Grundbegriffe der Informatik

Kapitel 4: Wörter

Thomas Worsch

KIT, Institut für Theoretische Informatik

Wintersemester 2015/2016

Ein Wort über einem Alphabet A ist eine Folge von Zeichen aus A.

`Apfelmus`

Ein Wort über einem Alphabet A ist eine Folge von Zeichen aus A.

`Milchreis`

Symbole dürfen mehrfach vorkommen.

Wo sind wir?

Definition des Begriffs „Wort“

Das leere Wort

Konkatenation von Wörtern

Eigenschaften der Konkatenation

Beispiel: Aufbau von E-Mails

Iterierte Konkatenation

Formale Sprachen

Binäre Operationen

Das Leerzeichen spielt bei uns (fast) keine Rolle.

heutzutage überall und ständig

- jedenfalls z. B. in europäischen Schriften

früher nicht!

Für uns ein Zeichen wie alle anderen auch

- manchmal explizit ␣ geschrieben

Also ist `Hallo␣Welt`

- *eine* Folge von Zeichen
- *ein* Wort (und nicht zwei)

Wort — formal definiert als Liste von Zeichen

Sinn der Übung

- an harmlosem Beispiel Wichtiges üben
- nicht: Einfaches möglichst kompliziert darstellen

Wort — formal definiert als Liste von Zeichen

Sinn der Übung

- an harmlosem Beispiel Wichtiges üben
- nicht: Einfaches möglichst kompliziert darstellen

wesentlich an einer „Folge“ oder „Liste“ (von Zeichen)?

Wort — formal definiert als Liste von Zeichen

Sinn der Übung

- an harmlosem Beispiel Wichtiges üben
- nicht: Einfaches möglichst kompliziert darstellen

wesentlich an einer „Folge“ oder „Liste“ (von Zeichen)?

Reihenfolge, deutlich gemacht durch Nummerierung:

0	1	2	3	4	5	6	7	8
`M`	`i`	`l`	`c`	`h`	`r`	`e`	`i`	`s`

Eine formale Definition von Wörtern

definiere für $n \in \mathbb{N}_0$ Menge der n kleinsten nichtnegativen ganzen Zahlen

- $\mathbb{Z}_n = \{i \in \mathbb{N}_0 \mid 0 \leq i \text{ und } i < n\}$
- Beispiele: $\mathbb{Z}_4 = \{0, 1, 2, 3\}$, $\mathbb{Z}_1 = \{0\}$ und $\mathbb{Z}_0 = \{\}$

Ein *Wort* (über dem Alphabet A)
ist eine *surjektive* Abbildung $w : \mathbb{Z}_n \to B$ mit $B \subseteq A$.

n heißt die *Länge eines Wortes*, geschrieben $|w|$

- erst mal nur an $n \geq 1$ denken, das leere Wort kommt später

Eine formale Definition von Wörtern

definiere für $n \in \mathbb{N}_0$ Menge der n kleinsten nichtnegativen ganzen Zahlen

- $\mathbb{Z}_n = \{i \in \mathbb{N}_0 \mid 0 \leq i \text{ und } i < n\}$
- Beispiele: $\mathbb{Z}_4 = \{0, 1, 2, 3\}$, $\mathbb{Z}_1 = \{0\}$ und $\mathbb{Z}_0 = \{\}$

Ein *Wort* (über dem Alphabet A)
ist eine *surjektive* Abbildung $w : \mathbb{Z}_n \to B$ mit $B \subseteq A$.

n heißt die *Länge eines Wortes*, geschrieben $|w|$

- erst mal nur an $n \geq 1$ denken, das leere Wort kommt später

Beispiel:

- $w = \texttt{hallo}$
- formal $w : \mathbb{Z}_5 \to \{\texttt{a}, \texttt{h}, \texttt{l}, \texttt{o}\}$ mit
  $w(0) = \texttt{h}$, $w(1) = \texttt{a}$, $w(2) = \texttt{l}$, $w(3) = \texttt{l}$ und $w(4) = \texttt{o}$.

