1	2	3	4	Σ

6.0/4.0 VU Formale Methoden der Informatik (185.291)

Dec 9, 2020

Kennz. (study id)	Matrikelnummer (student id)	Nachname (surname)	Vorname (first name)

1.) An undirected graph (V, E) is called a *mirror graph* if the following conditions hold:

- V can be partitioned into two sets of equal size $V' = \{v'_1, \dots, v'_n\}$ and $V'' = \{v''_1, \dots, v''_n\}$;
- for all $i \in \{1, \dots, n\}$ the edge $[v'_i, v''_i]$ is in E;
- for all $i, j \in \{1, \dots, n\}$, $[v'_i, v'_j] \in E$ iff $[v''_i, v''_j] \in E$;
- no other edges are contained in E.

In words, a mirror graph has each vertex from V' connected to a clone from V'' and the graph over V' is mirrored in V''.

Consider the following new problem **3COLMG** and recall the NP-complete problem **3COL** defined below:

> **3-COLORABILITY-MIRROR GRAPH (3COLMG)**
>
> INSTANCE: A mirror-graph $G = (V, E)$.
>
> QUESTION: Does there exists a function μ from vertices in V to values in $\{0, 1, 2\}$ such that $\mu(v_1) \neq \mu(v_2)$ for any edge $[v_1, v_2] \in E$.

> **3-COLORABILITY (3COL)**
>
> INSTANCE: An undirected graph $G = (V, E)$.
>
> QUESTION: Does there exists a function μ from vertices in V to values in $\{0, 1, 2\}$ such that $\mu(v_1) \neq \mu(v_2)$ for any edge $[v_1, v_2] \in E$.

(a) The following function f provides a polynomial-time many-one reduction from **3COL** to **3COLMG**: for a directed graph $G = (\{v_1, \dots, v_n\}, E)$, let

$$
\begin{aligned}
f(G) \quad = \big(\quad & \{v'_1, v''_1, \dots, v'_n, v''_n\}, \\
& \{[v'_i, v''_i] \mid i = 1 \dots n\} \cup \\
& \{[v'_i, v'_j], [v''_i, v''_j] \mid [v_i, v_j] \in E\} \big)
\end{aligned}
$$

Show that G is a yes-instance of **3COL** $\iff$ $f(G)$ is a yes-instance of **3COLMG**.

(9 points)

(b) Tick the correct statements (for ticking a correct statement a certain number of points is given; ticking an incorrect statement results in a substraction of the same amount; you cannot go below 0 points):

- ○ Given that **3COL** is in NP, our reduction shows that **3COLMG** is also in NP.
- ○ Given that **3COL** is in NP, our reduction shows that **3COLMG** is NP-hard.
- ○ Given that **3COL** is NP-hard, our reduction shows that **3COLMG** is also NP-hard.
- ○ Given that **3COL** is NP-hard, our reduction shows that **3COLMG** is in NP.
- ○ Since **3COLMG** is a special case of **3COL**, **3COLMG** must be contained in NP.
- ○ Since **3COLMG** is a special case of **3COL**, **3COLMG** must be NP-hard.

(6 points)

2.) (a) First define the concept of a theory and of a $\mathcal{T}$-interpretation. Then use them to define:

i. the $\mathcal{T}$-satisfiability of a formula;

ii. the $\mathcal{T}$-validity of a formula.

Additionally define the completeness of a theory $\mathcal{T}$. **(3 points)**

(b) Consider the function M, defined as follows.

Algorithm 1: The function M

Input: x, y, two *positive* integers
Output: The computed positive integer value for x, y

if $x == 1$ **then**
  return $2y$;
else if $y == 1$ **then**
  return x;
else return $\mathrm{M}(x-1, \mathrm{M}(x, y-1))$;

Let $\mathbb{N}$ denote the natural numbers *without* 0. Use well-founded induction to show

$$\forall x \, \forall y \; \big((x \in \mathbb{N} \wedge y \in \mathbb{N}) \;\rightarrow\; \mathrm{M}(x, y) \geq 2y\big).$$

(10 points)

(c) Suppose $\mathtt{M_C}$ is a correct implementation of M in the `C` programming language with x and y of type unsigned integers of size 32 bit (i.e., of type `uint32_t`). Is

$$\mathrm{M}(x', y') = \mathtt{M_C}(x', y')$$

true for all integers x', y' satisfying $1 \leq x', y' \leq$ `UINT32_MAX`, where `UINT32_MAX` is the largest value for a variable of type `uint32_t`?

If so, then prove this fact. Otherwise provide a counterexample with an exact explanation of what is computed and what is happening. **(2 points)**

3.) (a) Let p be the following program:

$$
\begin{array}{l}
y := n; \\
\textbf{while } y > 0 \textbf{ do} \\
\quad x := x - 4 * y + 2; \\
\quad y := y - 1 \\
\textbf{od}
\end{array}
$$

Give a loop invariant for the **while** loop in p and prove the validity of the partial correctness triple $\{n > 0 \wedge x = 2 * n^2 + 1\}\ p\ \{x = 1\}$.

