Diskrete Mathematik und Logik

Universität Konstanz, Wintersemester 2022/23

Dozent: Prof. Dr. Sven Kosub

Ausarbeitung: Dr. Matthias Droth

Inhaltsverzeichnis

Kapitel 1

Mathematische Grundlagen

1.1 Zuweisung

- Zuweisung als Standardform (linke Seite wird durch rechte Seite definiert):

$$x =_{\text{def}} y \quad \text{oder} \quad X := y\,. \tag{1.1}$$

  $\Rightarrow$ „x“ ist der Name für y.

- x und y dürfen beliebig vertauscht werden.

Beispiele:

1. $x =_{\text{def}} 2$,
2. $x =_{\text{def}} 2n + 1$,
3. $f(x) =_{\text{def}} 2n + 1$,
4. $p|q \Leftrightarrow_{\text{def}} \exists k :\ q = k \cdot p$ (es gibt ein k mit $q = k \cdot p$).

Beachte: „$x = y$“ **behauptet** eine Gleichheit $\Rightarrow$ Beweis nötig!

1.2 Iteration

- Definitionsform zum Ausdrücken von Wiederholungen in Variablen, aber mit bestimmten Grenzen:

$$\sum_{k=1}^{n} a_k =_{\text{def}} a_1 + a_2 + a_3 + \ldots + a_n\,, \quad \prod_{k=1}^{n} a_k = a_1 \cdot a_2 \cdot \ldots \cdot a_n\,. \tag{1.2}$$

Beispiel: $n! =_{\text{def}} \prod_{k=1}^{n} k\,.$

- Typisches Problem: Finde wertgleichen Ausdruck onhe die Laufvariable k.
  Besipiel: $\sum_{k=1}^{n} k = \frac{n\cdot(n+1)}{2}$.

1.3 Rekursion

- Definitionsform, bei der die definierte Seite auf der definierenden Seite vorkommen darf: $x =_{\text{def}} \ldots x \ldots$.

- Um unendliche Schachtelungen auszuschließen werden Abbruchbedingungen festgelegt. Ein paar Beispiele:

  1. $n! =_{\text{def}} n \cdot (n-1)!$ für $n \geq 1$ und $0! =_{\text{def}} 1$.
  2. $\text{Euklid}(m,\, n) =_{\text{def}} \begin{cases} \text{Euklid}\;(\text{mod(n, m), m}) & \text{falls } m \nmid n, \\ m & \text{falls } m \mid n. \end{cases}$
  3. Fibonacci Reihe: $F_n =_{\text{def}} F_{n-1} + F_{n-2}$ für $n \geq 2$ und mit $F_0 =_{\text{def}} 0$, $F_1 =_{\text{def}} 1$.

$$\begin{aligned} \Rightarrow F_5 &= F_4 + F_3 \\ &= (F_3 + F_2) + (F_2 + F_1) \\ &= ((F_2 + F_1) + (F_1 + F_0)) + ((F_1 + F_0) + F_1) \\ &= (((F_1 + F_0) + F_1) + (F_1 + F_0)) + ((F_1 + F_0) + F_1) \\ &= 5F_1 + 3F_0 = 5 \cdot 1 + 3 \cdot 0 = 5\,. \end{aligned} \tag{1.3}$$

  Merke: Rekursionen werden ausgewertet, indem sie in Iterationen umgewandelt werden.

  4. Ackermann Funktion (auf natürlichen Zahlen x, y):

$$\begin{aligned} A(0,\, y) &=_{\text{def}} y + 1\,, \\ A(x,\, 0) &=_{\text{def}} x\,, \\ A(x,\, y) &=_{\text{def}} A\left(x-1,\, A(x,\, y-1)\right)\,, \quad \text{für } x \geq 1,\, y \geq 1. \end{aligned} \tag{1.4}$$

  5. Typische Probleme: Terminierung, Auflösung (n-te Fibonacci-Zahl) und Abschätzung bei rekursiven Definitionen.

  Beispiele:

1. $\left(\frac{n}{2}\right)^{\frac{n}{2}} \leq n! \leq n^2$

2. Ackermann-Funktion:

$$
\begin{aligned}
A(5,y) &\geq 2^{2^{\cdot^{\cdot^{\cdot^{2}}}}}\Big\}\, y\text{-mal} \\
A(5,2) &\geq 2^2 = 4 \\
A(5,3) &\geq 2^{2^2} = 2^4 = 16 \\
A(5,4) &\geq 2^{2^{2^2}} = 2^{16} = 65536 \\
A(5,5) &\geq 2^{2^{2^{2^2}}} = 2^{65536} \approx 10^1 9660
\end{aligned}
$$

Die Ackermann Funktion ist nicht mit einer for-Schleife berechenbar.