Menge aller Wörter

A^*: *Menge aller Wörter* über Alphabet A

- alle Wörter, die nur Zeichen aus A enthalten

Beispiel: $A = \{\mathtt{a}, \mathtt{b}\}$.
Dann enthält A^* die Wörter

- a und b
- aa, ab, ba und bb
- aaa, aab, aba, abb, baa, bab, bba und bbb
- ...
- und außerdem ε, das *leere Wort*

unendlich viele Wörter, die alle *endliche* Länge haben!

A^* formalistisch: die Menge aller surjektiven Abbildungen $w : \mathbb{Z}_n \to B$ mit $n \in \mathbb{N}_0$ und $B \subseteq A$.

Wo sind wir?

Definition des Begriffs „Wort“

Das leere Wort

Konkatenation von Wörtern

Eigenschaften der Konkatenation

Beispiel: Aufbau von E-Mails

Iterierte Konkatenation

Formale Sprachen

Binäre Operationen

Das leere Wort

Zählen

- erst: ab Eins
- später: die Null ist auch nützlich

Das leere Wort

Zählen

- erst: ab Eins
- später: die Null ist auch nützlich

Analogon bei Wörtern: das *leere Wort*

- Es besteht aus 0 Symbolen.
- wir schreiben ε

Das leere Wort

Zählen

- erst: ab Eins
- später: die Null ist auch nützlich

Analogon bei Wörtern: das *leere Wort*

- Es besteht aus 0 Symbolen.
- wir schreiben ε

hilft Formalismus?

$$\varepsilon : \mathbb{Z}_0 \to \{\} \qquad \text{also} \qquad \varepsilon : \{\} \to \{\}$$

- nur eine Relation $R \subseteq \{\} \times \{\} = \{\}$, also $R = \{\} \times \{\} = \{\}$
- linkstotal, rechtstotal, rechtseindeutig
- also ein Wort
- *Das* leere Wort ist eindeutig.

Das leere Wort als Element von Mengen

Das leere Wort ist „etwas".

Die Kardinalität der Menge $\{\varepsilon, \mathtt{abaa}, \mathtt{bbbababb}\}$ ist

$$|\{\varepsilon, \mathtt{abaa}, \mathtt{bbbababb}\}| = 3$$

Die Kardinalität der Menge $\{\varepsilon\}$ ist

$$|\{\varepsilon\}| = 1$$

Das ist nicht die leere Menge!

Die Kardinalität der Menge $\{\}$ ist

$$|\{\}| = 0$$

Das ist die leere Menge.

A^n — die Wörter der Länge n

A^n: *Menge aller Wörter der Länge n* über dem Alphabet A

Beispiel $A = \{\mathtt{a}, \mathtt{b}\}$

- $A^0 = \{\varepsilon\}$
- $A^1 = \{\mathtt{a}, \mathtt{b}\}$
- $A^2 = \{\mathtt{aa}, \mathtt{ab}, \mathtt{ba}, \mathtt{bb}\}$
- $A^3 = \{\mathtt{aaa}, \mathtt{aab}, \mathtt{aba}, \mathtt{abb}, \mathtt{baa}, \mathtt{bab}, \mathtt{bba}, \mathtt{bbb}\}$

Also ist sozusagen

$$A^* = A^0 \cup A^1 \cup A^2 \cup A^3 \cup \cdots$$

bessere Schreibweise ohne Pünktchen:

$$A^* = \bigcup_{i \in \mathbb{N}_0} A^i$$

Pünktchen vermeiden

Sie erinnern sich?

$$\bigcup_{i \in \mathbb{N}_0} A^i = \{w \mid \text{es gibt ein } i \text{ mit: } w \in A^i\}$$

Wo sind wir?

