Block 1.)

Consider the following problem.

**CHAIN-HALTING**

**INSTANCE:** Two programs $\Pi_1$, $\Pi_2$, that take a string as input and output a string, and a string $I$.

**QUESTION:** Does $\Pi_2(\Pi_1(I))$ halt or $\Pi_1(\Pi_2(I))$ halt (or both), i.e., does one program halt when we use as input the output of the other program on input $I$?

**1.a)** The following function $f$ provides a polynomial-time many-one reduction from the **HALTING** problem to **CHAIN-HALTING**: for a program $\Pi$ and a string $I$, let $f((\Pi, I)) = (\Pi_1, \Pi_2, I')$ with $I' = I$, $\Pi_2 = \Pi$, and $\Pi_1$ given as follows:

```c
\Pi_1(string S) {
    return S;
}
```

(You can assume that the program $\Pi$ that is part the instances $(\Pi, I)$ of **HALTING** also takes a string as input and outputs a string.)

Show the correctness of the reduction, i.e.:

$(\Pi, I)$ is a yes-instance of **HALTING** $\iff f((\Pi, I))$ is a yes-instance of **CHAIN-HALTING**.

(10 points)
1.b) Check which statements are true/false. 1 point for each correct answer, -1 for each incorrect answer, 0 for no answer. Negative points do not carry over to other exercises.

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>true</td>
<td>false</td>
<td></td>
</tr>
<tr>
<td>○</td>
<td>○</td>
<td>The correctness of the reduction in (a) shows that <strong>CHAIN-HALTING</strong> is undecidable.</td>
</tr>
<tr>
<td>○</td>
<td>○</td>
<td>The correctness of the reduction in (a) shows that <strong>CHAIN-HALTING</strong> is semi-decidable.</td>
</tr>
<tr>
<td>○</td>
<td>○</td>
<td>The correctness of the reduction in (a) shows that the complement of <strong>CHAIN-HALTING</strong> is decidable.</td>
</tr>
<tr>
<td>○</td>
<td>○</td>
<td>If we would have a decision procedure for <strong>CHAIN-HALTING</strong>, we can solve <strong>HALTING</strong> using our reduction from (a).</td>
</tr>
<tr>
<td>○</td>
<td>○</td>
<td>If we would have a decision procedure for <strong>HALTING</strong>, we can solve <strong>CHAIN-HALTING</strong> using our reduction from (a).</td>
</tr>
</tbody>
</table>

(5 points)
2.a) Use Ackermann’s reduction and translate

\[ A(A(x)) \equiv A(B(x)) \rightarrow B(A(B(x))) \equiv y \lor C(x, y) \equiv C(A(x), B(x)) \]

to a satisfiability-equivalent E-formula \( \varphi^E \). \( A, B, \) and \( C \) are function symbols, \( x \) and \( y \) are variables. \( \text{(4 points)} \)
2.b) Consider the function $M$, defined as follows.

**Algorithm 1: The function $M$**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input:</strong></td>
<td>$x$, $y$, two positive integers</td>
</tr>
<tr>
<td><strong>Output:</strong></td>
<td>The computed positive integer value for $x$, $y$</td>
</tr>
<tr>
<td>1</td>
<td>if $x == 1$ then</td>
</tr>
<tr>
<td>2</td>
<td>return $2^y$;</td>
</tr>
<tr>
<td>3</td>
<td>else if $y == 1$ then</td>
</tr>
<tr>
<td>4</td>
<td>return $x$;</td>
</tr>
<tr>
<td>5</td>
<td>else return $M(x - 1, M(x, y - 1))$;</td>
</tr>
</tbody>
</table>

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

$$\forall x \forall y \left( (x \in \mathbb{N} \land y \in \mathbb{N}) \rightarrow M(x, y) \geq 2^y \right).$$

(11 points)
3.a) Let $p$ be the following IMP program, containing the integer-valued program variables $x, y, z$:

\[
\begin{align*}
    z &:= 0; y := 1 \\
    \textbf{while } x \neq 0 \textbf{ do} \\
    &\quad x := x - 1; \\
    &\quad z := z + x + y; \\
    &\quad y := y + 1; \\
    \textbf{od}
\end{align*}
\]

Give a loop invariant and variant for the \textbf{while} loop in $p$ and use them to formally prove the validity of the total correctness triple $[x = n \land n > 2] \ p [z = n^2]$.

\textbf{Note:} Make sure that your invariant expresses equalities among $x, y, z$ as well equalities among $x, y$.

(10 points)
3.b) Let \( p \) be the following IMP program, containing the integer-valued program variable \( x \):

\[
\textbf{while } x \geq 0 \textbf{ do } x := 1
\]

Which of the following Hoare triples are correct? Provide short justifications for your answers (no formal proofs are required).

(i) \( \{ x = 0 \} \quad p \quad \{ x = 1 \} \)

(ii) \( [ x = 0 ] \quad p \quad [ x = 1 ] \)

(iii) \( [ x \leq 0 ] \quad p \quad [ x = 1 ] \)

(iv) \( \{ x = -1 \} \quad p \quad \{ x = 1 \} \)

(v) \( [false] \quad p \quad [ x = 1 ] \)

(5 points)
4.a) Consider the Kripke structures $M_1$ and $M_2$. The initial state of $M_1$ is $s_0$, the initial state of $M_2$ is $t_0$.

**Kripke structure $M_1$:**

![Diagram of $M_1$]

**Kripke structure $M_2$:**

![Diagram of $M_2$]

i. Check whether $M_1$ and $M_2$ are bisimilar. If they are bisimilar, provide a bisimulation relation that witnesses $M_1 \equiv M_2$. If they are not bisimilar, provide a CTL* formula $\varphi$ that holds in exactly one of the structures, i.e., either $M_1 \models \varphi$ and $M_2 \not\models \varphi$, or $M_1 \not\models \varphi$ and $M_2 \models \varphi$. Indicate clearly which of the two structures satisfies the formula.

ii. Check whether $M_2$ simulates $M_1$, i.e., provide a simulation relation that witnesses $M_1 \preceq M_2$, or briefly explain why $M_2$ does not simulate $M_1$.

(4 points)
4.b) Consider the following Kripke structure $M$:

For each of the following formulae $\varphi$,

i. indicate whether the formula is in CTL, LTL, and/or CTL*, and

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

(If $\varphi$ is a path formula, list the states $s_i$ such that $M, s_i \models A \varphi$.)

<table>
<thead>
<tr>
<th>$\varphi$</th>
<th>CTL</th>
<th>LTL</th>
<th>CTL*</th>
<th>States $s_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>EG(c)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E(c U G b)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E(a U b)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G(c)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>F(a \land b)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(5 points)
4.c) An LTL formula is a \textit{tautology} if it holds for every Kripke structure $M$ and every path $\pi$ in $M$. For each of the following formulas, prove that it is a tautology, or find a Kripke structure $M$ and path $\pi$ in $M$ for which the formula does not hold and justify your answer.

i. $((Ga U Gb) \land \neg b) \Rightarrow FG(a \land b)$

ii. $FG(a \land b) \Rightarrow ((Ga U Gb) \land \neg b)$

(6 points)

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