

# Logik in der Informatik

Wintersemester 2024/2025

## Übungsblatt 4

**Abgabe:** bis 18. November 2024, 13.00 Uhr

### Aufgabe 1:

(Moodle-Quiz)

Absolvieren Sie das Quiz 4 auf der Moodle-Plattform.

### Aufgabe 2:

(Präsenzaufgabe)

Da Dinosaurier nie aus der Mode kommen, ist die Dinopark GmbH & Co. KG im 68. Jahrtausend zum intergalaktischen Hit geworden. Um dem Besucheransturm des gesamten Universums gerecht zu werden, wurde für den Park ein unendlich großes Gelände erworben, welches in Gehege  $\langle i, j \rangle$  für alle  $i, j \in \mathbb{Z}$  unterteilt wurde. Dementsprechend benötigen wir für den Nutzungsplan nun auch unendlich viele Aussagensymbole  $B_{i,j}, R_{i,j}, S_{i,j}, T_{i,j}$  für alle  $i, j \in \mathbb{Z}$ .

- (a) Stellen Sie unendliche Formelmengen  $\Phi_1$  und  $\Phi_2$  auf, die die in Aufgabe 2 (a) und (b) von Blatt 2 beschriebenen Bedingungen repräsentieren, allerdings für den neuen Park.
- (b) Warum kann die Bedingung aus Aufgabe 2 (d) von Blatt 2 nicht durch eine unendliche Formelmenge über den gegebenen Aussagensymbolen repräsentiert werden?
- (c) Da auch intergalaktische Kinder schnell quengelig werden, soll auf jedem drei mal drei Gehege großen Abschnitt jede der vier wichtigen Arten vertreten sein. Stellen Sie eine Formelmenge  $\Phi_3$  auf, die diese Bedingung repräsentiert.
- (d) Aufgrund von bestehenden Tierpatenschaften im alten Park sollen gewisse Arten auch im neuen Park im selben Gehege gehalten werden. Sei  $\Phi := \Phi_1 \cup \Phi_2 \cup \Phi_3$  und sei  $\Psi$  die Formelmenge

$$\begin{aligned} \Psi := & \left\{ B_{i,j} : \text{Es gibt eine Patenschaft für Brachiosaurier in Gehege } \langle i, j \rangle \right\} \\ & \cup \left\{ R_{i,j} : \text{Es gibt eine Patenschaft für Raptoren in Gehege } \langle i, j \rangle \right\} \\ & \cup \left\{ S_{i,j} : \text{Es gibt eine Patenschaft für Stegosaurier in Gehege } \langle i, j \rangle \right\} \\ & \cup \left\{ T_{i,j} : \text{Es gibt eine Patenschaft für Tyrannosaurier in Gehege } \langle i, j \rangle \right\}. \end{aligned}$$

Nun stellt sich für das Parkmanagement die Frage, ob es einen Nutzungsplan gibt, der neben den obigen Bedingungen *auch* die Patenschaften respektiert, d.h. ob  $\Phi \cup \Psi$  erfüllbar ist.

Zeigen Sie: Es gibt genau dann einen korrekten Nutzungsplan für den gesamten Park, wenn es für jeden endlichen quadratischen Abschnitt einen solchen Nutzungsplan gibt.

(a) Sei  $n \in \mathbb{N}$  mit  $n \geq 1$  und sei  $\varphi_n$  die in Satz 2.45 der Vorlesung betrachtete aussagenlogische Formel.

(i) Bestimmen Sie alle Interpretationen  $\mathcal{I}$ , für die gilt:

- $\mathcal{I}$  erfüllt  $\varphi_n$  und
- für jedes  $i \in \{1, \dots, n\}$  existiert eine Interpretation, die sich von  $\mathcal{I}$  nur dadurch unterscheidet, dass sie *genau* eines der beiden Aussagensymbole  $X_i, Y_i$  auf einen anderen Wahrheitswert abbildet als  $\mathcal{I}$ , und die  $\varphi_n$  *nicht* erfüllt.

(ii) Beweisen Sie Satz 2.45 der Vorlesung.

*Hinweis:* Eine Möglichkeit, dies zu zeigen, ist einen Beweis durch Widerspruch zu führen. Nehmen Sie dafür an, dass  $\psi_n$  eine zu  $\varphi_n$  äquivalente Formel in DNF ist, die aus weniger als  $2^n$  konjunktiven Klauseln besteht. D.h. es gibt eine natürliche Zahl  $N < 2^n$  und  $N$  konjunktive Klauseln  $\kappa_1, \dots, \kappa_N$ , sodass  $\psi_n = \kappa_1 \vee \dots \vee \kappa_N$ . Folgern Sie aus Aufgabenteil (i), dass mindestens eine der Klauseln  $\kappa_1, \dots, \kappa_N$  von mindestens zwei essenziell verschiedenen Interpretationen  $\mathcal{I}$  aus (i) erfüllt wird. Leiten Sie daraus einen Widerspruch her.