Definition des Begriffs „Wort“

Konkatenation von Wörtern

- Konkatenation mit dem leeren Wort

Eigenschaften der Konkatenation

Beispiel: Aufbau von E-Mails

Iterierte Konkatenation

Formale Sprachen

Binäre Operationen

Konkatenation — anschaulich: Wörter werden hintereinander geschrieben

Operationssymbol üblicherweise der Punkt „·“,

- wie bei Multiplikation manchmal weggelassen

Beispiele

$$\mathtt{SCHRANK} \cdot \mathtt{SCHLÜSSEL} = \mathtt{SCHRANKSCHLÜSSEL}$$
$$\mathtt{SCHLÜSSEL} \cdot \mathtt{SCHRANK} = \mathtt{SCHLÜSSELSCHRANK}$$

Reihenfolge!

$$\mathtt{SCHRANKSCHLÜSSEL} \neq \mathtt{SCHLÜSSELSCHRANK}$$

Konkatenation — formal:
aus zwei surjektiven Abbildungen wird eine dritte

Beispiel

$$
\begin{array}{cccccccccc}
 & 0 & 1 & 2 & 3 & 4 & & 0 & 1 & 2 \\
 & \mathtt{A} & \mathtt{P} & \mathtt{F} & \mathtt{E} & \mathtt{L} & \cdot & \mathtt{M} & \mathtt{U} & \mathtt{S}
\end{array}
$$

$$
\begin{array}{ccccccccc}
 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
= & \mathtt{A} & \mathtt{P} & \mathtt{F} & \mathtt{E} & \mathtt{L} & \mathtt{M} & \mathtt{U} & \mathtt{S}
\end{array}
$$

Konkatenation — formal:
aus zwei surjektiven Abbildungen wird eine dritte

Beispiel

$$\begin{array}{ccccccccc} 0 & 1 & 2 & 3 & 4 & & 0 & 1 & 2 \\ \mathtt{A} & \mathtt{P} & \mathtt{F} & \mathtt{E} & \mathtt{L} & \cdot & \mathtt{M} & \mathtt{U} & \mathtt{S} \end{array}$$

$$\begin{array}{ccccccccc} & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ = & \mathtt{A} & \mathtt{P} & \mathtt{F} & \mathtt{E} & \mathtt{L} & \mathtt{M} & \mathtt{U} & \mathtt{S} \end{array}$$

definiere für $w_1 : \mathbb{Z}_m \to A_1$ und $w_2 : \mathbb{Z}_n \to A_2$

$$w_1 \cdot w_2 : \mathbb{Z}_{m+n} \to A_1 \cup A_2$$

$$i \mapsto \begin{cases} w_1(i) & \text{falls } 0 \leq i < m \\ w_2(i-m) & \text{falls } m \leq i < m+n \end{cases}$$

übrigens: $|w_1 \cdot w_2| = |w_1| + |w_2|$

Umgehen mit Definitionen — am Beispiel Konkatenation

Definition
Es seien $w_1 : \mathbb{Z}_m \to A_1$ und $w_2 : \mathbb{Z}_n \to A_2$ zwei Wörter.
Wir definieren

$$w_1 \cdot w_2 : \mathbb{Z}_{m+n} \to A_1 \cup A_2$$
$$i \mapsto \begin{cases} w_1(i) & \text{falls } 0 \leq i < m \\ w_2(i-m) & \text{falls } m \leq i < m+n \end{cases}$$

Umgehen mit Definitionen — am Beispiel Konkatenation

Definition

Es seien $w_1 : \mathbb{Z}_m \to A_1$ und $w_2 : \mathbb{Z}_n \to A_2$ zwei Wörter.
Wir definieren

$$\begin{aligned} w_1 \cdot w_2 : \mathbb{Z}_{m+n} &\to A_1 \cup A_2 \\ i &\mapsto \begin{cases} w_1(i) & \text{falls } 0 \le i < m \\ w_2(i-m) & \text{falls } m \le i < m+n \end{cases} \end{aligned}$$

Was tun mit so etwas?

- Nicht abschrecken lassen!
- Abbildung: *überall* definiert?
- Fallunterscheidungen: widerspruchsfrei?
- Hat das Definierte die erforderlichen Eigenschaften?
- Verstehen!

