

Grundbegriffe der Informatik

Tutorium 1 - 6. Sitzung

Dennis Felsing

dennis.felsing@student.kit.edu

http://www.stud.uni-karlsruhe.de/~ubcqr/2010w/tut_gbi/

2010-11-29



Überblick

1 Relationen

- Produkt
- Reflexivität und Transitivität
- Aufgabe Beweis

2 Codierungen

Relationen

Definitionen

Seien $R \subseteq M_1 \times M_2$ und $S \subseteq M_2 \times M_3$.

Produkt $S \circ R = \{(x, z) \in M_1 \times M_3 \mid \exists y \in M_2 : (x, y) \in R \wedge (y, z) \in S\}$

assoziativ: $(R \circ S) \circ T = R \circ (S \circ T)$

Beispiel

$R = \{(1, 2), (1, 3), (2, 1)\}$, $S = \{(1, 1), (2, 3), (3, 1)\}$

$S \circ R = \{(1, 3), (1, 1), (2, 1)\}$

$R \circ S = \{(1, 2), (1, 3), (3, 2), (3, 3)\}$

Achtung

Reihenfolge spielt eine Rolle!

Relationen

Definitionen

Seien $R \subseteq M_1 \times M_2$ und $S \subseteq M_2 \times M_3$.

Produkt $S \circ R = \{(x, z) \in M_1 \times M_3 \mid \exists y \in M_2 : (x, y) \in R \wedge (y, z) \in S\}$

assoziativ: $(R \circ S) \circ T = R \circ (S \circ T)$

Beispiel Verwandtschaft

$xRy \Leftrightarrow x$ ist Elternteil von y

$xSy \Leftrightarrow x$ ist verheiratet mit y

$x(S \circ R)y \Leftrightarrow x$ ist Schwiegermutter/-vater von y

$x(R \circ S)y \Leftrightarrow x$ ist Ehepartner des Elternteils von y

Beispiel Funktion

$f(x) = 3x, g(x) = x + 1, (f \circ g)(x) = f(g(x)) = 3(x + 1)$

Relationen

Definitionen

Produkt $S \circ R = \{(x, z) \in M_1 \times M_3 \mid \exists y \in M_2 : (x, y) \in R \wedge (y, z) \in S\}$

assoziativ: $(R \circ S) \circ T = R \circ (S \circ T)$

Identität $I_M = \{(x, x) \mid x \in M\}$

$R \circ I_M = R$

$I_M \circ R = R$

Potenz $R^0 = I_M$

$\forall i \in \mathbb{N}_0 : R^{i+1} = R \circ R^i$

Reflexiv-transitive Hülle $R^* = \bigcup_{i=0}^{\infty} R^i$

Kleinste Relation, die R enthält und reflexiv und transitiv ist.

Reflexivität und Transitivität

Definitionen

Reflexiv : $I_M \subseteq R$

Transitiv : $\forall x, y, z \in M : xRy \wedge yRz \Rightarrow xRz$

Beispiele

- Gleichheitsrelation $R_= \subseteq \mathbb{R} \times \mathbb{R}$ ist reflexiv und transitiv.
- $\{(a, a), (a, b), (b, b), (b, c), (c, c)\} \subseteq \{a, b, c\} \times \{a, b, c\}$ ist reflexiv und nicht transitiv.
- $R_{\leq} \subseteq \mathbb{R} \times \mathbb{R}$ ist reflexiv und transitiv.
- $R_{<} \subseteq \mathbb{R} \times \mathbb{R}$ ist transitiv und nicht reflexiv.
- $xVy \Leftrightarrow x$ ist Vorfahre von y . V ist transitiv und nicht reflexiv.
- $x \heartsuit y \Leftrightarrow x$ liebt y . \heartsuit ist nicht transitiv und nicht reflexiv.
- $R \subseteq \{\} \times \{\}$ ist transitiv und reflexiv.

Aufgabe

Seien $R \subseteq M \times M$ eine beliebige Relation
und $S \subseteq M \times M$ eine reflexive und transitive Relation.

