Exercise 3-1 Revision: Semantic Notions etc. for $\mathcal{ALC}$

a) Given an $\mathcal{ALC}$-term $\varphi$ and an interpretation $I$. Which are the possible relations between $\varphi$ and $I$?

 Proposed Solution:

Two possibilities:

- The interpretation $I$ satisfies the term $\varphi$, written $\exists I \models \varphi$ (i.e. $\varphi^I \neq \emptyset$, and $\varphi^I \subseteq I_D$)
- the interpretation $I$ does not satisfy the term $\varphi$ (i.e. $\varphi^I = \emptyset$)

If $\varphi^I = \emptyset$ for all interpretations $I$, then $\varphi$ is inconsistent/unsatisfiable.
If $\varphi^I = I_D$ for all interpretations $I$, then $\varphi$ is universally valid/tautology.

b) Given an $\mathcal{ALC}$-term $\varphi$ and a T-Box $T$. Which are the possible relations between $\varphi$ and $T$?

 Proposed Solution:

Two possibilities:

- The term $\varphi$ is consistent with the T-Box $T$
  i.e. there is an interpretation $I$ with $I \models T$ (i.e. for all formulae $\tau \in T$ we have $\tau^I = I_D$)
  and $I$ satisfies $\varphi$ ($\varphi^I \neq \emptyset$)
  - Special Case: $\varphi$ entails $T$
    i.e. for every interpretation $I$, satisfying $T (I \models \varphi)$, we have $\varphi^I = I_D$
- the term $\varphi$ is inconsistent with the T-Box $T$
  i.e for every interpretation $I$ with $I \models T$ holds $\varphi^I = \emptyset$

c) Which properties should an inference system like the Tableau Calculus have. What are the properties of the presented Tableau Calculus for $\mathcal{ALC}$?

 Proposed Solution:

One investigates termination, completeness, soundness, space- and time complexity of the procedure.

The presented Tableau Calculus for $\mathcal{ALC}$ and propositional logic terminate, are complete and sound.

The procedure for $\mathcal{ALC}$ has polynomial space complexity and exponential time complexity (in the worst case).
d) List three typical inference problems for $\mathcal{ALC}$ and explain how they can be investigated with the tableau Calculus.

**Proposed Solution:**

- **Consistency test:** is $\varphi$ consistent with the T-Box $T$?
  Method: Normalize $\varphi$ (by expanding acyclically defined concepts) and compute the negation normal form. Apply the Tableau Calculus. If $T$ has cyclic definitions or other axioms, these must be added for every new variable in the tableau. Apply the loop check to ensure termination.

- **Subsumption test:** follows $\varphi \sqsubseteq \psi$ from a T-Box $T$?
  Method: Normalize $\varphi$ and $\neg \psi$ (by expanding acyclically defined concepts) and compute the negation normal form. Apply the Tableau Calculus to $x : NNF(\varphi)$ and $x : NNF(\neg \psi)$. If $T$ has cyclic definitions or other axioms, these must be added for every new variable in the tableau. Apply the loop check to ensure termination. If all branches close then the subsumption test is successful.

- **Instance test:** follows $a : \varphi$ from the T-Box $T$ and the A-Box $A$?
  Method: Normalize $\neg \varphi$ and the A-Box (by expanding acyclically defined concepts) and compute the negation normal form. Start the Tableau Calculus with $a : NNF(\neg \varphi)$. If $T$ has cyclic definitions or other axioms, these must be added for every new variable in the tableau. Apply the loop check to ensure termination. If all branches close then the instance test is successful.

**Exercise 3-2  Tableau for $\mathcal{ALC}$: Concept Definitions and Subsumption**

Given the following T-Box:

- $\text{Hermaphrodites} = \text{Male} \sqcap \text{Female}$
- $\text{Parents-of-Sons-and-Daughters} = \exists\text{has-Child} \text{Male} \sqcap \exists\text{has-Child} \text{Female}$
- $\text{Parents-of-Hermaphrodites} = \exists\text{has-Child} \text{Hermaphrodites}$

Use the Tableau Calculus to check if the following formula is entailed by the T-Box:

$\text{Parents-of-Sons-and-Daughters} \sqsubseteq \text{Parents-of-Hermaphrodites}$

If it is not, give a counter model.

**Proposed Solution:**

a) Expansion:

$\exists\text{has-Child} \text{Male} \sqcap \exists\text{has-Child} \text{Female} \sqsubseteq \exists\text{has-Child} (\text{Male} \sqcap \text{Female})$

b) Reduction to unsatisfiability: Is

$\exists\text{has-Child} \text{Male} \sqcap \exists\text{has-Child} \text{Female} \sqsubseteq \neg (\exists\text{has-Child} (\text{Male} \sqcap \text{Female}))$

unsatisfiable?

c) NNF:

$\exists\text{has-Child} \text{Male} \sqcap \exists\text{has-Child} \text{Female} \sqsubseteq \forall\text{has-Child} (\neg \text{Male} \sqcup \neg \text{Female})$
d) Tableau Calculus

\[
x : \exists h-C \text{ Male} \sqcap \exists h-C \text{ Female} \sqcap \forall h-C (\neg \text{ Male} \sqcup \neg \text{ Female})
\]

\[
x : \exists h-C \text{ Male}
\]

\[
x : \exists h-C \text{ Female}
\]

\[
x : \forall h-C (\neg \text{ Male} \sqcup \neg \text{ Female})
\]

\[
x : h-C y
\]

\[
y : \text{ Male}
\]

\[
y : \neg \text{ Male} \sqcup \neg \text{ Female}
\]

\[
y : \neg \text{ Male}
\]

\[
\bot
\]

\[
y : \neg \text{ Female}
\]

\[
x : h-C z
\]

\[
z : \text{ Female}
\]

\[
z : \neg \text{ Male} \sqcup \neg \text{ Female}
\]

\[
z : \neg \text{ Male}
\]

\[
\bot
\]

We have a counter model. The domain is \(\{x,y,z\}\). \(y\) is Male, \(z\) is Female. \(\{x,y\}\) are the children of \(x\). The subsumption relation does not hold.