1.4 Strukturierte Induktion

- Beim rekursiven Definieren geht es ums **Zerlegen** und beim induktiven Definieren geht es ums **Zusammensetzen**.
- Tpyisch für Konstruktionen von Mengen.
- Form:
  1. (IA) Festlegen der Basiselemente.
  2. (IS) Festlegen der Operationen zur Konstruktion neuer Elemente aus bestehenden Elementen.
  3. Nichts sonst ist ein Element dieser Menge.

Beispele:

1. Natürliche Zahlen:
   - 0 ist eine natürliche Zahl.
   - Ist n eine natürliche Zahl, so ist $n+1$ auch eine natürliche Zahl.
   - Nichts sonst ist eine natürliche Zahl.
2. Korrekte Klammerausdrücke
   (a) ( ) ist ein korrekter Klammerausdruck.
   (b) Sind H_1, H_2 korrekte Klammerausdrücke, so sind (H_1) und $H_1 H_2$ korrekte Klammerausdrücke.
   (c) Nichts sonst ist ein korrekter Klammerausdruck.

Beispiel: „$(\underbrace{(())()}_{\substack{(())\,() \\ \downarrow \\ ()}})$“ ist ein korrekter Klammerausdruck, „)(“ ist kein korrekter Klammerausdruck. Fallbeispiel: Suchbäume

- Datenstruktur zur Suche in geordneten Mengen
- wollen wisse, ob $1, 2, 3, 5, 7, 11, 13, 17, 19$ eine Zahl a enthält.
- Die kombinatorische Struktur: volle, gewurzelte Binärbäume.
- voller, gewurzelter Binärbaum T besteht aus Konten $(0, \square)$ und Verweise $(\rightarrow)$. Ein Knoten r ist die Wurzel.
  (IA) Für jedes r ist der KNoten r ein Baum, $\square$
  (IS) Soind T_1, T_2 Bäume mit Wurzeln r_1, r_2 und keinen gemeinsamen Knoten, so ist die Kollektion der Knoten und Kanten aus T_1 und T_2 sowie den neuen Kanten $r \rightarrow r_1$, $r \rightarrow r_2$ für $r \neq r_1$, r_2 ein Baum; r ist die neuer Wurzel.
  Nichts sonst ist ein voller, gewurzelter Binärbaum.
  Beispiel: XXX image2 XXX
- Wollen die induktive Struktur der Bäume benutzen, um Sachverhalte zu beweisen.
- Knoten ohne ausgehende Kantne heißt *Blatt*, sonst heißt der Knoten *innerer Knoten*.

Proposition
Für einen vollen, gewurzelten Binärbaum T seien n_T die Anzahl der inneren Knoten und m_T die Anzahl der Blätter. Dann gilt stets $nT = m_T - 1$.

Beweis (induktion über Aufbau der Bäume):

(IA) Ist T ein Baum mit einem Knoten τ, so gilt $m_\tau = 1$, $n_\tau = 0$.
(IS) Es sei T ein Baum mit mehr als einem Knoten, mit Wurzel r. Dann gibt es Bäume T_1, T_2 mit Wurzeln τ_1, τ_2, aus denen T zusammengesetzt ist.
Insbesondere:Blätter bzw. innere Knoten von τ_1, τ_2 sind auch Blätter bzw. innere Knoten von T. Die Wurzel r ist ein innerer Knoten von T.
Es gilt:

$$\begin{aligned} n_T &= n_{T_1} + n_{T_2} + 1 \\ &= (m_{T_1} - 1) + (m_{T_2} - 1) + 1 \\ &= (m_{T_1} + m_{T_2}) - 1 \\ &= m_T - 1\,. \end{aligned} \tag{1.5}$$

Literaturverzeichnis

[1] G. Baym, *Lectures on Quantum Mechanics* (Addison Wesley, 1993).

[2] F. Schwabl, *Quantenmechanik* und *Quantenmechanik für Fortgeschrittene* (Springer, 2007 und 2005).

[3] W. Nolting, *Grundkurs Theoretische Physik*, Bände 4, 5/2 und 7 (Springer, 2005, 2006 und 2005).

[4] A. Messiah, *Quantenmechanik 2* (de Gruyter, 1990).

[5] H. Bruus and K. Flensberg, *Many-Body Quantum Theory in Condensed Matter Physics: An Introduction* (Oxford, 2004).