Prof. Dr. Jürgen Giesl
Marc Brockschmidt, Carsten Fuhs, Thomas Ströder

Exercise 1 (Anwendung des Kongruenzabschlusses): (2 + 3 + 3 = 8 points)

Gegeben sei das folgende Code-Fragment eines imperativen Programms.

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

Das Fragment wurde in das folgende Termgleichungssystem $\mathcal{E}$ übersetzt.

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

a) Zeigen Sie mittels $\leftrightarrow_{\mathcal{E}}^*$, dass $d \equiv_{\mathcal{E}} f(c)$ gilt.

b) Zeigen Sie mittels des Kongruenzabschlusses, dass $d \equiv_{\mathcal{E}} f(c)$ gilt.

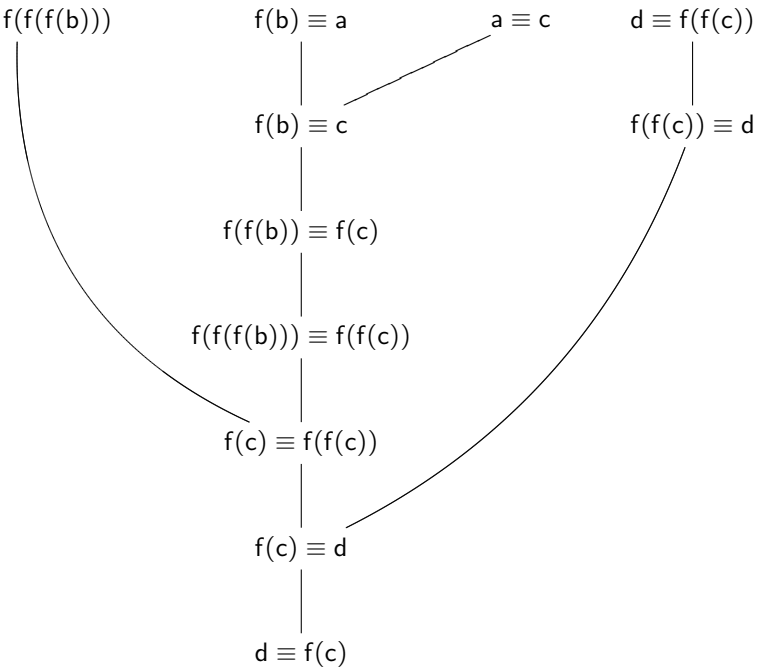
c) Geben Sie eine Anfangsbelegung der Variablen `a, b, c, d` und des Arrays `f` an, so dass an der Stelle `(*)` der Wert von `d` ungleich dem von `f[c]` ist. Wo liegt das Problem?

Solution:

a)

$$\begin{aligned} & \underline{d} \\ \rightarrow_{\mathcal{E}} \quad & f(f(\underline{c})) \\ \leftarrow_{\mathcal{E}} \quad & f(f(\underline{a})) \\ \leftarrow_{\mathcal{E}} \quad & \underline{f(f(f(b)))} \\ \leftarrow_{\mathcal{E}} \quad & f(c) \end{aligned}$$

b)



c) Wir wählen `a = b = c = d = 0`, `f[0] = 1`, `f[1] = 2`, `f[2] = 0`. Dann ist die Bedingung `f[b] == a` erfüllt, aber es gilt an der Stelle (*) `d = 2` $\neq$ `0 = f[c]`. Das Problem liegt an dem destruktiven Update in der Zeile `f[c] = ...`. Nach Ausführung dieser Zuweisung kann die vorherige Gleichheit `d = f[f[c]]` zunichte gemacht werden. Folglich war die einfache Übersetzung des Code-Fragments falsch, da sie Seiteneffekte nicht berücksichtigt hat. Eine korrekte Übersetzung ist möglich, i.a. aber aufwändiger und führt nicht immer zu Grundidentitäten.

Exercise 2 (Der Algorithmus KONGRUENZABSCHLUSS): (4 points)

Gegeben sei das Termgleichungssystem $\mathcal{E}$, das aus folgenden Grundidentitäten besteht:

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

Entscheiden Sie $g(c) \equiv_{\mathcal{E}} f(f(a))$ mittels des Algorithmus KONGRUENZABSCHLUSS aus der Vorlesung. Geben Sie die Menge S sowie als Zwischenergebnisse die Mengen L in jedem Durchlauf von Schritt 4 an.