Umgehen mit Definitionen — am Beispiel Konkatenation

$$w_1 \cdot w_2 : \mathbb{Z}_{m+n} \to A_1 \cup A_2$$
$$i \mapsto \begin{cases} w_1(i) & \text{falls } 0 \leq i < m \\ w_2(i-m) & \text{falls } m \leq i < m+n \end{cases}$$

Überprüfung:

- ? $w_1(i)$ für $0 \leq i < m$ und $w_2(i-m)$ für $m \leq i < m+n$ definiert
- ? Funktionswerte aus dem Bereich $A_1 \cup A_2$
- ? Fallunterscheidung widerspruchsfrei
- ? $w_1 \cdot w_2 : \mathbb{Z}_{m+n} \to A_1 \cup A_2$ surjektiv

Umgehen mit Definitionen — am Beispiel Konkatenation

$$w_1 \cdot w_2 : \mathbb{Z}_{m+n} \to A_1 \cup A_2$$
$$i \mapsto \begin{cases} w_1(i) & \text{falls } 0 \leq i < m \\ w_2(i-m) & \text{falls } m \leq i < m+n \end{cases}$$

Überprüfung:

- ✓ $w_1(i)$ für $0 \leq i < m$ und $w_2(i-m)$ für $m \leq i < m+n$ definiert
- ? Funktionswerte aus dem Bereich $A_1 \cup A_2$
- ? Fallunterscheidung widerspruchsfrei
- ? $w_1 \cdot w_2 : \mathbb{Z}_{m+n} \to A_1 \cup A_2$ surjektiv

Umgehen mit Definitionen — am Beispiel Konkatenation

$$w_1 \cdot w_2 : \mathbb{Z}_{m+n} \to A_1 \cup A_2$$
$$i \mapsto \begin{cases} w_1(i) & \text{falls } 0 \leq i < m \\ w_2(i-m) & \text{falls } m \leq i < m+n \end{cases}$$

Überprüfung:

- ✓ $w_1(i)$ für $0 \leq i < m$ und $w_2(i-m)$ für $m \leq i < m+n$ definiert
- ✓ Funktionswerte aus dem Bereich $A_1 \cup A_2$
- ? Fallunterscheidung widerspruchsfrei
- ? $w_1 \cdot w_2 : \mathbb{Z}_{m+n} \to A_1 \cup A_2$ surjektiv

Umgehen mit Definitionen — am Beispiel Konkatenation

$$w_1 \cdot w_2 : \mathbb{Z}_{m+n} \to A_1 \cup A_2$$
$$i \mapsto \begin{cases} w_1(i) & \text{falls } 0 \leq i < m \\ w_2(i-m) & \text{falls } m \leq i < m+n \end{cases}$$

Überprüfung:

- ✓ $w_1(i)$ für $0 \leq i < m$ und $w_2(i-m)$ für $m \leq i < m+n$ definiert
- ✓ Funktionswerte aus dem Bereich $A_1 \cup A_2$
- ✓ Fallunterscheidung widerspruchsfrei
- ? $w_1 \cdot w_2 : \mathbb{Z}_{m+n} \to A_1 \cup A_2$ surjektiv

Umgehen mit Definitionen — am Beispiel Konkatenation

$$w_1 \cdot w_2 : \mathbb{Z}_{m+n} \to A_1 \cup A_2$$
$$i \mapsto \begin{cases} w_1(i) & \text{falls } 0 \le i < m \\ w_2(i-m) & \text{falls } m \le i < m+n \end{cases}$$

Überprüfung:

- ✓ $w_1(i)$ für $0 \le i < m$ und $w_2(i-m)$ für $m \le i < m+n$ definiert
- ✓ Funktionswerte aus dem Bereich $A_1 \cup A_2$
- ✓ Fallunterscheidung widerspruchsfrei
- ✓ $w_1 \cdot w_2 : \mathbb{Z}_{m+n} \to A_1 \cup A_2$ surjektiv

Wo sind wir?

Das leere Wort — neutrales Element bezüglich Konkatenation

bei den Zahlen:

$$\text{für jedes } x \in \mathbb{N}_0 : x + 0 = x \text{ und } 0 + x = x$$

Die Null ist das *neutrale Element* bezüglich der Addition.

Analog bei Wörtern:

Lemma. Für jedes Alphabet A gilt:

$$\text{für jedes } w \in A^* : w \cdot \varepsilon = w \text{ und } \varepsilon \cdot w = w \ .$$

anschaulich: klar (?)

formal: können wir das jetzt beweisen

Das leere Wort — neutrales Element bezüglich Konkatenation

Wie beweist man das für alle denkbaren Alphabete A?