(iii) Gibt es für alle  $n \in \mathbb{N}$  mit  $n \geq 1$  DNF-Formeln  $\varphi'_n$  der Länge  $\mathcal{O}(n)$ , sodass jede zu  $\varphi'_n$  äquivalente KNF-Formel mindestens  $2^n$  disjunktive Klauseln hat? Beweisen Sie, dass Ihre Antwort korrekt ist.

(b) Wandeln Sie analog zu Beispiel 2.52 die Formel

$$\varphi := \left( ((P \vee \neg Q) \wedge S) \rightarrow \neg(\neg Q \vee R) \right)$$

mit dem Tseitin-Verfahren in eine erfüllbarkeitsäquivalente Formel  $\varphi_K$  in 3-KNF um.

**Achtung:** Halten Sie sich strikt an die Art der Notation und Zeilenaufteilung von Beispiel 2.52. Dies beinhaltet folgende Eigenschaften:

- Die Subformeln von  $\varphi$  erhalten aufsteigend Bezeichner der Form  $\psi_i$ , wobei  $i \in \mathbb{N}_{\geq 1}$  ( $\varphi$  erhält keinen zusätzlichen Bezeichner).
- Negierte Aussagensymbole und  $\varphi$  bilden hier keine eigene Subformel. Im Gegensatz dazu bilden aber negierte Formeln, die aus mehr als nur einem Aussagensymbol bestehen, eine eigene Subformel.
- Der Index  $i$  entspricht der Position der Subformeln in der Traversierung des Syntaxbaumes von  $\varphi$  nach der pre-order Tiefensuche. Eine solche Traversierung listet zunächst die Wurzel des Baumes auf, dann die Traversierung des kompletten linken Teilbaumes (so dieser existiert) und dann die des rechten Teilbaumes (so dieser existiert).
- Die neuen Aussagensymbole sind entsprechend aus der Menge  $\{X_\varphi, X_{\psi_1}, X_{\psi_2}, \dots\}$  zu wählen. Für jede Subformel wird in  $\varphi'$  eine neue Zeile begonnen und rechtsseitig die passende Begründung angegeben.
- In  $\varphi_K$  entspricht die Zeilenaufteilung der Zeilenaufteilung von  $\varphi'$ .

Lösungen, die sich nicht an obige Formregeln halten, werden nicht oder nur teilweise korrigiert. Bei fehlerhaften Zeilen in  $\varphi'$  können eventuell die entsprechenden Zeilen in  $\varphi_K$  nicht korrigiert werden.

#### Aufgabe 4:

(20 Punkte)

Lesen Sie Kapitel 4 aus dem Buch “Learn Prolog Now!”.

**Achtung:** Fertigen Sie Ihre Lösung für die Aufgabenteile (a) und (b) handschriftlich an (also wie Aufgabe 3) und reichen Sie diese bei Moodle ein. Die Lösung des Aufgabenteils (c) muss unter Beachtung der bekannten Abgabehinweise (siehe <https://hu.berlin/prolog>) für Prolog-Code in einem zusätzlichen Abgabefach in Moodle eingereicht werden!

(a) Wie antwortet Prolog auf die folgenden Anfragen?

(i) `?- [a, X, a] = [Y, b, Y].`

(ii) `?- [Y, c] = [c, Y | []].`

(iii) `?- [H | T] = [a, b | [c | [d]]].`

(iv) `?- [a | [b | T]] = [X, H | [c | [d]]].`

(b) Das Prädikat `member/2` wird in Abschnitt 4.2 des Buchs “Learn Prolog Now!” definiert. Zeichnen Sie den Suchbaum für die Anfrage

`?- member(a, [X, a, b]).`

(c) Definieren Sie *rekursiv* ein Prädikat `nimm/3`, so dass `nimm(E, X, Y)` genau dann erfolgreich ist, wenn E ein Element der Liste X ist und Y aus der Liste X durch Löschung eines Vorkommens von E entsteht. So sollte beispielsweise die Anfrage

`?- nimm(E, [1, 2, 3], Y).`

zu der folgenden Antworten führen:

```
E = 1,  
Y = [2, 3] ;  
E = 2,  
Y = [1, 3] ;  
E = 3,  
Y = [1, 2] ;  
false.
```