Wörter sind Zeichenketten

- Programmiersprachen:

  Namen, Ausdrücke, Datentypen, Programme,...
- Systemmodellierung:

  mögliche Abläufe, Ereignisfolgen, verbotene Folgen,...
- Natürliche Sprachen:

  Wörter, Sätze, Texte, ...
- Textsysteme:

  Schreiben und Lesen, Verändern und analysieren: Cut & Paste, Zeichen zählen, Vergleichen, Zusammensetzen, ...
- Eine der wichtigsten Datenstrukturen

Erzeugung von Wörtern

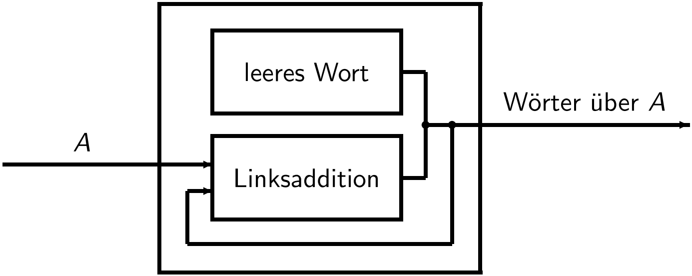
- A : Alphabet (Menge von Zeichen)
- A^*: Menge aller Wörter über A.

Rekursive Definition von A^*

- $\lambda \in A^*$ (leeres Wort, enthält keine Zeichen)
- mit $v \in A^*$ und $x \in A$ ist auch $xv \in A^*$ (Linksaddition)

Schreibweise: $x\lambda$ wird mit x abgekürzt.

Erzeugung von Wörtern



Länge

1. $length(\lambda) = 0$
2. $length(xv) = length(v) + 1$ für $x \in A, v \in A^*$

Beispiel

$length(aba) = 1 + length(ba) = 1 + 1 + length(a) =$
$2 + 1 + length(\lambda) = 3 + 0 = 3$

Zeichen zählen

1. $count(x, \lambda) = 0$
2. $count(x, yv) = \begin{cases} count(x, v) + 1, & \text{falls } x = y \\ count(x, v) & \text{sonst} \end{cases}$

für alle $x, y \in A$, $v \in A^*$.

Induktionsprinzip

- Induktionsanfang (IA):
  Zeige THEOREM für $v = \lambda$.
- Induktionsvoraussetzung (IV):
  Nimm THEOREM für v an.
- Induktionsschluss (IS):
  Zeige THEOREM für xv mit $x \in A$.

Eindeutige Zerlegbarkeit von Wörtern

- Für alle $w \in A^*$:
  - entweder $w = \lambda$, oder
  - $w = xv$ für genau ein $x \in A$ und genau ein $v \in A^*$
- $head(xv) = x,\ tail(xv) = v$
- Falls A selbst Wörter enthält, muss die eindeutige Zerlegbarkeit garantiert werden.

Beispiel Für $A = \{B, E, I, BEI, SPIEL\}$ gilt
$head(B \circ E \circ I \circ SPIEL) = B \neq$
$head(BEI \circ SPIEL) = BEI$

Bemerkungen über Wörter

- $A \subseteq A^*$
- Rechtsaddition kann anolog zur Linksaddition definiert werden.
- Iterative Darstellung von Wörtern: $w = x_1 \cdots x_n$ mit $n \in \mathbb{N}$ und $x_i \in A$ $(i = 1, \ldots, n)$.
- Falls $A = \{|\}$ repräsentiert A^* die natürlichen Zahlen:

$$\lambda = |^0 \text{ repräsentiert } 0$$
$$||^n \text{ repräsentiert } n + 1$$

Konkatenation von Wörtern

1. für $v = \lambda$ gilt $\lambda \cdot w = w$,
2. für $v = xu$ mit $x \in A$ gilt $(xu) \cdot w = x(u \cdot w)$.

Beispiel

$$aba \cdot cd = a(ba \cdot cd) = a(b(a \cdot cd)) = a(b(a(\lambda \cdot cd))) = a(b(a(cd))) = a(b(acd)) = a(bacd) = abacd$$

> Satz (Assoziativität)
> Für alle Wörter u, v, w gilt $(uv)w = u(vw)$.

Länge

1. $length(\lambda) = 0$
2. $length(xv) = length(v) + 1$ für $x \in A, v \in A^*$

$length(aba) = 1 + length(ba) = 1 + 1 + length(a) =$
$2 + 1 + length(\lambda) = 3 + 0 = 3$

Satz

Für alle $u, v \in A^*$ gilt:

$$length(uv) = length(u) + length(v).$$

Zeichen zählen

1. $count(x, \lambda) = 0$
2. $count(x, yv) = \begin{cases} count(x, v) + 1, & \text{falls } x \equiv y \\ count(x, v) & \text{sonst} \end{cases}$

Satz

Für alle $x \in A$, $u, v \in A^*$ gilt:

$$count(x, uv) = count(x, u) + count(x, v).$$

Gleichheitstest

1. $\lambda \equiv \lambda$,
2. $xu \equiv yv$, falls $x \equiv y$ und $u \equiv v$ für alle $x, y \in A$, $u, v \in A^*$

(Voraussetzung: Gleichheitstest $\equiv$ auf A gegeben.)

Satz

Für alle $u, v, w \in A^*$ gilt:

$$u \equiv v \text{ impliziert } uw \equiv vw.$$