- Z. B.: betrachte „beliebiges“ Alphabet A, über das keine Annahmen gemacht werden.

Das leere Wort — neutrales Element bezüglich Konkatenation

Wie beweist man das für alle denkbaren Alphabete A?

- Z. B.: betrachte „beliebiges“ Alphabet A, über das keine Annahmen gemacht werden.

Wie beweist man die Behauptung für alle $w \in A^*$?

- Z. B.: betrachte „beliebiges“ Wort w, über das keine Annahmen gemacht werden.

Das leere Wort — neutrales Element bezüglich Konkatenation

Wie beweist man das für alle denkbaren Alphabete A?

- Z. B.: betrachte „beliebiges“ Alphabet A, über das keine Annahmen gemacht werden.

Wie beweist man die Behauptung für alle $w \in A^*$?

- Z. B.: betrachte „beliebiges“ Wort w, über das keine Annahmen gemacht werden.

Also:

- Es sei A ein Alphabet und $w \in A^*$, d. h. eine surjektive Abbildung $w : \mathbb{Z}_m \to B$ mit $B \subseteq A$.
- Außerdem ist $\varepsilon : \mathbb{Z}_0 \to \{\}$.
- berechne $w' = w \cdot \varepsilon$ anhand der formalen Definition:
- w' ist eine Abbildung $w' : \mathbb{Z}_{m+0} \to B \cup \{\}$, also $w' : \mathbb{Z}_m \to B$.

Das leere Wort — neutrales Element bezüglich Konkatenation (2)

$w' : \mathbb{Z}_m \to B$

für $i \in \mathbb{Z}_m$ gilt

$$w'(i) = \begin{cases} w(i) & \text{falls } 0 \leq i < m \\ \varepsilon(i-m) & \text{falls } m \leq i < m+0 \end{cases}$$
$$= w(i)$$

- w und w' haben gleichen Definitionsbereich
- w und w' haben gleichen Zielbereich
- w und w' haben überall die gleichen Funktionswerte.
- also $w' = w$, also $w \cdot \varepsilon = w$.

Wo sind wir?

Eigenschaften der Konkatenation — nicht kommutativ aber assoziativ

Konkatenation ist *nicht kommutativ*

$\mathtt{SCHRANKSCHLÜSSEL} \neq \mathtt{SCHLÜSSELSCHRANK}$

- Reihenfolge ist wichtig

Eigenschaften der Konkatenation — nicht kommutativ aber assoziativ

Konkatenation ist *nicht kommutativ*

$\mathtt{SCHRANKSCHLÜSSEL} \neq \mathtt{SCHLÜSSELSCHRANK}$

- Reihenfolge ist wichtig

Bei Zahlen gilt: $(x \cdot y) \cdot z = x \cdot (y \cdot z)$

Konkatenation ist *assoziativ*

Lemma.
Für jedes Alphabet A und alle $w_1, w_2, w_3 \in A^*$ gilt:

$$(w_1 \cdot w_2) \cdot w_3 = w_1 \cdot (w_2 \cdot w_3) \ .$$

Beweis: nachrechnen

Wo sind wir?

RFC

RFC ist die Abkürzung für *Request For Comment*.

alle RFCs zum Beispiel unter
`http://tools.ietf.org/html/`

Spezifikationen der *Internet Engineering Task Force (IETF)*

z. B. Struktur von E-Mails

aktuelle Fassung in RFC 5322
`http://tools.ietf.org/html/rfc5322`

im folgenden einige Zitate aus Abschnitt 2.1 des RFC 5322 und Kommentare dazu

E-Mails, RFC 5322 (1)

„This document specifies that messages are made up of characters in the US-ASCII range of 1 through 127.“

Alphabet für Emails
ASCII-Zeichensatz mit Ausnahme des Zeichens mit Nr. 0

E-Mails, RFC 5322 (2)

„Messages are divided into lines of characters.
A line is a series of characters that is delimited with the two characters carriage-return and line-feed; that is, the carriage return (CR) character (ASCII value 13) followed immediately by the line feed (LF) character (ASCII value 10).
(The carriage-return/line-feed pair is usually written in this document as "CRLF".)"

