Exercise 1.2 (WinMoveGame) Betrachten Sie das win-move-Game (Folie 104).

- Ein Knoten ist ein "win-Knoten", wenn derjenige Spieler, der am Zug ist gewinnen kann. Ein Knoten ist ein "lose-Knoten", wenn er verliert. Es wird angenommen, dass jeder Spieler optimal spielt.
- Betrachten Sie ein Spiel, das aus den Kanten (p_1, p_2), (p_2, p_3), ... (p_{n-1}, p_n) besteht. Welche Knoten sind win-Knoten, welche Knoten sind lose-Knoten.
- Beschreiben Sie darauf basierend allgemein, welche Knoten in einem beliebigen Spiel win-Knoten sind, welche lose-Knoten sind, und welche Unentschieden sind.
  - betrachten Sie dabei zuerst nur Abzweige, z.B. ein Spiel, das aus den Kanten (p_1, p_2), (p_2, p_3), ..., (p_{n-1}, p_n) und (p_2, p_{n+1}), (p_{n+1}, p_{n+2}), ..., (p_{n+k-1}, p_{n+k}) besteht.
  - betrachten Sie dann Zyklen, aus denen Pfade hinausführen.
- Wie entsprechen die logischen Quantoren dabei dem Gedanken "gewinnen wollen" bzw. "nicht verlieren wollen" und der Annahme einer optimalen Spielweise beider Teilnehmer?

- Lineare Sequenz: der "letzte" Knoten p_n ist ein Lose-Knoten (wer dort am Zug ist, kann nicht ziehen), der vorletzte ein Win-Knoten und immer weiter abwechselnd.
- Abzweige, keine Zyklen: ein Knoten ist ein win-Knoten, wenn *mindestens ein* Zug zu einem Lose-Knoten führt. Diesen Zug wählt derjenige, der gerade am Zug ist aus, und wird gewinnen. Ansonsten ist der Knoten ein Lose-Knoten. Mit dieser Argumentation kann man in einem nichtzyklischen Graphen alle Knoten entscheiden ("well-founded").
- Mit Zyklen gilt dasselbe: man kann feststellen, dass ein Knoten ein win-Knoten ist, wenn *mindestens ein* Zug zu einem Lose-Knoten führt. Damit lassen sich anhand der Ausgänge aus einem Zyklus *einige* Knoten bestimmen. Darauf aufbauend kann man wieder weitere Win- und Lose-Knoten im Zyklus bestimmen (wieder "well-founded" Argumentation).
- Übrig bleiben Zyklen ohne Win-Ausgänge, in denen man beliebig lange kreiseln kann ("Unentschieden – Draw"), sowie Pfade, die in einen solchen Zyklus hineinführen.
- Win-Knoten sind definiert wie in der logischen Regel:
  [win] $\mathsf{win}(X) \leftarrow \exists Y : (\mathsf{move}(X, Y) \wedge \mathsf{lose}(Y))$.
  Lose-Knoten ergeben sich dann daraus, dass man, egal wohin man zieht, auf einem Win-Knoten (win für den anderen) ankommt:
  $\mathsf{lose}(X) \leftarrow \forall Y : (\mathsf{move}(X, Y) \rightarrow \mathsf{win}(Y))$.
  (Hinweis: die universell quantifizierte Teilformel ist trivialerweise erfüllt, wenn es keine Zugmöglichkeiten gibt).
  Die Formel für lose ist in dieser Form keine logische Regel, da sie einen Allquantor enthält. Analog zu SQL-Anfragen muss man diesen durch eine doppelte Negation und ein Hilfsprädikat ersetzen:
  [nlose] $\mathsf{notlose}(X) \leftarrow \exists Y : (\mathsf{move}(X, Y) \wedge \neg\mathsf{win}(Y))$.
  und
  [lose] $\mathsf{lose}(X) \leftarrow \neg\mathsf{notlose}(X))$.

  Beide Mengen kann man well-founded gegenseitig berechnen (mit T_P bzw. der stratifizierten Semantik geht das nicht, da lose, nlose und win zyklisch negativ voneinander abhängig sind).
  Draw-Nodes sind "der Rest".
  Sie haben eine rekursive Charakterisierung: es gibt keine Züge zu einem Lose-Knoten (damit hätte man gewonnen), aber möglicherweise Züge zu Win-Knoten (wo der andere gewinnt, die man also nicht machen würde), und mindestens einen Zug zu einem Draw-Knoten:
  $\mathsf{draw}(X) \leftarrow (\forall Y : (\mathsf{move}(X, Y) \rightarrow \neg\mathsf{lose}(Y)) \wedge \exists Y : (\mathsf{move}(X, Y) \wedge \mathsf{draw}(Y)))$.
  Man kann dann auch (per Tableau) formal zeigen, dass diese Charakterisierung äquivalent zu "der Rest" ist.

