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

**NON-TRIVIAL-HALTING (NTH)**

INSTANCE: A program $\Pi'$ that takes a string as input.

QUESTION: Is it true, that there exist input strings $I_1, I_2$, such that $\Pi'$ halts on $I_1$ and $\Pi'$ does not halt on $I_2$.

(a) The following function $f$ provides a polynomial-time many-one reduction from the co-**HALTING** problem (the complement of **HALTING**) to **NTH**: for a program $\Pi$ and a string $I$, let $f(\Pi, I) = (\Pi')$ with

$$\Pi'(\text{string } S) = \begin{cases} \text{if } (S = I) \{ \text{call } \Pi(S); \} \text{ return; } & \end{cases}$$

Show that $(\Pi, I)$ is a yes-instance of co-**HALTING** $\iff (\Pi')$ is a yes-instance of **NTH**.

(b) Recall that co-**HALTING** is not even semi-decidable and suppose that our reduction from co-**HALTING** to **NTH** is correct. Tick the correct statements that can be concluded from these observations (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):

- NTH is undecidable.
- NTH is semi-decidable.
- NTH is decidable.
- Suppose we have a decision procedure for NTH; then we would have a decision procedure for co-**HALTING**.
- Since HALTING is semi-decidable, our reduction also shows that the complement of NTH is semi-decidable.
- A problem or its complement is semi-decidable, thus the complement of NTH is semi-decidable.

(9 points)

(6 points)
2.) (a) Consider the following clause set $\hat{\delta}(\varphi)$ which has been derived from an (unknown) formula $\varphi$ by Tseitin translation (atoms have not been labeled).

\[ C_1 : \ell_1 \lor \neg \ell_1 \lor \neg y \quad C_2 : ~\neg \ell_1 \lor x \quad C_3 : ~\neg \ell_1 \lor y \]
\[ C_4 : ~\neg \ell_2 \lor \neg y \lor z \quad C_5 : \ell_2 \lor y \quad C_6 : \ell_2 \lor \neg z \]
\[ C_7 : ~\neg \ell_3 \lor \neg \ell_4 \lor z \quad C_8 : \ell_3 \lor \ell_1 \quad C_9 : \ell_3 \lor \neg z \]
\[ C_{10} : ~\neg \ell_4 \lor \neg x \lor \ell_2 \quad C_{11} : \ell_4 \lor x \quad C_{12} : \ell_4 \lor \neg \ell_2 \]
\[ C_{13} : ~\neg \ell_5 \lor \neg \ell_4 \lor \ell_3 \quad C_{14} : \ell_5 \lor \ell_4 \quad C_{15} : \ell_5 \lor \neg \ell_3 \]

(i) Reconstruct $\varphi$ from $\hat{\delta}(\varphi)$.
(ii) Start from $\hat{\delta}(\varphi)$ and extend it by a single nonempty clause $C$ in such a way that $\varphi$ is valid iff $\hat{\delta}(\varphi) \land C$ is unsatisfiable.
(iii) Prove the validity of $\varphi$ by resolution (no additional translation to normal form is allowed!).

(4 points)

(b) 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 validity-equivalent E-formula $\varphi^E$. $A$, $B$, and $C$ are function symbols, $x$ and $y$ are variables.

(3 points)

(c) Let $\varphi_{uf}$ be an equality formula containing uninterpreted functions. Let $FC^E(\varphi_{uf})$ and $flat^E(\varphi_{uf})$ be obtained by Ackermann’s reduction. Prove the following.

$\varphi_{uf}$ is satisfiable iff $FC^E(\varphi_{uf}) \land flat^E(\varphi_{uf})$ is satisfiable.

Hint: $FC^E$ is the same for $\varphi_{uf}$ and $\neg \varphi_{uf}$.

(8 points)

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

```plaintext
while $i < n$ do
    $x := x - 5$;
    $i := i + 1$
end
```

Which of the following program assertions are inductive loop invariants of $p$?

- $I_1$: $i \geq n$
- $I_2$: $i < n$
- $I_3$: $x + 5 \times i = 0$

Give formal details justifying your answer. That is, if an assertion is an inductive loop invariant, provide a formal proof of it based on Hoare logic. If an assertion is not an inductive loop invariant, give a counterexample.

**Note:** You need to use the definition of an assertion being inductive invariant.

(11 points)

(b) Let $p$ be the following IMP program loop, containing the integer-valued program variable $i$:

```plaintext
while $i \geq 3$ do
    if $i < 2$ then abort
    else $i := i - 1$
end
```

Which of the following Hoare triple are valid total correctness assertions?

- $[false] p [i = 2]$
- $[i = 3] p [i = 2]$
- $[true] p [i = 2]$

(4 points)
4.) (a) For two LTL formulas $\varphi_1$ and $\varphi_2$, the weak until operator ($\varphi_1 W \varphi_2$) requires $\varphi_1$ to remain true until $\varphi_2$ becomes true, but does not require that $\varphi_2$ ever becomes true.

i. Give a formal definition of the semantics of the weak until operator, i.e., provide a first order formula defining $M, \pi \models \varphi_1 W \varphi_2$ for a Kripke structure $M$ and a path $\pi$ of $M$.

ii. Express $W$ in terms of the LTL operators $U, X, F, G$ defined in the lecture.

(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>$E[(a) U (b)]$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G(a)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$F(a)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$EG(c)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$AX(b)$</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. $(G a) U (F b) \Rightarrow G(a U Fb)$

ii. $G(a U Fb) \Rightarrow (Ga) U (Fb)$

(6 points)