(9 points)

(b) Provide a non-trivial pre-condition A and a non-trivial post-condition B, such that the total correctness triple $\{A\}\ p\ \{B\}$ is valid. Trivial means equivalent to `true` or `false`, so your precondition A and postcondition B should not be equivalent to `true` or `false`. The program p is given below.

Program p:

$$
\begin{array}{l}
y := x + y; \\
\textbf{while } x \neq y \textbf{ do} \\
\quad y := y + 1; \\
\quad x := x + 3; \\
\textbf{od}
\end{array}
$$

(2 points)

(c) Provide a state σ such that $\sigma \models [N \geq 0]$ `abort` $[B]$. In case such a σ does not exist, explain why there exists no such σ.

(2 points)

(d) What went wrong in the following argumentation:

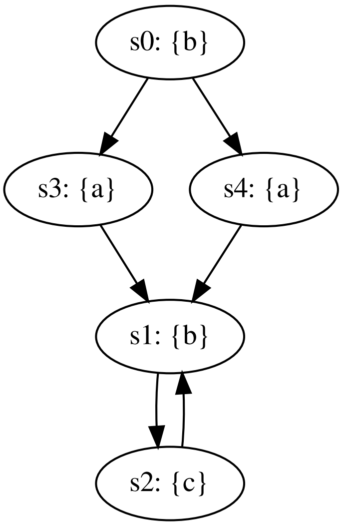
"In order to prove $\{A\}\ p\ \{B\}$ for any program p, we can simply start at the bottom of p and compute the weakest liberal precondition of the last expression with respect to B. We use the result as the new postcondition and work my way up to the first line of code. If the weakest liberal precondition of the first line is implied by A, we successfully proved partial correctness of $\{A\}\ p\ \{B\}$. "

Give a concise answer, 2-3 sentences suffice!

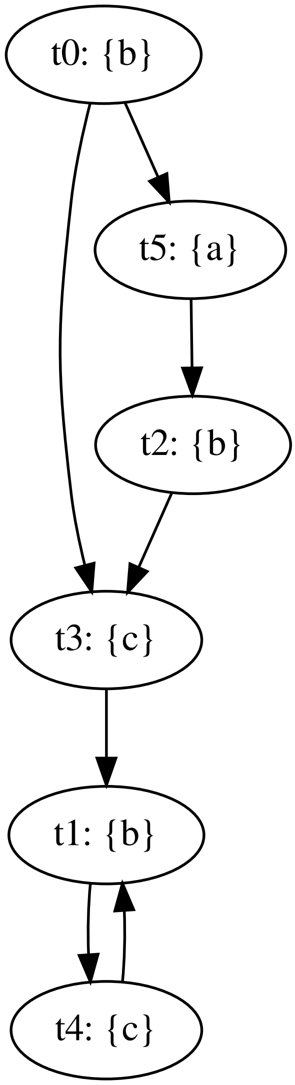
(2 points)

4.) (a) Provide a non-empty simulation relation H that witnesses $M_1 \leq M_2$, where M_1 and M_2 are shown below. The initial state of M_1 is s_0, the initial state of M_2 is t_0:

Kripke structure M_1:

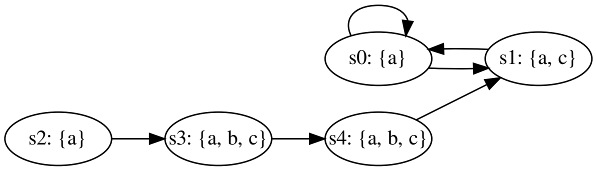


Kripke structure M_2:



(4 points)

(b) Consider the following Kripke structure M:



For each of the following formulae φ,

i. check the respective box if the formula is in CTL, LTL, and/or CTL*, and

ii. list the states s_i on which the formula φ holds; i.e. for which states s_i do we have $M, s_i \models \varphi$?

φ	CTL	LTL	CTL*	States s_i
$\mathbf{F}(c)$	☐	☐	☐	
$\mathbf{AX}(c)$	☐	☐	☐	
$\mathbf{E}[(a)\ \mathbf{U}\ (b)]$	☐	☐	☐	
$\mathbf{G}(a \wedge b)$	☐	☐	☐	
$\mathbf{EG}(a)$	☐	☐	☐	

(5 points)

(c) **LTL tautologies**

Prove that the following formulas are tautologies, i.e., they hold for every Kripke structure M and every path π in M, or find a Kripke structure M and path π in M, for which the formula does not hold and justify your answer.

i. $((\neg a \ \mathbf{U}\ b) \ \mathbf{U}\ \neg c) \Leftrightarrow (\neg a \ \mathbf{U}\ (b \ \mathbf{U}\ \neg c))$

ii. $(\mathbf{FG}a) \Rightarrow \mathbf{G}(a \vee \mathbf{XF}a)$

(6 points)

Grading scheme: 0–29 nicht genügend, 30–35 genügend, 36–41 befriedigend, 42–47 gut, 48–60 sehr gut