Exercise 1.3 (Quantifiers) Consider the following formulas:

$$\begin{array}{ll}(F_1) & \forall x : (p(x) \to \exists y : q(x,y)) \\ (F_2) & \forall x : \exists y : (p(x) \to q(x,y)) \\ (F_3) & \exists y : \forall x : (p(x) \to q(x,y))\end{array}$$

Check the validity of the following claims:

- $F_1 \models F_2$
- $F_2 \models F_1$
- same for F_1 and F_3
- same for F_2 and F_3

First "think" about them, then use the tableau calculus to prove your claim.

F_1 und F_2: Die Aussagen können z.B. wie folgt gedeutet werden:

- F_1 "jedes Land hat eine Hauptstadt" – bzw. exakter: "für alle x gilt: wenn x ein Land ist, gibt es (für dieses x) ein y das seine Hauptstadt ist".
- F_2 ist damit $\forall x : \exists y : (\mathsf{country}(x) \to \mathsf{cap}(x,y))$
  "für alle x gilt: man kann passend zu jedem x ein y finden, so dass, falls x ein Land ist, y dessen Hauptstadt ist".
  Auf den ersten Blick ist F_2 schwerer zu verstehen (was soll es aussagen für diejenigen x die keine Länder sind?), bedeutet aber das gleiche wie F_1:
  Zu allen x, die ein Land sind, wähle als y die jeweilige Hauptstadt; für diejenigen x, die kein Land sind, kann man y vollkommen beliebig wählen: dann ist $\mathsf{country}(x) \to \mathsf{capital}(x,y)$ trivialerweise erfüllt.

Diskussion: was bedeutet $\models$. $F_1 \models F_2$ ("aus F_1 folgt logisch F_2" bzw. "F_1 logically entails F_2") wenn: "wann immer F_1 erfüllt ist, ist auch F_2 erfüllt".
Formal: für alle $\mathcal{M}$, so dass $\mathcal{M} \models F_1$, gilt auch $\mathcal{M} \models F_2$.

Das wiederum bedeutet: es gibt kein $\mathcal{M}$, so dass darin F_1, aber nicht F_2 erfüllt ist. Oder, äquivalent, die *Implikation* $F_1 \to F_2$ ist *allgemeingültig*.

Beides zeigt man, indem man ein Tableau über $\neg(F_1 \to F_2)$ ansetzt (d.h. ein Modell dafür sucht), und es schließt: dann kann es ein solches Modell nicht geben. Äquivalent zu $\neg(F_1 \to F_2)$ ist ja $F_1 \wedge \neg F_2$, womit man dann das Tableau beginnt.

Tableaux Zu $F_1 \to F_2$: zeige wie oben , dass $F_1 \wedge \neg F_2$ unerfüllbar ist. Dazu kann man erstmal $\neg F_2$ äquivalent umformen:

$$\begin{array}{ll} & \neg\forall x : \exists y : (p(x) \to q(x,y)) \\ \Leftrightarrow & \exists x : \neg\exists y : (p(x) \to q(x,y)) \\ \Leftrightarrow & \exists x : \forall y : \neg(p(x) \to q(x,y)) \\ \Leftrightarrow & \exists x : \forall y : (p(x) \wedge \neg q(x,y))\end{array}$$

$$
\begin{array}{rl}
F_1: & \forall x : (p(x) \rightarrow \exists y : q(x,y)) \quad (1) \\
\neg F_2: & \exists x : \forall y : (p(x) \wedge \neg q(x,y)) \quad (2) \\
& \quad | \\
\text{aus (2):} & \forall y : (p(f_0) \wedge \neg q(f_0,y)) \quad (3) \\
& p(f_0) \wedge \neg q(f_0,Y) \quad (4) \\
& p(f_0) \quad (5) \\
& \neg q(f_0,Y) \quad (6) \\
& \quad | \\
\text{aus (1):} & p(X) \rightarrow \exists y : q(X,y) \quad (7)
\end{array}
$$

$$
\begin{array}{ll}
\neg p(X) \quad (8) & \exists y : q(f_0,X) \quad (9) \\
\quad | & \quad | \\
\square\{X \leftarrow f_0\}(8,5) & q(f_0,f_1(X)) \quad (10) \\
& \quad | \\
& q(f_0,f_1(f_0)) \quad (11) \\
& \quad | \\
& \square\{Y \leftarrow f_1(f_0)\}(6,11)
\end{array}
$$

