Exercise 1:
Let $A$ be a NFTA on an alphabet $F = F' \cup \{\{, \}\}$ where $F'$ contains a constant. Construct a two-way automata in the Horn clause formalism which recognizes the set of terms that can be deduced from this set of rules:

\[
\begin{align*}
\text{if } t \in L(A) & \quad \frac{u \quad v}{\{u\}_v} \text{ encrypt} \\
\text{decrypt} & \quad \frac{\{u\}_v \quad v}{u}
\end{align*}
\]

Explain why it is difficult to do the same construction with the other definition of two-way automata.

Solution:

- Let $A = \langle Q, F, \Delta \rangle$. We construct the two-way automata $A' = \langle Q', F', C' \rangle$ (in the Horn clause formalism) this way:
  - $Q' = Q \cup \{q_f, q_{dec}, q_{OK}, q_{\top}\}$
  - $F' = \{q_f\}$
  - $C' =$
    * $q_1(x_1), \ldots, q_n(x_n) \rightarrow q(f(x_1, \ldots, x_n))$ for all $f(q_1, \ldots, q_n) \rightarrow q \in \Delta$ (push clause)
    * $q(x) \rightarrow q_f(x)$ for all $q \in F$ (alternating clause)
    * $q_f(x), q_f(y) \rightarrow q_f(\{x\}_y)$ (push clause)
    * $q_\top(x_1), \ldots, q_\top(x_n) \rightarrow q_\top(f(x_1, \ldots, x_n))$ for all $f \in F$ (push clause)
    * $q_f(y), q_\top(x) \rightarrow q_{dec}(\{x\}_y)$ (push clause)
    * $q_{dec}(x), q_f(x) \rightarrow q_{OK}(x)$ (alternating clause)
    * $q_{OK}(\{x\}_y) \rightarrow q_f(x)$ (pop clause)
  - The problem with rule decrypt is that we deduce a tree from a bigger one which is difficult to do with the other definition.
  - That explains why there is no direct translation from Horn clause formalism to formalism of the lecture.

Exercise 2:
We have a naive translation from two-way automata (in the 'classic' sense) to two-way automata (in the Horn clause formalism) which does not work; let us assume that $F$ only contains unary symbols and constants and that you have an automaton $A$ which does not have non-deterministic or alternating rules. So $A$ only has rules of the form $\delta(q, a) = (q', d)$ with $d \in \{-1, 1\}$ or $\delta(q, a) = true$. For each transition $\delta(q, a) = (q', 1)$, we consider the clause $q'(x) \rightarrow q(a(x))$, for each transition $\delta(q, a) = (q', -1)$, we consider the clause $q'(f(a(x))) \rightarrow q(a(x))$ for all $f \in F$ and for each transition $\delta(q, a) = true$, we consider the clause $\rightarrow q(a(x))$. Let us call $C(A)$ this set of clauses.

1) Which clauses of $C(A)$ are not of the shape of the Horn clause formalism of a two-way automaton?
2) Give a two-way automaton in the Horn clause formalism which recognizes the interpretation of $q_f$ in the smallest Herbrand model of $C(A)$.
3) Give an example of an automaton $A$ such that the automaton obtained at the question 2) does not recognize the same language.
Solution:
1) The last two kinds.
2) You can replace \( \rightarrow q(a(x)) \) by the two clauses \( \rightarrow [q, a](x) \) (alternating clause) and \( [q, a](x) \rightarrow q(a(x)) \) (push clause). You can replace \( q'(f(a(x))) \rightarrow q(a(x)) \) by the two clauses \( q'(f(a(x))) \rightarrow [q, a](x) \) (pop clause) and \( [q, a](x) \rightarrow q(a(x)) \) (push clause).
3) Let the following two-way automata (in the ‘classic’ sense) \( \mathcal{A} \):
   - \( Q = \{q_0, q_1, q_i\} \)
   - \( I = \{q_i\} \)
   - \( \mathcal{F} = \{a(1), b(1), c(1), 0(0)\} \)
   - \( \delta = \)
     - \( \delta(q_0, 0) = \top \)
     - \( \delta(q_i, a) = (q_1, 1) \)
     - \( \delta(q_1, b) = (q_1, -1) \)
     - \( \delta(q_0, b) = (q_1, 1) \)
     - \( \delta(q_1, c) = (q_0, 1) \)
   Then \( a(b(c(0))) \notin L(\mathcal{A}) \). Indeed, if it were the case, an accepting run must start by:
     \[
     \begin{align*}
     (\epsilon, q_i) \quad &| (1, q_1) \quad | (\epsilon, q_0)
     \end{align*}
     \]
   But there is no transition of the form \( \delta(q_0, a) \) and so we cannot complete it to an accepting run.
   Let us prove now that \( a(b(c(0))) \) is in the language of the automata constructed at question 2. As, this set of Horn clause generates the same language as the \( C(\mathcal{A}) \), then we prove that we can generate \( a(b(c(0))) \) from \( C(\mathcal{A}) \). First, \( C(\mathcal{A}) \) consists of the following clauses:
   - i) \( \rightarrow q_0(0) \)
   - ii) \( q_1(x) \rightarrow q_i(a(x)) \)
   - iii) \( q_1(d(b(x))) \rightarrow q_1(b(x)) \) for \( d \in \{a, b, c\} \)
   - iv) \( q_1(x) \rightarrow q_0(b(x)) \)
   - v) \( q_0(x) \rightarrow q_1(c(x)) \)
   Note \( [q] \) be the model of the predicate (= state in our case) \( q \) in the smallest Herbrand model of \( C(\mathcal{A}) \).
   Form i), we deduce that \( 0 \in [q_0]. \)
   From v), we deduce that \( c(0) \in [q_1]. \)
   From iv), we deduce that \( b(c(0)) \in [q_0]. \)
   From v), we deduce that \( c(b(c(0))) \in [q_1]. \)
   From iii i.c) we deduce that \( b(c(0)) \in [q_1]. \)
   From ii), we deduce that \( a(b(c(0))) \in [q_i]. \)

