6.0/4.0 VU Formale Methoden der Informatik				
185.291	**WS 2011**	**23 March 2012**		
Kennzahl (study id)	Matrikelnummer (student id)	Familienname (family name)	Vorname (first name)	Gruppe (version) **A**

1.) Consider the following problem:

> **EXISTS-HALTING**
>
> INSTANCE: A program Π, which takes as input a string over the alphabet $\{a, b, c, ..., z\}$.
>
> QUESTION: Does there exist an input string I, such that Π halts on I?

Prove that the problem **EXISTS-HALTING** is semi-decidable. For this, provide a procedure that shows the semi-decidability of the problem (i.e. a semi-decision procedure for **EXISTS-HALTING**) and argue that it is correct.

Hint: You may use the fact that the following problem is decidable:

SPECIAL-HALTING

INSTANCE: A program Π, a string I and a natural number n.

QUESTION: Does the program Π terminate on input I within n computational steps?

(15 points)

2.) (a) We consider a simplified variant of Tseitin's reduction. Let φ be a propositional formula, let $\Sigma(\varphi)$ be the set of all subformulas of φ, and let ℓ_φ be the label for φ. Prove that

$$(\bigwedge_{\psi\in\Sigma(\varphi)}(\ell_\psi \leftrightarrow \psi)) \rightarrow \ell_\varphi \text{ is valid if and only if } \varphi \text{ is valid.}$$

(9 points)

(b) Let F be an EUF-formula (EUF=equality logic with uninterpreted function symbols). You have a sound and complete SAT solver (say MiniSAT) which accepts formulas in CNF. Explain in detail how you transform F into a CNF for the SAT solver. What is the relation between F and the resulting CNF? **(6 points)**

3.) (a) Show that for all programs p, $\mathrm{wp}(p, \mathrm{true}) = \mathrm{false}$ if and only if $\mathrm{sp}(p, \mathrm{true}) = \mathrm{false}$.

Remember the definition of wp and sp:

$$\mathrm{wp}(p, \mathcal{S}_{\mathrm{out}}) = \{\, \sigma \in \mathcal{S} \mid [p]\,\sigma \text{ defined and } [p]\,\sigma \in \mathcal{S}_{\mathrm{out}} \,\}$$
$$\mathrm{sp}(p, \mathcal{S}_{\mathrm{in}}) = \{\, [p]\,\sigma \mid \sigma \in \mathcal{S}_{\mathrm{in}} \,\}$$

(5 points)

(b) Compute (not guess!) the weakest precondition of the following program for the postcondition $x = 4x_0$. What is the weakest *liberal* precondition for this program and postcondition?

$y \leftarrow 3x;$
while $2x \neq y$ do
  $x \leftarrow x + 1;$
  $y \leftarrow y + 1;$
od

Remember the weakest precondition of loops: $\mathrm{wp}(\text{while } e \text{ do } p \text{ od}, G) = \exists i\,(i \geq 0 \wedge F_i)$, where $F_0 = \neg e \wedge G$ and $F_{i+1} = e \wedge \mathrm{wp}(p, F_i)$. **(10 points)**

4.) Simulation and Bisimulation.

Let $M_1 = (S_1, I_1, R_1, L_1)$ and $M_2 = (S_2, I_2, R_2, L_2)$ be two Kripke structures.
Remember, a relation $H \subseteq S_1 \times S_2$ is a *simulation relation* from M_1 to M_2, if for each $(s, s') \in H$ it holds:

- $L_1(s) = L_2(s')$, and
- for each $(s, t) \in R_1$ there is a $(s', t') \in R_2$ such that $(t, t') \in H$.

Further remember, M_2 *simulates* M_1, if there is a simulation relation H from M_1 to M_2 such that for every initial state $s_1 \in I_1$ there is an initial state $s' \in I_2$ with $(s, s') \in H$. In this case we say that the relation H *witnesses* the simulation from M_1 to M_2.

Let $M_1 = (S_1, I_1, R_1, L_1)$ and $M_2 = (S_2, I_2, R_2, L_2)$ be two Kripke structures.
Remember, a relation $H' \subseteq S_1 \times S_2$ is a *bisimulation relation* if for each $(s, s') \in H'$ it holds:

- $L_1(s) = L_2(s')$,
- for each $(s, t) \in R_1$ there is a $(s', t') \in R_2$ such that $(t, t') \in H'$, and
- for each $(s', t') \in R_2$ there is a $(s, t) \in R_1$ such that $(t, t') \in H'$.

Further remember, M_1 and M_2 are bisimilar if there is a bisimulation relation $H' \subseteq S_1 \times S_2$ *such that*

- for each initial state $s \in I_1$ there is an initial state $s' \in I_2$ with $(s, s') \in H'$, and
- for each initial state $s' \in I_2$ there is an initial state $s \in I_1$ with $(s, s') \in H'$.

In the following, we say that H' *witnesses the bisimilarity of M_1 and M_2* in case H' is a bisimulation relation between M_1 and M_2 that satisfies the conditions stated above.

(a) State an algorithm that takes two finite Kripke structures $M_1 = (S_1, I_1, R_1, L_1)$ and $M_2 = (S_2, I_2, R_2, L_2)$ as input and returns a simulation relation from M_1 to M_2. In case M_2 simulates M_1, the relation computed by your algorithm must witness the simulation from M_1 to M_2. You may use *pseudo code.*

Prove that your algorithm is *complete*, i.e., prove that your algorithm eventually computes a relation which is a simulation relation, and *correct*, i.e., prove that the computed relation witnesses the simulation from M_1 to M_2 in case that M_2 simulates M_1.

(10 points)

(b) State two Kripke structures $M_1 = (S_1, I_1, R_1, L_1)$ and $M_2 = (S_2, I_2, R_2, L_2)$ and a relation H with the following properties:

- M_1 and M_2 are bisimilar,
- H is a simulation relation from M_1 to M_2 but not a bisimulation relation.

Explain why M_1 and M_2 are bisimilar and why H is not a bisimulation relation.

(5 points)