Behauptung: $R \subseteq S \Rightarrow R^* \subseteq S$

Beweis: **Vollständige Induktion**: $\forall i \in \mathbb{N}_0 : R \subseteq S \Rightarrow R^i \subseteq S$

Induktionsanfang : $i = 0$: $R^0 = I_M$. Da S reflexiv ist, gilt $I_M \subseteq S$
und somit $R^0 \subseteq S$.

Induktionsvoraussetzung : Für ein beliebiges aber festes $n \in \mathbb{N}_0$
gelte $R \subseteq S \Rightarrow R^n \subseteq S$.

Induktionsschluss : $n \rightsquigarrow n + 1$: $R^{n+1} = R \circ R^n$.

Nach IV gilt $R^n \subseteq S$, also ist $R \circ R^n \subseteq R \circ S$. Nach
Annahme gilt $R \subseteq S$. Somit ist $R \circ S \subseteq S \circ S$.

(Ausführlicher auf Ü-Blatt 6.1 b) Da S transitiv ist,
gilt $S \circ S = S$ \square

Überblick

1 Relationen

2 Codierungen

- Dezimaldarstellung von Zahlen
- Andere Zahldarstellungen
- Aufgabe Algorithmus
- Übersetzungen
- Homomorphismen

Codierungen

Definition

Dezimaldarstellung von Zahlen

$$Num_{10}(\varepsilon) = 0$$

$$\forall w \in Z_{10}^* \forall x \in Z_{10} : Num_{10}(wx) = 10 \cdot Num_{10}(w) + num_{10}(x)$$

Beispiel

$$Num_{10}(479) = \dots = 10 \cdot (10 \cdot (10 \cdot Num_{10}(\varepsilon) + 4) + 7) + 9 = 479$$

Wohldefiniertheit von Num_{10}

Behauptung

Num_{10} ist wohldefiniert, das heißt für jedes Wort $w \in Z^*$ wird eindeutig der Funktionswert Num_{10} festgelegt.

Beweis

Vollständige Induktion über Wortlänge n :

Induktionsanfang : $n = 0$: $|w| = 0$, also $w = \varepsilon$. Nach Definition gilt $Num_{10}(\varepsilon) = 0$. Andere Werte für $Num_{10}(\varepsilon)$ existieren nicht, da $Num_{10}(wx)$ ein wx der Länge ≥ 1 verlangt.

Induktionsvoraussetzung : Für ein beliebiges aber festes $n \in \mathbb{N}_0$ gelte: Für alle Wörter $w \in Z^n$ ist $Num_{10}(w)$ eindeutig definiert.

Wohldefiniertheit von Num_{10}

Beweis (fortgesetzt)

Induktionsschluss : $n \rightsquigarrow n + 1$: Sei w' ein beliebiges Wort mit $|w'| = n + 1$. w' lässt sich schreiben als $w' = wx$ mit $w \in Z^n$ und $x \in Z$.

In der Definition von Num_{10} passt nur die zweite Gleichung: $Num_{10}(w') = Num_{10}(wx) = 10 \cdot Num_{10}(w) + num_{10}(x)$.

Nach IV ist $Num_{10}(w)$ eindeutig festgelegt, also auch $Num_{10}(w')$. \square

Andere Zahldarstellungen

Definition

Wie allgemeine Umwandlung bei Basis k ?

$$Num_k(\varepsilon) = 0$$

$$\forall w \in Z_k^* \forall x \in Z_k : Num_k(wx) = k \cdot Num_k(w) + num_k(x)$$

Beispiele

$$Num_3(\mathbf{111}) = \dots = 13$$

$$Num_2(\mathbf{1}) = 1, Num_2(\mathbf{11}) = 3, Num_2(\mathbf{111}) = 7, Num_2(\mathbf{1111}) = 15$$

$$\text{Allgemeiner: } \forall m \in \mathbb{N}_0 : Num_2(\mathbf{1}^m) = 2^m - 1$$

$$\text{Für Basis } k = 3: \forall m \in \mathbb{N}_0 : Num_3(\mathbf{2}^m) = 3^m - 1$$