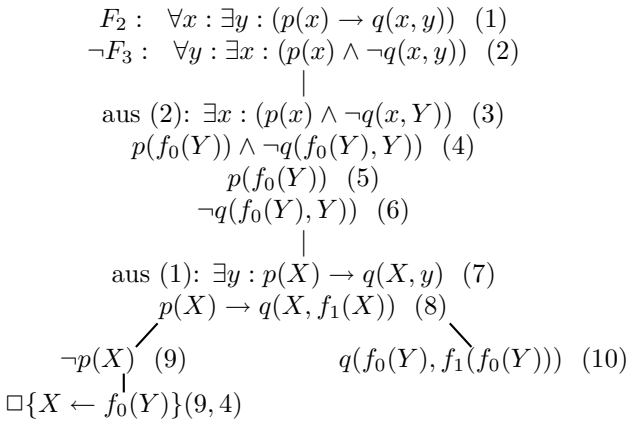
(2) auf (3): das $\exists x$ durch ein nullstelliges neues Funktionssymbol (Konstante) ersetzen, das "dieses eine x" repräsentiert.
(5) und (6) beschreiben, was in dem "vermuteten" Modell "uber f_0 gelten muss.
(8) auf (9): man hätte hier auch gleich $X \mapsto f_0$ ersetzen können, dann würde man später $f_1()$ anstatt $f_1(X)$ setzen können. Die gewählte Variante beschreibt klarer, dass das y abhängig von dem X zu wählen ist.
Abschluss: natürlich ist das $f_1(f_0)$ der "Zeuge", mit dem man die Annahme, dass es f_0 mit $\neg q(f_0,Y)$ gibt, widerlegen kann.

F_2 und F_3: (man hat ja eben gesehen, dass es egal ist, F_1 oder F_2 zu betrachten).
Feststellung: F_2 und F_3 unterscheiden sich nur in der Reihenfolge der Quantoren: "für alle x gibt es ein y" oder "es gibt *ein* y, so dass für alle x ..." – aber genau das macht den Unterschied ob y einmal für alle gemeinsam gewählt wird, oder als Funktion $f(x)$ für jedes X anders sein kann.

Die Aussagen können z.B. wie folgt gedeutet werden: "Jede Person kennt jemanden", "Für alle x gilt: wenn x eine Person ist, kennt sie jemanden", "es gibt jemanden, den alle Personen kennen". Man vermutet, dass F_3 stärker ist als F_2 (d.h. $F_3 \models F_2$, aber nicht $F_2 \models F_3$).

Man kann zeigen, dass $F_3 \wedge \neg F_2$ unerfüllbar ist, und man kann ein Modell für $F_2 \wedge \neg F_3$ angeben. Dazu kann man erstmal $\neg F_3$ äquivalent umformen:

$$
\begin{array}{rl}
& \neg\exists y : \forall x : (p(x) \rightarrow q(x,y)) \\
\Leftrightarrow & \forall y : \neg\forall x : (p(x) \rightarrow q(x,y)) \\
\Leftrightarrow & \forall y : \exists x : \neg(p(x) \rightarrow q(x,y)) \\
\Leftrightarrow & \forall y : \exists x : (p(x) \wedge \neg q(x,y))
\end{array}
$$



Man kann dieses Tableau nicht schliessen, auch wenn man es durch wiederholte Anwendung beliebiger Regeln beliebig expandiert (dies festzustellen ist aber auch eine nichttriviale Aufgabe für automatische Beweiser). (10) beschreibt, wie ein Gegenbeispiel aussieht: für jedes y gibt es ein $f_0(y)$, für das $\neg q(f_0(y), y)$ gilt (d.h., Person $f_0(y)$ kennt y nicht). Dennoch gibt es eine (andere) Person $f_1(f_0(Y)))$, die $f_0(Y)$ kennt.

Die Formeln (6) und (10) kann man nicht gegeneinander verwenden, um das Tableau mit der Substitution $\sigma = \{Y \leftarrow f_1(f_0(Y)))\}$ zu schlieẽsen, da damit zwar mit $\sigma(6) = (10)$, aber *nicht* $\sigma(6) = \sigma(10)$ ist!

Allerdings beschreibt dieser Zusammenhang, dass in dem Modell $f_1(f_0(Y)))$ jemand anders als Y sein muss, da sonst (10) mit (6) für diesen Einzelfall einen Widerspruch bilden würden.

Hinweis: wenn man F_2 statt F_1 nimmt, gibt es keine relevanten Unterschiede: $f_1(X)$ tritt in beiden Fällen nur im rechten Ast auf.