Universität Paderborn
Prof. Dr. Johannes Blömer
Prof. Dr. Eyke Hüllermeier

2. Dezember 2016

Modellierung – WS 2016/2017

Präsenzübung 7
5. Dezember - 9. Dezember 2016
(Dieser Übungszettel enthält 5 Aufgaben)

Hinweis: In der Präsenzübung haben Sie die Möglichkeit unter Anleitung Ihres Tutors, das Entwickeln von Lösungen zu üben und Ihre Fragen zu klären. Jeder Präsenzübungszettel enthält eine große Auswahl an Aufgaben, von denen ein Teil in der Präsenzübung besprochen wird. Es ist *nicht* das Ziel der Präsenzübung "Musterlösungen" zu verteilen.

Aufgabe 1 (Notationen von Termen)
Geben Sie für die folgenden Terme in Infixdarstellung die passende Baum-, Postfix- und Präfixform an.

1. $\big(a + ((x + 5) * x)\big) - (y + z)$
2. $a + \big((y * (x + z)) - (x + 3)\big)$
3. $((a * b) - (c - d)) + ((5 + x) * (y - c))$

Aufgabe 2 (Abstrakte Algebra)
Die Datenstruktur *Schlange* verwaltet Zahlen in der Reihenfolge ihres Eintreffens. Operationen auf einer Schlange erlauben es, Zahlen hinten anzufügen (*enqueue*) oder vorne die am längsten wartende Zahl zu entnehmen (*dequeue*). Zusätzlich gibt es Operationen, um eine neue leere Schlange zu erzeugen (*createQueue*), die erste Zahl zurückzugeben (*front*) sowie eine Methode, die anzeigt, ob die Schlange leer ist (*empty*).

Die folgende abstrakte Algebra beschreibt eine Schlange (τ, Σ, Q) mit der **Signatur**

$$
\begin{array}{lllrll}
\Sigma & = (\mathcal{S}, \mathcal{F}) \text{ mit} & & & & \\
\mathcal{S} & = \{ & \text{Queue, Nat0, BOOL} \quad \} & & & \\
\mathcal{F} & = \{ & \mathit{createQueue}: & & \to \text{Queue}, & (F_1) \\
 & & \mathit{enqueue}: & \text{Queue} \times \text{Nat0} & \to \text{Queue}, & (F_2) \\
 & & \mathit{dequeue}: & \text{Queue} & \to \text{Queue}, & (F_3) \\
 & & \mathit{front}: & \text{Queue} & \to \text{Nat0}, & (F_4) \\
 & & \mathit{empty}: & \text{Queue} & \to \text{BOOL} \quad \} & (F_5)
\end{array}
$$

Die Sorte Queue stellt eine Schlange dar. Die Sorte Nat0 beschreibt Elemente der Schlange und stellt somit Zahlen dar. Für die Sorte BOOL sind die Konstanten *true* und *false* definiert. *createQueue* ist eine 0-stellige Operation und deshalb eine Konstante.

Gegeben ist die **Menge von Axiomen**

$$
\begin{array}{lll}
Q = \{ & & \\
Q_1: & dequeue(enqueue(createQueue, x)) & \rightarrow createQueue, \\
Q_2: & dequeue(enqueue(enqueue(q, x), y)) & \rightarrow enqueue(dequeue(enqueue(q, x)), y), \\
Q_3: & front(enqueue(createQueue, x)) & \rightarrow x, \\
Q_4: & front(enqueue(enqueue(q, x), y)) & \rightarrow front(enqueue(q, x)), \\
Q_5: & empty(createQueue) & \rightarrow true, \\
Q_6: & empty(enqueue(createQueue, x)) & \rightarrow false \\
\} & &
\end{array}
$$

wobei x, y Variablen der Sorte Nat0 sind und q eine Variable der Sorte Queue ist.
Im Folgenden seien x, y Terme der Sorte Nat0.

