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

Consider the following code fragment of an imperative program:

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

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

\[
\begin{align*}
a & \equiv c \\
d & \equiv f(f(c)) \\
f(c) & \equiv f(f(f(b))) \\
f(b) & \equiv a
\end{align*}
\]

a) Show via $\leftrightarrow_\mathcal{E}$ that $d \equiv_\mathcal{E} f(c)$ holds.

b) Show via congruence closure that $d \equiv_\mathcal{E} f(c)$ 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 KONGRUENZABSCHLUSS (CONGRUENCE CLOSURE)): (4 points)

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

\[
\begin{align*}
a & \equiv b \\
c & \equiv f(d) \\
f(b) & \equiv g(a) \\
d & \equiv c \\
g(b) & \equiv d
\end{align*}
\]
Decide \( g(c) \equiv E \) using the Algorithm KONGRUENZABSCHLUSS (CONGRUENCE CLOSURE) from the lecture. Give the set \( S \) and as intermediate results also the sets \( L \) during each iteration of Step 4.

**Exercise 3 (Congruence Closure for Validity):**  
\((6 + 3^* + 3 = 9 + 3^* \text{ points})\)

The goal of this exercise is to develop a decision procedure for the validity of (implicitly) universally quantified first-order logic formulas (FO-formulas). FO-formulas consist of term equalities and can be connected by the Boolean operators \( \neg, \lor, \land \) in the usual way. For example, \( \varphi = \neg(x \equiv f(f(x))) \land x \equiv f(f(f(f(x)))) \lor x \equiv f(x) \) is a FO-formula with \( x \in V \).

For interpretations \( I = (A, \alpha, \beta) \) the model relationship for FO-formulas is defined in the usual way:

- \( I \models \varphi_1 \lor \varphi_2 \) iff \( I \models \varphi_1 \) or \( I \models \varphi_2 \)
- \( I \models \varphi_1 \land \varphi_2 \) iff \( I \models \varphi_1 \) and \( I \models \varphi_2 \)
- \( I \models \neg \varphi \) iff \( I \not\models \varphi \)
- \( I \models u \equiv v \) iff \( I(u) = I(v) \)

A FO-formula \( \varphi \) is called *valid* iff for all interpretations \( I \) we have \( I \models \varphi \). A FO-formula \( \varphi \) is called *unsatisfiable* iff no interpretation \( I \) with \( I \models \varphi \) exists.

a) Use the congruence closure procedure to develop a decision procedure for validity of FO-formulas. Hints:
- Show how one can reduce validity of FO-formulas with variables to validity of FO-formulas without variables.
- Reduce validity of FO-formulas to unsatisfiability of several conjunctions of the shape \( u_1 \equiv v_1 \land \cdots \land u_n \equiv v_n \land \neg s_1 \equiv t_1 \land \cdots \land \neg s_m \equiv t_m \).
- Use the following lemma for ground terms \( s_i, t_i, u_j, v_j \) with \( i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\} \):
  \[ \text{If there exist algebras } A_1, \ldots, A_m \text{ with } A_i \models u_1 \equiv v_1 \land \cdots \land u_n \equiv v_n \land \neg s_1 \equiv t_1 \land \cdots \land \neg s_m \equiv t_m, \text{ then there exists also an algebra } A \text{ with } A \models u_1 \equiv v_1 \land \cdots \land u_n \equiv v_n \land \neg s_1 \equiv t_1 \land \cdots \land \neg s_m \equiv t_m \text{ (and vice versa).} \]

b) Apply your procedure to show validity of the FO-formula \( \varphi \) given above.

---

\footnote{and prove it to obtain the bonus points}