretation $I = (\mathcal{A}, \alpha, \beta)$ with $I \models \varphi$.

Hints:

- Look for a way to reduce satisfiability of $\varphi \in \mathcal{F}$ to satisfiability of $\varphi' \in \mathcal{F}$ where $\varphi'$ does not contain variables.
- You can use the following lemma for ground terms $s_i, t_i, u_j, v_j$ with $i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}$:
  If there exist algebras $A_1, \ldots, A_m$ with $A_i \models u_1 \equiv v_1 \wedge \cdots \wedge u_n \equiv v_n \wedge \neg s_i \equiv t_i$ ($i \in \{1, \ldots, m\}$), then there exists also an algebra $A$ with $A \models u_1 \equiv v_1 \wedge \cdots \wedge u_n \equiv v_n \wedge \neg s_1 \equiv t_1 \wedge \cdots \wedge \neg s_m \equiv t_m$ (and vice versa).

**b)** Use your algorithm from **a)** to prove or disprove that that the formula $\neg(\neg(x_1 \equiv x_2) \vee \neg(x_2 \equiv x_3)) \wedge x_4 \equiv x_5 \wedge \neg(\mathsf{f}(x_1) \equiv \mathsf{f}(x_2)) \wedge \neg(x_5 \equiv x_1)$ is satisfiable.

**c)** Give a formula $\varphi \in \mathcal{F}$ and a *finite* carrier $\mathcal{A}$ such that $\varphi$ is satisfiable for all infinite carriers, but there is no interpretation $I = (\mathcal{A}, \alpha, \beta)$ such that $I \models \varphi$.