Solution:

$S = \{a, b, c, d, f(a), f(b), f(d), f(f(a)), g(a), g(b), g(c)\}$.
L_i bezeichnet die Menge L vor Schritt 4 im i-ten Durchlauf, K_{i+1} ist die zugehörige Menge K direkt nach Schritt 4. In den K_i ist der Teil, der nur einelementige Mengen enthält, mit ... abgekürzt.

- $L_0 = \{\{a, b\}, \{c, d, f(d), g(b)\}, \{f(a)\}, \{f(b), g(a)\}, \{f(f(a))\}, \{g(c)\}\}$

- $K_1 = L_0 \cup \{\{f(a), f(b)\}, \{g(a), g(b)\}\} \cup \ldots$
- $L_1 = \{\{a, b\}, \{c, d, f(a), f(b), f(d), g(a), g(b)\}, \{f(f(a))\}, \{g(c)\}\}$
- $K_2 = L_1 \cup \{\{f(a), f(b)\}, \{g(a), g(b)\}, \{f(d), f(f(a))\}\} \cup \ldots$
- $L_2 = \{\{a, b\}, \{c, d, f(a), f(b), f(d), f(f(a)), g(a), g(b)\}, \{g(c)\}\}$
- $K_3 = L_2 \cup \{\{f(a), f(b)\}, \{g(a), g(b)\}, \{f(d), f(f(a))\}\} \cup \ldots$
- Nach Vereinigung ergibt sich $K_3 = L_2$, also sind wir fertig.

Da g(c) und f(f(a)) in der finalen Menge L_2 in unterschiedlichen Mengen liegen, gilt $g(c) \not\equiv_{\mathcal{E}} f(f(a))$.

Exercise 3 (Kongruenzabschluss für Allgemeingültigkeit): (6 + 3* + 3 = 9 + 3* points)

Ziel der Aufgabe ist es, ein Entscheidungsverfahren für die Allgemeingültigkeit von (implizit) allquantifizierten First-Order-Logik Formeln (FO-Formeln) zu entwickeln. FO-Formeln bestehen aus Termgleichungen und können mittels der Booleschen Operatoren $\neg, \vee, \wedge$ auf die übliche Art verbunden werden. Beispielsweise ist $\varphi = \neg\big(x \equiv f(f(x)) \wedge x \equiv f(f(f(f(f(x))))) \vee x \equiv f(x)$ eine FO-Formel mit $x \in \mathcal{V}$. Für Interpretationen $I = (\mathcal{A}, \alpha, \beta)$ ist die Modellbeziehung für FO-Formeln in der üblichen Weise definiert:

- $I \models \varphi_1 \vee \varphi_2$ gdw. $I \models \varphi_1$ oder $I \models \varphi_2$
- $I \models \varphi_1 \wedge \varphi_2$ gdw. $I \models \varphi_1$ und $I \models \varphi_2$
- $I \models \neg\varphi$ gdw. $I \not\models \varphi$
- $I \models u \equiv v$ gdw. $I(u) = I(v)$

Eine FO-Formel φ heißt *allgemeingültig* gdw. für alle Interpretationen I der Zusammenhang $I \models \varphi$ gilt. Eine FO-Formel φ heißt *unerfüllbar* gdw. es keine Interpretation I mit $I \models \varphi$ gibt.

a) Entwickeln Sie unter Nutzung des Kongruenzabschlussverfahrens ein Entscheidungsverfahren für die Allgemeingültigkeit von FO-Formeln. Hinweise:

