

Gesamtpunktzahl: 30

Abgabe der Lösungen bis zum 23.11.2015

Hinweis: Bitte beachten Sie bei der Bearbeitung der Aufgaben, dass Sie

- Prädikatsdefinitionen immer übersichtlich strukturieren und ausführlich kommentieren,
- in jedem Fall ein Prädikatsschema mit Zusicherungen für die zulässigen Datentypen und den möglichen Instanziierungsvarianten für die einzelnen Argumentpositionen angeben und
- die von Ihnen durchgeführten Tests mit ihren jeweiligen Resultaten dokumentieren und ggf. diskutieren.

Aufgabe 1: Unifikation

3 Punkte

maximale Bearbeitungszeit: 10 Minuten

Unifizieren Sie die folgenden Strukturen und geben Sie die dabei ggf. erzeugten Variablenbindungen an. Begründen Sie, warum die Unifikation erfolgreich ist bzw. scheitert.

$b(x, y)$	$b(Y, X)$
$t(r, i)$	$t(Z, Z)$
$h(g(F, k), g(k, F))$	$h(g(m, H), g(H, m))$
$m(X, c(g), h(X))$	$m(t(r, s), c(u), h(g(T)), t)$
<code>false</code>	<code>not(true)</code>
<code>False</code>	<code>not(true)</code>

Aufgabe 2: PEANO-Arithmetik

12 Punkte

maximale Bearbeitungszeit: 70 Minuten

- Definieren Sie folgende Prädikate auf der Basis der PEANO-Arithmetik und testen Sie sie mit geeigneten Beispielen. Geben Sie immer ein Prädikatsschema an, aus dem die möglichen Instanziierungsvarianten hervorgehen. Begründen Sie die ggf. notwendigen Einschränkungen der Richtungsunabhängigkeit.
 - Ein Prädikat, das den unmittelbaren Nachfolger einer PEANO-Zahl ermittelt.
 - Ein Prädikat, das den unmittelbaren Vorgänger einer PEANO-Zahl ermittelt.

- ein Prädikat für die Subtraktion zweier PEANO-Zahlen.
 - ein Prädikat `verdoppelt(?Peano1,?Peano2)`, das für zwei PEANO-Zahlen prüft, ob `Peano2` doppelt so groß ist, wie `Peano1`.
 - Ein Prädikat `modulo(?Dividend,?Divisor,?Modulo)`, für das gilt

$$\text{Dividend modulo Divisor} = \text{Modulo}.$$
 - Ein Prädikat, das eine PEANO-Zahl in eine Integer-Zahl umwandelt.
Hinweis: Sie können dieses (und das in der Vorlesung angegebene Prädikat `peano2int/2`) gerne zum Testen, nicht aber zum Implementieren der vorangegangenen Peano-Prädikate verwenden.
2. Modifizieren Sie die im Skript angegebenen Prädikatsdefinitionen für `lt/2` und `add/3`, indem Sie Typtests für die Argumentbelegungen hinzufügen. Wie ändert sich das Verhalten? Warum?

Aufgabe 3: Deduktive Datenbanken (3)

4 Punkte

maximale Bearbeitungszeit: 20 Minuten

1. Modifizieren Sie das Prädikat `ist_betroffen_von/2` aus Aufgabenblatt 4, so dass auch die Entfernung zwischen den beiden Orten ermittelt werden kann.
2. Ergänzen Sie die Datenbank um Angaben zur Fließgeschwindigkeit auf den einzelnen Flusstrecken. Definieren Sie ein Prädikat, das die Zeit ermittelt, die zwischen einer Schadstoffeinleitung an einem Ort und der Ankunft der Schadstoffe an einem stromabwärts gelegenen Ort vergeht.

Aufgabe 4: Deduktive Datenbanken (4)

7 Punkte

maximale Bearbeitungszeit: 40 Minuten

Definieren Sie für das in der Datei `dateiverzeichnis.pl` gegebene Dateiverzeichnis aus Aufgabenblatt 3 die folgenden Prädikate:

- Ein Prädikat, das überprüft, ob ein Zugriffspfad zwischen zwei Verzeichnissen existiert.
- Ein Prädikat, das überprüft, ob eine Datei vom Wurzelverzeichnis aus nicht mehr erreichbar ist.
- Ein Prädikat, das zu einem gegebenen Verzeichnis alle seine Unterverzeichnisse, d.h. auch die nur indirekt erreichbaren, ermittelt.

- Ein Prädikat, das die Gesamtgröße aller Dateien in einem Teilbaum des Verzeichnisbaums berechnet.

Aufgabe 5: Deduktive Datenbanken im Vergleich

4 Punkte

maximale Bearbeitungszeit: 20 Minuten

Vergleichen Sie die Prädikatsdefinitionen für die Wegplanung, die Verwandtschaftsbeziehungen, die Vorbesitzer von Immobilien, die Verzeichnisstruktur eines Dateisystems, sowie die Verbindungen in einem Flusssystem.

Welche Gemeinsamkeiten und Unterschiede stellen Sie fest? Welche Bedingungen müssen für terminierungssichere Prädikatsaufrufe gegeben sein? Geben Sie Beispiele für deduktive Datenbanken an, wo diese Bedingungen nicht erfüllt sind.

Bonus assignment: Thinking about recursion (just for fun)

The fundamental idea of recursion becomes increasingly commonplace in our everyday life. Over the years “The Onion”, America’s finest news source, announced several major breakthrough inventions based on the notion of a recursive relationship, see

[http://www.theonion.com/articles/
new-smokable-nicotine-sticks-can-they-help-smokers,538/](http://www.theonion.com/articles/new-smokable-nicotine-sticks-can-they-help-smokers,538/)

[http://www.theonion.com/articles/
new-remote-control-can-be-operated-by-remote,1666/](http://www.theonion.com/articles/new-remote-control-can-be-operated-by-remote,1666/)

[http://www.theonion.com/articles/
apple-unveils-new-productunveiling-product,2162/](http://www.theonion.com/articles/apple-unveils-new-productunveiling-product,2162/)

[http://www.theonion.com/articles/
pizza-huts-new-pizza-lovers-pizza-topped-with-smal,9891/](http://www.theonion.com/articles/pizza-huts-new-pizza-lovers-pizza-topped-with-smal,9891/)

Compare these innovations with recursive relationships as discussed in our course. Pay particular attention to the availability of termination criteria in the different cases.