Eine Zeile (*line*) ist

- eine Folge von Zeichen, also ein Wort,
- das mit den „nicht druckbaren" Symbolen $\boxed{\texttt{CR}}$ $\boxed{\texttt{LF}}$ endet.

- an anderer Stelle:
  - als Zeile nicht beliebige Wörter zulässig,
  - sondern nur solche, deren Länge kleiner oder gleich 998 ist.

E-Mails, RFC 5322 (3)

A message consists of

- *[...] header fields [...] called the header section [...] followed,*
- *optionally, by a body.“*

eine E-Mail (*message*) ist die Konkatenation von

- Kopf (*header section*) der E-Mail und
- Rumpf (*body*) der E-Mail.

Rumpf optional,

- darf also sozusagen fehlen darf,
- d.h. der Rumpf darf auch das leere Wort sein.

Das ist noch nicht ganz vollständig. Gleich anschließend wird der RFC genauer:

E-Mails, RFC 5322 (4)

- *„The header section is a sequence of lines of characters with special syntax as defined in this specification.*
- *The body is simply a sequence of characters that follows the header section and*
- *is separated from the header section by an empty line (i.e., a line with nothing preceding the CRLF). [...]“*

also:

- Kopf einer E-Mail: Konkatenation (mehrerer) Zeilen.
- Rumpf einer E-Mail: Konkatenation von Zeilen.
  - (an anderer Stellen spezifiziert)
  - Es können aber auch 0 Zeilen oder 1 Zeile sein.
- Leerzeile (*empty line*) ist das Wort $\boxed{\texttt{CR}}\,\boxed{\texttt{LF}}$.
- Email ist Konkatenation von
  - Kopf
  - einer Leerzeile
  - Rumpf

Wo sind wir?

Definition des Begriffs „Wort“

Konkatenation von Wörtern

Eigenschaften der Konkatenation

Beispiel: Aufbau von E-Mails

Iterierte Konkatenation

Formale Sprachen

Binäre Operationen

Iterierte Konkatenation: Potenzen von Wörtern

bei Zahlen: Potenzschreibweise x^3 für $x \cdot x \cdot x$ usw.

Ziel: analog für Wörter so etwas wie

$$w^n = \underbrace{w \cdot w \cdot \cdots \cdot w}_{n \text{ mal}}$$

Wie kann man die Pünktchen vermeiden?

- Was ist mit $n = 1$? $n = 2$?
- $n = 0$?

Möglichkeit: eine *induktive Definition*

für *Potenzen von Wörtern* geht das so:

$$\begin{aligned} w^0 &= \varepsilon \\ \text{für jedes } n \in \mathbb{N}_0 : \quad w^{n+1} &= w^n \cdot w \end{aligned}$$

Induktive Definitionen kann man zu Rechnen benutzen

Definition

$$w^0 = \varepsilon$$

$$\text{für alle } n \in \mathbb{N}_0 : \; w^{n+1} = w^n \cdot w$$

Was ist w^1 ?

$$w^1 = w^{0+1} = w^0 \cdot w = \varepsilon \cdot w = w$$

Was ist w^2 ?

$$w^2 = w^{1+1} = w^1 \cdot w = w \cdot w$$

Was ist w^3 ?

$$w^3 = w^{2+1} = w^2 \cdot w = (w \cdot w) \cdot w$$

Ein einfaches Lemma zu Längen von Wortpotenzen

Lemma.
Für jedes Alphabet A, jedes $w \in A^*$ und jedes $n \in \mathbb{N}_0$ gilt:

$$|w^n| = n|w| \ .$$

Wie kann man das beweisen?

Wenn etwas induktiv Definiertes eine Rolle spielt, sollte man *vollständige Induktion* als Beweismethode erwägen.