- Zeigen Sie, wie man die Allgemeingültigkeit von FO-Formeln mit Variablen auf die Allgemeingültigkeit von FO-Formeln ohne Variablen zurückführen kann.
- Führen Sie die Allgemeingültigkeit von FO-Formeln auf die Unerfüllbarkeit mehrerer Konjunktionen der Art $u_1 \equiv v_1 \wedge \cdots \wedge u_n \equiv v_n \wedge \neg s_1 \equiv t_1 \wedge \cdots \wedge \neg s_m \equiv t_m$ zurück.
- Benutzen Sie das folgende Lemma für Grundterme s_i, t_i, u_j, v_j mit $i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}$:[1]
  Wenn es Algebren $A_1, \ldots, A_m$ mit $A_i \models u_1 \equiv v_1 \wedge \cdots \wedge u_n \equiv v_n \wedge \neg s_i \equiv t_i$ $(i \in \{1, \ldots, m\})$ gibt, dann gibt es auch eine Algebra A mit $A \models u_1 \equiv v_1 \wedge \cdots \wedge u_n \equiv v_n \wedge \neg s_1 \equiv t_1 \wedge \cdots \wedge \neg s_m \equiv t_m$ (und umgekehrt).

b) Wenden Sie Ihr Verfahren an, um die Allgemeingültigkeit der oben angegebenen FO-Formel φ nachzuweisen.

Solution:

[1] und beweisen Sie es für die Zusatzpunkte

a) Ein Entscheidungsverfahren für die Allgemeingültigkeit einer Formel φ:

1. Ersetze jede Variable x in φ durch eine neue Konstante c_x, erhalte φ_c.
2. Überführe $\neg\varphi_\mathsf{c}$ in disjunktive Normalform (DNF) (und zeige die Unerfüllbarkeit davon).
3. Finde für jedes Disjunkt der Form $u_1 \equiv v_1 \wedge \ldots u_n \wedge \equiv v_n \wedge \neg s_1 \equiv t_1 \wedge \cdots \wedge \neg s_m \equiv t_m$ ein $i \in \{1, \ldots, m\}$ mit $s_i \equiv_{\{u_1 \equiv v_1, \ldots, u_n \equiv v_n\}} t_i$. Benutze für das Wortproblem von $\equiv_{\{\ldots\}}$ den Algorithmus KONGRUENZABSCHLUSS.
4. Falls dies gelingt, gib JA aus, sonst gib NEIN aus.

Die Terminierung des Algorithmus ist offensichtlich. Betrachten wir nun die Korrektheit des Algorithmus:

1. Die Allgemeingültigkeit einer Formel φ (kurz: $\models \varphi$) ändert sich nicht unter der angegeben Transformation: Sei φ_c die Formel nach Ersetzung der Variablen $\{x_1, \ldots, x_n\}$. Für jede Interpretation $I = (\mathcal{A}, \alpha, \beta)$ über der ursprünglichen Signatur Σ gibt es eine zugehörige Interpretation $I_\mathsf{c} = (\mathcal{A}, \alpha \cup \{(\mathsf{c}_{x_1}, \beta(x_1)), \ldots, (\mathsf{c}_{x_n}, \beta(x_n))\}, \emptyset)$ über $\Sigma \cup \{\mathsf{c}_{x_1}, \ldots, \mathsf{c}_{x_n}\}$ und umgekehrt.[2] Man kann induktiv leicht zeigen, dass $I \models \varphi \Leftrightarrow I_\mathsf{c} \models \varphi_\mathsf{c}$ gilt. Daraus ergibt sich direkt $\models \varphi \Leftrightarrow \models \varphi_\mathsf{c}$.
2. Die Allgemeingültigkeit von φ_c ist offensichtlich äquivalent zur Unerfüllbarkeit von $\neg\varphi_\mathsf{c}$ und die Überführung zur DNF ändert die Semantik einer Formel nicht.
3. Die Unerfüllbarkeit von $\psi_1 \vee \cdots \vee \psi_n$ ist äquivalent zur Unerfüllbarkeit aller ψ_i. Daher ergibt sich der Schritt "Für jedes Disjunkt ...".

   Die Unerfüllbar eines Disjunkts der Form $u_1 \equiv v_1 \wedge \cdots \wedge u_n \equiv v_n \wedge \neg s_1 \equiv t_n \wedge \cdots \wedge \neg s_m \equiv t_m$ ist aber wegen des Lemmas unmittelbar äquivalent zur Unerfüllbarkeit einer der Formeln $u_1 \equiv v_1 \wedge \cdots \wedge u_n \equiv v_n \wedge \neg s_i \equiv t_i$. Daraus ergibt sich das "Finde ein $i \in \{1, \ldots, m\}$".