1. Prüfen Sie, ob die folgenden Terme entsprechend der Signatur korrekt sind. Zeichnen Sie die Terme als Baum und notieren Sie an jedem Blatt und an jedem inneren Knoten die entsprechende Sorte. Kennzeichnen Sie die Stellen, an denen die Terme nicht korrekt gebildet wurden.
   a) $front\Big(enqueue\Big(createQueue, front(enqueue(createQueue, x))\Big)\Big)$
   b) $empty\Big(dequeue(enqueue(createQueue, x), y)\Big)$
   c) $enqueue\Big(dequeue(createQueue), front(enqueue(createQueue, x))\Big)$
   d) $enqueue(enqueue(createQueue, x), y)$
   e) $front\Big(dequeue(enqueue(createQueue, x))\Big)$
   f) $front(createQueue, x)$
   g) $enqueue(a, createQueue)$

2. Formen Sie die folgenden Terme mit Hilfe der Axiome so um, dass Sie als Ergebnis eine einzelne Variable oder Konstante erhalten. Geben Sie in jedem Schritt das verwendete Axiom an.
   a) $empty\Big(dequeue(enqueue(createQueue, y))\Big)$
   b) $empty\Big(enqueue\Big(createQueue, front(enqueue(enqueue(createQueue, y), x))\Big)\Big)$
   c) $front\Big(enqueue(dequeue(enqueue(createQueue, y)), x)\Big)$
   d) $dequeue(enqueue(createQueue, z))$
   e) $empty(enqueue(createQueue, y))$
   f) $front\Big(dequeue(enqueue(enqueue(createQueue, z), x))\Big)$
   g) $front(enqueue(enqueue(createQueue, x), z))$

3. Erweitern Sie die Algebra um eine zusätzliche Operation *greatest*, die die größte Zahl in der Schlange zurückgibt. Wenn die Schlange beispielsweise die Zahlen (2,3,8,5,7) enthält, soll *greatest* für diese Schlange 8 ausgeben.
   - Nehmen Sie an, dass für die Sorte *Nat0* die Operation $max : Nat0 \times Nat0 \rightarrow Nat0$ gegeben ist.
   - $max(x, y)$ gibt x zurück, falls $x \geq y$. Andernfalls wird y zurückgegeben.

- Sie können davon ausgehen, dass keine Zahl mehrfach in der Schlange vorhanden ist und dass die kleinste Zahl in der Schlange größer als 0 ist.
- Für die Sorte *Nat0* ist eine Konstante *zero* definiert, die dem Wert 0 entspricht.

Geben Sie die nötigen Ergänzungen der Signatur sowie der Axiome an.

Aufgabe 3 (Alternierende Summen)

1. Seien A, B, C, D endliche Mengen. Geben Sie für diese Mengen das Prinzip der Inklusion und Exklusion explizit (also ohne Summenzeichen) an.
2. Berechnen Sie
$$\sum_{k=0}^{n} (-1)^k \binom{n}{k} \ .$$

Aufgabe 4 (Kombinatorik)
Berechnen Sie

1. die Anzahl möglicher Sitzordnungen von n Personen an einem runden Tisch mit n Sitzplätzen.
2. die Anzahl Möglichkeiten, 36 Spielkarten gleichmäßig auf 4 Personen zu verteilen.
3. die Anzahl Möglichkeiten, aus n Kandidaten ein k-köpfiges Komitee, bestehend aus einem Leiter und $k-1$ weiteren Mitgliedern, zusammenzustellen ($k, n \in \mathbb{N}$).
4. Eine Reise durch die Zahlen ist eine Sequenz von Schritten. Ein Schritt ausgehend von einer Zahl k geht entweder nach $k+1$ oder $k-1$. Wir starten bei 0. Wie viele Reisen der Länge n gibt es, die wieder bei 0 enden.
5. die Anzahl Bitstrings der Länge 8, bei denen keine zwei Nullen aufeinander folgen.
6. die Anzahl Bitstrings der Länge 8 mit einer geraden Anzahl von Nullen.

Aufgabe 5 (Permutationen)

1. Notieren Sie die folgenden Permutationen als Ordnung und Zyklus:
   a) $\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 1 & 4 & 5 & 6 \end{bmatrix}$
   b) $\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 5 & 1 & 4 & 6 & 3 \end{bmatrix}$
   c) $\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 1 & 6 & 5 & 2 & 4 \end{bmatrix}$
2. Notieren sie die folgenden Zyklen als Permutation:
   a) (2 5 3)(4 1 6)
   b) (6)(1)(2)(5)(3)(4)
   c) (2 3 1 5 6 4)
3. Wir betrachten die symetrische Gruppe $\mathfrak{S}_n$ mit $n = 3$. Bestimmen Sie mit Hilfe der Stirlingzahlen erster Art die Anzahl der Permutationen mit $k = 1, 2$ und 3 Zyklen. Notieren Sie jeden einzelnen Schritt.