- Details nach dem Kapitel zu Aussagenlogik

Das einfache Lemma — einfache Fälle

- $n = 0$: Das ist einfach: $|w^0| = |\varepsilon| = 0 = 0 \cdot |w|$.
- $n = 1$: Man kann ähnlich rechnen wie bei $w^1 = w$:

$$\begin{aligned} |w^1| = |w^{0+1}| &= |w^0 \cdot w| \\ &= |w^0| + |w| \\ &= 0|w| + |w| && \text{siehe Fall } n = 0 \\ &= 1|w| \end{aligned}$$

  weil für $n = 0$ richtig auch für $n = 1$ beweisbar
- $n = 2$: Wir gehen analog zu eben vor:

$$\begin{aligned} |w^2| = |w^{1+1}| &= |w^1 \cdot w| \\ &= |w^1| + |w| \\ &= 1|w| + |w| && \text{siehe Fall } n = 1 \\ &= 2|w| \end{aligned}$$

  weil für $n = 1$ richtig auch für $n = 2$ beweisbar

Vollständige Induktion — kurzer Ausblick

allgemeines Muster:

- weil w^{n+1} mit Hilfe von w^n definiert,
- folgt aus Richtigkeit der Behauptung für $|w^n|$ die für $|w^{n+1}|$.

M: Menge aller natürlichen Zahlen n,
für die die Behauptung $|w^n| = n|w|$ gilt:

1. $0 \in M$
2. für jedes $n \in \mathbb{N}_0$: wenn $n \in M$, dann $n + 1 \in M$

Mathematik: Wenn eine Menge M

- nur natürliche Zahlen enthält,
- Eigenschaft 1 hat und
- Eigenschaft 2 hat,

dann ist $M = \mathbb{N}_0$.

Wo sind wir?

Sprachen

natürliche Sprache:

- Aussprache, Stil, Wortwahl, Satzbau, …
- Welche Formulierungen sind syntaktisch korrekt?
- Ist und syntaktische welcher korrekt nicht?

Informatik: künstliche Sprachen

- Programmiersprachen
- Aufbau von Emails, WWW-Seiten, …
- Eingabedateien für …

Syntax

- Wie spezifiziert man, was korrekt ist?
- Wie überprüft man, ob etwas korrekt ist?

Semantik

- Wie definiert man Bedeutung syntaktisch korrekter Gebilde?

Formale Sprache über einem Alphabet A — eine Teilmenge $L \subseteq A^*$

primitiver Standpunkt hier:

- Bedeutung spielt keine Rolle

im Zusammenhang mit syntaktischer Korrektheit:

- formale Sprache L der syntaktisch korrekten Gebilde
- syntaktisch falsche Gebilde gehören *nicht* zu L

Beispiele:

- $A = \{$`0`, `1`, `2`, `3`, `4`, `5`, `6`, `7`, `8`, `9`, `-`$\}$.
  formale Sprache der Dezimaldarstellungen ganzer Zahlen
  - enthält z.B. `1`, `-22` und `192837465`,
  - aber nicht `2-3--41`.
- formale Sprache der syntaktisch korrekten Java-Programme
  - enthält alle Java-Programme
  - enthält zum Beispiel *nicht*: `[2] class int)(`

Wo sind wir?

Binäre Operationen

binäre Operation auf Menge M ist eine Abbildung

$$f : M \times M \to M$$

üblich: Infixschreibweise

- Statt $+(3, 8) = 11$ schreibt man $3 + 8 = 11$.

binäre Operation $\diamond : M \times M \to M$ genau dann *kommutativ*, wenn gilt:

$$\text{für alle } x \in M \;\; \text{für alle } y \in M : \;\; x \diamond y = y \diamond x \; .$$

binäre Operation $\diamond : M \times M \to M$ genau dann *assoziativ*, wenn gilt:

$$\text{für alle } x \in M, y \in M, z \in M : \;\; (x \diamond y) \diamond z = x \diamond (y \diamond z) \; .$$

Was ist wichtig

Das sollten Sie mitnehmen:

- ein *Wort* ist eine Folge von Symbolen
- *induktive Definitionen*
  - erlauben, Pünktchen zu vermeiden …

Das sollten Sie üben:

- „Rechnen“ mit Wörtern