   Die Unerfüllbarkeit von $\theta_i = u_1 \equiv v_1 \wedge \cdots \wedge u_n \equiv v_n \wedge \neg s_i \equiv t_i$ ist aber äquivalent zu $s_i \equiv_{\{u_1 \equiv v_1, \ldots, u_n \equiv v_n\}} t_i$.

   - Denn falls $s_i \equiv_{\{u_1 \equiv v_1, \ldots, u_n \equiv v_n\}} t_i$ gilt, dann gilt für alle Algebren, die Modelle von $u_1 \equiv v_1, \ldots, u_n \equiv v_n$ sind, dass sie auch Modell von $s_i \equiv t_i$ sind. Also ist θ_i unerfüllbar.
   - Gilt hingegen $s_i \equiv_{\{u_1 \equiv v_1, \ldots, u_n \equiv v_n\}} t_i$ nicht, so gibt es eine Algebra A, die Modell von $u_1 \equiv v_1, \ldots, u_n \equiv v_n$, aber nicht von $s_i \equiv t_i$ ist. Folglich ist A ein Modell von θ_i, also ist θ_i erfüllbar.

$*$**)** Seien s_i, t_i, u_j, v_j mit $i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}$ Grundterme, sei $\varphi = u_1 \equiv v_1 \wedge \cdots \wedge u_n \equiv v_n$, sei $\psi_i = \varphi \wedge \neg s_i \equiv t_i$, sei $\psi = \varphi \wedge \neg s_1 \equiv t_1 \wedge \cdots \wedge \neg s_m \equiv t_m$. Die Aussage des Lemmas ist:

Die Formel ψ ist erfüllbar genau dann, wenn alle ψ_i erfüllbar sind.

$\Rightarrow$: Sei A eine Algebra mit $A \models \psi$, so folgt auch $A \models \psi_i$, weil ψ_i nur einige der Konjunktionen von ψ übernimmt.

$\Leftarrow$: Seien nun $A_i = (\mathcal{A}_i, \alpha_i)$ Algebren, so dass $A_i \models \psi_i$ gilt. Da keine Variablen vorhanden sind, kann man schon mittels der Algebra Terme auf Elemente des Universums eindeutig abbilden. Man braucht nicht die Variablenbelegung aus der Interpretation. Deshalb schreiben wir hier auch $A(t)$ anstelle von $I(t)$.

Wir definieren $A = (\mathcal{A}_1 \times \cdots \times \mathcal{A}_m, \alpha)$ mit

$$\begin{aligned}&\alpha(f)((a_{1,1}, \ldots, a_{m,1}), \ldots, (a_{1,n}, \ldots, a_{m,n}))\\ &= (\alpha_1(f)(a_{1,1}, \ldots, a_{1,n}), \ldots, \alpha_m(f)(a_{m,1}, \ldots, a_{m,n}))\end{aligned}$$

für jedes n-stellige Funktionssymbol f. Man kann nun induktiv über den Termaufbau zeigen, dass für jeden Term t der Zusammenhang $A(t) = (A_1(t), \ldots, A_m(t))$ gilt.

Weil $A_i \models \psi_i$ gilt, erhalten wir für jedes $j \in \{1, \ldots, n\}$ die Gleichheit

$$A(u_j) = (A_1(u_j), \ldots, A_m(u_j)) = (A_1(v_j), \ldots, A_m(v_j)) = A(v_j)$$

[2] Sei $I \models \varphi_\mathsf{c}$ mit $I = (\mathcal{A}, \alpha, \beta)$, so gibt es folgende Interpretation $I_\mathcal{V}$ für φ mit $I_\mathcal{V} \models \varphi$:
$I_\mathcal{V} = (\mathcal{A}, \alpha, \beta')$ mit $\beta'(x) = \beta(x)$ für $x \notin \mathcal{V}(\varphi)$ und $\beta'(x) = \alpha_{\mathsf{c}_x}$ für $x \in \mathcal{V}(\varphi)$.