Aufgabe Algorithmus

Gib einen Algorithmus an, der eine Binärdarstellung zu einer Zahl umwandelt.

```
//Eingabe :  $w \in Z_2^*$   
 $x \leftarrow 0$   
 $v \leftarrow \varepsilon$   
for  $i \leftarrow 0$  to  $|w| - 1$  do  
     $v \leftarrow v \cdot w(i)$   
     $x \leftarrow 2x + num_2(w(i))$   
od  
//am Ende:  $v = w \wedge x = Num_2(w)$ 
```

Wie lautet eine sinnvolle Schleifeninvariante? $x = Num_2(v)$

Übersetzungen

Definition

Seien A und B Alphabete und $sem_A : L_A \rightarrow Sem$ und $sem_B : L_B \rightarrow Sem$ Abbildungen von formalen Sprachen $L_A \subseteq A^*$ und $L_B \subseteq B^*$ in die Menge Sem von Bedeutungen.

Eine Abbildung $f : L_A \rightarrow L_B$ heißt **Übersetzung** bezüglich sem_A und sem_B , wenn f **die Bedeutung erhält**, d.h.

$\forall w \in L_A : sem_A(w) = sem_B(f(w))$.

Wenn f injektiv ist, heißt es **Codierung**. Für $w \in L_A$ heißt $f(w)$ dann ein **Codewort** und die Menge $\{f(w) \mid w \in L_A\}$ aller Codewörter heißt **Code**.

Übersetzungen

Warum macht man Übersetzungen?

- Lesbarkeit
- Kompression
- Verschlüsselung
- Fehlererkennung und Fehlerkorrektur

Homomorphismen

Definition

Seien A, B Alphabete und $h : A \rightarrow B^*$ eine Abbildung. Ein **Homomorphismus** $h^{**} : A^* \rightarrow B^*$ mit:

$$h^{**}(\varepsilon) = \varepsilon$$
$$\forall w \in A^* \forall x \in A : h^{**}(wx) = h^{**}(w)h(x)$$

Häufig schreibt man h statt h^{**} .

Beispiel

$h(a) = 001$ und $h(b) = 1101$

$h^{**}(bba) = h(bba) = h(b)h(b)h(a) = 1101 \cdot 1101 \cdot 001 = 11011101001$

Homomorphismen

Definitionen

Ein Homomorphismus heißt ε -frei, wenn für alle $x \in A$ gilt:

$$h(x) \neq \varepsilon.$$

Was bringt ε -Freiheit?

Sonst gehen „Informationen verloren“.

Beispiel

$$h(a) = 001 \text{ und } h(b) = \varepsilon$$

$$h(w) = 001. \text{ Was war } w?$$

Ein Wort aus b^*ab^* , genaueres wissen wir nicht. Also ist h nicht injektiv und somit keine Codierung.

Homomorphismen

Information kann auch anders verloren gehen:

Beispiele

- $h(\mathbf{a}) = 0, h(\mathbf{b}) = 1, h(\mathbf{c}) = 10$
- $h(\mathbf{a}) = h(\mathbf{b})$

Definition

„Information geht verloren“ \Leftrightarrow Es gibt Wörter $w_1 \neq w_2$ mit $h^{**}(w_1) = h^{**}(w_2)$.

Homomorphismen

Definition

Präfixfreier Code : für keine zwei verschiedenen Symbole $x_1, x_2 \in A$ gilt: $h(x_1)$ ist Präfix von $h(x_2)$.

Beispiele

$h(a) = 001$ und $h(b) = 1101$ ist präfixfrei.

$h(a) = 01$ und $h(b) = 011$ ist nicht präfixfrei.

Wenn alle $h(x)$ gleich lang sind: präfixfrei $\Leftrightarrow h$ injektiv

Überblick

1 Relationen

- Produkt
- Reflexivität und Transitivität
- Aufgabe Beweis

2 Codierungen

- Dezimaldarstellung von Zahlen
- Andere Zahldarstellungen
- Aufgabe Algorithmus
- Übersetzungen
- Homomorphismen

pbfcomics.com