**Excercise 3-3  Tableau Calculus with T-Box**

Check with the Tableau Calculus if the T-Box or Exercise 2.4, sheet 2 entails:

a) Train \(\not\sqsubseteq\) Motor_Vehicel

b) Freight_Train \(\sqsubseteq\) Transport_Vehicel

**Proposed Solution:**

a) If Train \(\not\sqsubseteq\) Motor_Vehicel would be entailed by the T-Box then Train \(\sqsubseteq\) Motor_Vehicel would be inconsistent. One could start the tableau with \(x : \neg \text{ Train} \sqcup \text{ Motor_Vehicel}\) and derive the two cases, but finds no contradiction.

Actually one would like to have that Trains and Motor_Vehicels are Disjoint. In order to show this, we could start the tableau with an \(x\) which is both a Train and a Motor_Vehicel and try to find a contradiction.

1  
   \(x : \text{ Train}\)
2  
   \(x : \text{ Motor_Vehicel}\)
3  
   \(x : \text{ Vehicle} \sqcap \forall \text{ drive-on Rail}\) (Def. Train)
4  
   \(x : \text{ Vehicle} \sqcap \forall \text{ powered-by Motor} \sqcap \forall \text{ drive-on Road}\) (Def. Motor_Vehicel)
5  
   \(x : \text{ Vehicle}\) (3)
6  
   \(x : \exists \text{ drive-on Transport_Route}\) (Def. Vehicle)
7  
   \(x : \text{ drive-on } y\) (6)
8  
   \(y : \text{ Transport_Route}\)
9  
   \(y : \text{ Rail}\) (7 and 3,2)
10  
    \(y : \text{ Road}\) (7 and 4,3)
11  
    \(y : \text{ Road} \sqcup \text{ Rail}\)
12  
    \(y : \text{ Road}\)
    \(y : \text{ Rail}\)
We don’t find a contradiction. It becomes clear what is missing. The T-Box does not entail the disjointness of Road and Rail. Therefore there may be things which are Trains and Motor_Vehicles at the same time.

b)

1. \( x : \text{Freight\_Train} \)
2. \( x : \neg\text{Transport\_Vehicles} \)
3. \( x : \text{Train} \sqcap \exists \text{transports\_Freight} \) (Def. Freight\_Train)
4. \( x : \neg\text{Vehicle} \sqcup \neg \exists \text{transports\_Freight} \) (Def. Transport\_Vehicles, \( \neg \) inwards)
5. \( x : \neg\text{Vehicle} \) (4 split, Contradiction with 3,2)
6. \( x : \text{Vehicle} \sqcap \forall \text{drive-on\_Rail} \) (Def. Train)
7. Contradiction with 5
**Exercise 3-4 Recursive Definitions**

Consider the following definition of binary trees ($bT$ stands for binary tree, $lN$ stands for leaf node, $hlB$ stands for has left Branch, $hrB$ stands for has right Branch):

\[ bT = lN \sqcup (\exists hlB bT \sqcap \exists hrB bT) \]

Consider the following tree:

```
K1
  
K2    K3

K4  K5
```

a) Axiomatize the tree as A-Box

b) Prove with the Tableau Calculus that $K_1$ is a binary tree, i.e. $K_1 : bT$

Hint: Use the recursive definition only on demand.

**Proposed Solution:**

a) A-Box:

a) $K_1: \text{hlB } K_2$  e) $K_3: \text{lN}$

b) $K_1: \text{hrB } K_3$  f) $K_4: \text{lN}$

c) $K_2: \text{hlB } K_4$  g) $K_5: \text{lN}$

d) $K_2: \text{hrB } K_5$

b) We do a proof by contradiction and start with $K_1 : \neg bT$: We use the negation of the definition of a binary tree:

\[ \neg bT = \neg lN \sqcap (\forall hlB \neg bT \sqcup \forall hrB \neg bT) \]

\[
\begin{align*}
K_1 : \neg bT \\
K_1 : \neg lN
\end{align*}
\]

\[
\begin{align*}
K_1 : \forall hlB \neg bT & & K_1 : \forall hrB \neg bT \\
K_2 : \neg bT & & K_3 : \neg bT \\
K_2 : \neg lN & & K_3 : \neg lN \\
& & \text{Cont. with e)}
\end{align*}
\]

\[
\begin{align*}
K_2 : \forall hlB \neg bT & & K_2 : \forall hrB \neg bT \\
K_4 : \neg bT & & K_5 : \neg bT \\
K_4 : \neg lN & & K_5 : \neg lN \\
& & \text{Cont. with f)} & & \text{Cont. with g)}
\end{align*}
\]