und für jedes $i \in \{1, \ldots, m\}$ den Zusammenhang

$$A(s_i) = (A_1(s_i), \ldots, A_m(s_i)) \neq (A_1(t_i), \ldots, A_m(t_i)) = A(t_i)$$

da bereits $A_i(s_i) \neq A_i(t_i)$ gilt. Also ist A ein Modell von $\psi = u_1 \equiv v_1 \wedge \cdots \wedge u_n \equiv v_n \wedge \neg s_1 \equiv t_1 \wedge \cdots \wedge \neg s_m \equiv t_m$.

b) Wir wenden nun den Algorithmus auf

$$\neg\big(x \equiv \mathsf{f}(\mathsf{f}(x)) \wedge x \equiv \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(x))))) \vee x \equiv \mathsf{f}(x)$$

an, um die Allgemeingültigkeit dieser Formel zu entscheiden.

- Ersetzung der Variable x durch c_x liefert die Allgemeingültigkeits-äquivalente Formel

$$\neg\big(\mathsf{c}_x \equiv \mathsf{f}(\mathsf{f}(\mathsf{c}_x)) \wedge \mathsf{c}_x \equiv \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))))) \vee \mathsf{c}_x \equiv \mathsf{f}(\mathsf{c}_x)$$

- Negierung und Überführung in DNF führt zum Unerfüllbarkeitsproblem von

$$\mathsf{c}_x \equiv \mathsf{f}(\mathsf{f}(\mathsf{c}_x)) \wedge \mathsf{c}_x \equiv \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))))) \wedge \neg \mathsf{c}_x \equiv \mathsf{f}(\mathsf{c}_x)$$

- Dies ist gleichbedeutend zu $\mathsf{c}_x \equiv_{\mathcal{E}} \mathsf{f}(\mathsf{c}_x)$ für

$$\mathcal{E} = \{ \quad \mathsf{c}_x \equiv \mathsf{f}(\mathsf{f}(\mathsf{c}_x)), \quad \mathsf{c}_x \equiv \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))))) \quad \}$$

- Beim Nachweis mit dem Kongruenzabschlussverfahren ergeben sich die Mengen
  - $L_0 = \{\{\mathsf{c}_x, \mathsf{f}(\mathsf{f}(\mathsf{c}_x)), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))))\}, \{\mathsf{f}(\mathsf{c}_x)\}, \{\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x)))\}, \{\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))))\}\}$
  - $K_1 = L_0 \cup \{\{\mathsf{f}(\mathsf{c}_x), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x)))\}\} \cup \ldots$
  - $L_1 = \{\{\mathsf{c}_x, \mathsf{f}(\mathsf{f}(\mathsf{c}_x)), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))))\}, \{\mathsf{f}(\mathsf{c}_x), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x)))\}, \{\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))))\}\}$
  - $K_2 = L_1 \cup \{\{\mathsf{f}(\mathsf{c}_x), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x)))\}, \{\mathsf{f}(\mathsf{f}(\mathsf{c}_x)), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))))\}, \} \cup \ldots$
  - $L_2 = \{\{\mathsf{c}_x, \mathsf{f}(\mathsf{f}(\mathsf{c}_x)), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x)))), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))))\}, \{\mathsf{f}(\mathsf{c}_x), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x)))\}\}$
  - $K_3 = L_2 \cup \{\{\mathsf{f}(\mathsf{c}_x), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x)))\}, \{\mathsf{f}(\mathsf{c}_x), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))))\}, \{\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))))\}, \{\mathsf{f}(\mathsf{f}(\mathsf{c}_x)), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x)))\}, \} \cup \ldots$
  - $L_3 = \{\{\mathsf{c}_x, \mathsf{f}(\mathsf{c}_x), \mathsf{f}(\mathsf{f}(\mathsf{c}_x)), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x)))), \mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{f}(\mathsf{c}_x))))))\}\}$

  Also gilt $\mathsf{c}_x \equiv_{\mathcal{E}} \mathsf{f}(\mathsf{c}_x)$.

Folglich ist die Eingabe-Formel allgemeingültig.