Definition (PDL).
The syntax is the following:
\[
\begin{align*}
\phi ::= a & | \top & | \neg \phi & | \phi \lor \phi & | (\pi) \phi & | (\pi) (position \ formulae) \\
\pi ::= \downarrow & | \rightarrow & | \pi^{-1} & | \pi \cdot \pi & | \pi + \pi & | \pi^* & | \phi? (path \ formulae)
\end{align*}
\]
The semantic is defined this way: let \( t \) be a tree, we define \( \llbracket \phi \rrbracket_t \) (resp. \( \llbracket \pi \rrbracket_t \)) as a set of
Exercise 4:

Give a translation of PDL in MSO which preserves models. That is, given a position formula \( \phi \) (resp. \( \pi \)), construct a MSO formula \( \hat{\phi} \) (resp. \( \hat{\pi} \)) whose set of free variable is \( \{ X_a \mid a \in \mathcal{F} \} \) (resp. \( \{ X_a \mid a \in \mathcal{F} \} \cup \{ x, y \} \)) such that \( t, w \models \phi \) iff \( (P_a(t))_{a \in \mathcal{F}}, w, w' \models \hat{\phi} \) (resp. \( t, w, w' \models \pi \) iff \( (P_a(t))_{a \in \mathcal{F}}, w, w' \models \hat{\pi} \)) where \( P_a(t) = \{ w \in Pos(t) \mid t(w) = a \} \).

Solution:

By induction on the size of the formula :
- \( \phi = b \in \mathcal{F} : \hat{\phi}((X_a)_{a \in \mathcal{F}}, x) = x \in X_b \)
- \( \phi = \phi_1 \land \phi_2 : \hat{\phi}((X_a)_{a \in \mathcal{F}}, x) = \hat{\phi}_1((X_a)_{a \in \mathcal{F}}, x) \land \hat{\phi}_2((X_a)_{a \in \mathcal{F}}, x) \)
- idem for \( \lor \) and \( \top \)
- \( \phi = (\tau)\phi' : \hat{\phi}((X_a)_{a \in \mathcal{F}}, x) = \exists y. \hat{\pi}((X_a)_{a \in \mathcal{F}}, x, y) \land \hat{\phi}'((X_a)_{a \in \mathcal{F}}, y) \)
- \( \pi = \downarrow : \hat{\pi}((X_a)_{a \in \mathcal{F}}, x, y) = \text{child}(x, y) \)
- \( \pi = \rightarrow : \hat{\pi}((X_a)_{a \in \mathcal{F}}, x, y) = \text{nextSibling}(x, y) \)
- \( \pi = \pi'^{-1} : \hat{\pi}((X_a)_{a \in \mathcal{F}}, x, y) = \hat{\pi}'((X_a)_{a \in \mathcal{F}}, y, x) \)
- \( \pi = \pi_1, \pi_2 : \hat{\pi}((X_a)_{a \in \mathcal{F}}, x, y) = \exists z. \hat{\pi}_1((X_a)_{a \in \mathcal{F}}, x, z) \land \hat{\pi}_2((X_a)_{a \in \mathcal{F}}, z, y) \)

Exercise 3:

Let \( t \) be the tree :

```
     a
    / \
   b  c
  /|
 / \
a b
```

Which formulae are satisfied by \( t \)?

1. \( \phi_1 = \neg a \lor \downarrow \left( \neg \leftarrow \top \land b \land (\rightarrow^* (c \land \neg \downarrow (\rightarrow) \top)) \right) \)
2. \( \phi_2 = \neg a \lor \downarrow \left( \neg \leftarrow \top \land b \land ((\rightarrow; c)^*) (\neg (\rightarrow) \top) \right) \)
3. \( \phi_3 = ((a^?; \downarrow^*)^* (a \land \neg \downarrow (\downarrow) \top) \)

Solution:

1. yes
2. no
3. no
\[ \pi = \pi_1 + \pi_2 : \tilde{\pi}(\{x_a \in F, x, y\} \models \pi_1((X_a)_{a \in F}, x, y) \lor \pi_2((X_a)_{a \in F}, x, y)) \]

\[ \pi = \pi^\ast : \tilde{\pi}(\{x_a \in F, x, y\} \models \forall X (x \in \forall z_1, z_2. ((z_1 \in X \land \pi^\ast((X_a)_{a \in F}, z_1, z_2)) \Rightarrow z_2 \in X)) \Rightarrow y \in X \]

\[ \pi = \tilde{\phi} : \tilde{\pi}(\{x_a \in F, x, y\} \models (x = y) \land \tilde{\phi}(\{x_a \in F, x\}) \]

Exercise 5:
Fix an alphabet \( F \). Give a PDL formula \( \pi \) such that:

- for all tree \( t \) and all position \( p \) of \( t \), there exists exactly one position \( q \) of \( t \) such that \( (p, q) \in [\pi]_t \) (\( \pi \) defines a function on positions).
- for all tree \( t \) and position \( p \) of \( t \), \( (p, q) \in [\pi^\ast]_t \iff q \) is a position of \( t \) such that \( t(q) = t(p) \).

Solution:
Hard. There is no point to give you an answer. It will take you more time to understand it than find one by yourself.