1.) Recall the **HALTING** problem which takes a program and a string as input, and consider the following variant thereof:

**HALTING-EVEN**

INSTANCE: A program II that takes a string as input, a string I.

QUESTION: Does II terminate on I in an even number of computation steps.

(a) The following function f provides a polynomial-time many-one reduction from **HALTING** to **HALTING-EVEN**: for a program II and a string I let f(II, I) = (II′, I′) with I′ = I and II′ given as follows:

\[
\text{II'}(\text{string } S) \{
    \text{call II}(S);
    \text{call II}(S);
    \text{return};
\}
\]

(Remark: We assume that calls and return-statements do not count as computation steps).

Show that (II, I) is a yes-instance of **HALTING** ⇐⇒ (II′, I′) is a yes-instance of **HALTING-EVEN**.

(9 points)

(b) Please answer the following questions and explain your answers

- Is **HALTING-EVEN** undecidable?
- Is **HALTING-EVEN** semi-decidable?

(6 points)
2.) (a) Consider the following clause set \( \hat{\delta}(\varphi) \) which has been derived from an (unknown) formula \( \varphi \) by an improved version of Tseitin’s translation (atoms have not been labeled and \( \overline{z} \) means \( \neg z \)).

\[
\begin{align*}
C_1: & \ell_1 \lor x_1 \lor x_2 \\
C_2: & \ell_1 \lor \overline{x_1} \lor \overline{x_2} \\
C_3: & \ell_1 \lor \overline{x_1} \lor x_2 \\
C_4: & \ell_1 \lor x_1 \lor \overline{x_2} \\
C_5: & \ell_2 \lor x_2 \lor x_3 \\
C_6: & \ell_2 \lor \overline{x_2} \lor \overline{x_3} \\
C_7: & \ell_2 \lor \overline{x_2} \lor x_3 \\
C_8: & \ell_2 \lor x_2 \lor \overline{x_3} \\
C_9: & \ell_3 \lor \ell_1 \\
C_{10}: & \ell_3 \lor \ell_2 \\
C_{11}: & \ell_3 \lor \ell_1 \lor \ell_2 \\
C_{12}: & \ell_4 \lor x_1 \\
C_{13}: & \ell_4 \lor \ell_3 \\
C_{14}: & \ell_4 \lor \overline{x_1} \lor \ell_3 \\
C_{15}: & \ell_5 \lor x_1 \\
C_{16}: & \ell_5 \lor \ell_3 \\
C_{17}: & \ell_5 \lor \overline{x_1} \lor \ell_3 \\
C_{18}: & \ell_6 \lor \ell_4 \lor \ell_5 \\
C_{19}: & \ell_6 \lor \ell_4 \\
C_{20}: & \ell_6 \lor \ell_5 \\
\end{align*}
\]

(i) Reconstruct \( \varphi \) from \( \hat{\delta}(\varphi) \).

(ii) Prove the validity of \( \varphi \) by resolution (no additional translation to normal form is allowed!). You are allowed to add a single unit clause (i.e., a clause containing exactly one literal). Please explain your approach!

\(7\) points

(b) Let \( \mathcal{T}_E \) be the theory containing all equality axioms (from \( \mathcal{T}_E \)) and the following two additional axioms.

- \( \forall x \forall y (f(x) = f(y) \rightarrow x = y) \) (f-injectivity)
- \( \forall x f(x) = f(f(x)) \) (f-idempotency)

Clarify the logical status of each of the following formulas. If one is \( \mathcal{T}_E \)-valid or \( \mathcal{T}_E \)-unsatisfiable, then prove it using the semantic argument method. If one is \( \mathcal{T}_E \)-satisfiable but not \( \mathcal{T}_E \)-valid, then present a satisfying and a falsifying interpretation. Argue formally that the formula evaluates to true resp. false under the constructed interpretations.

\[
\begin{align*}
\varphi: & f(f(f(a))) = f(b) \rightarrow a \equiv b \\
\psi: & f(f(f(a))) \neq f(f(b)) \land a \equiv b \\
\end{align*}
\]

\(8\) points
3. (a) Let $p$ be the following IMP program loop, containing the integer-valued program variables $i, x, y$:

```
while $i < 10$ do
  $x := x - 2 \times i$;
  $i := i + 1$;
  $y := y + 2 \times i$;
od
```

Give a loop invariant for the `while` loop in $p$ and prove the validity of the total correctness triple $[i = 0 \land x = 100 \land y = 0] \implies p [x + y = 120]$.  

(10 points)

(b) Let $p$ be the IMP program:

```
if $2 \times x \geq y$ then $x := x + 2$ else $y := y - 2$
```

What is the \textit{wlp}($p, x = 4 \land y = 2$)?  

(3 points)

(c) Let $p$ be an IMP program such that $[2 \times x \geq y] \implies p [x \neq y]$ is valid. Is $[x = 2 \land y = 1] \implies p [2 \times y - 2 \times x \neq 0]$ valid? If so, give a formal proof. Otherwise, give a counterexample.  

(2 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 $M_1$:**

Kripke structure $M_2$:

---

i. Briefly explain why $M_2$ does not simulate $M_1$.

ii. Add a minimal set of edges to $M_2$ such that the extended Kripke structure $M'_2$ simulates $M_1$. Provide the additional edges and a non-empty simulation relation $H$ that witnesses $M_1 \preceq M'_2$.

(4 points)
(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>$G(b)$</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>$X(c)$</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>$EF(b)$</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>$AF(a \land b)$</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
<td>☐</td>
</tr>
</tbody>
</table>

(5 points)
(c) **LTL tautologies**

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

i. $G(a \Rightarrow Fb) \Rightarrow (GFa) \land (GFb)$

ii. $(GFa) \land (GFb) \Rightarrow G(a \Rightarrow Fb)$

(6 points)