Terme und Algebren

Übersicht:

1. **Teil 1: Sorten und Terme**
2. Teil 2: Algebren

Sorten und Terme

Motivation

Formalisierung in der Prädikatenlogik 1. Stufe:

Beispiel: $\forall x \, \exists y P(x, f(x, y))$

Eine Interpretation besteht (unter anderem) aus *nur einem* Grundbereich ω und der Angabe einer konkreten Funktion $f_\omega : \omega \times \omega \to \omega$.

Wir möchten eine speziellere Interpretation verwenden:

$$f^* : \{0, 1, 2, 3\} \ x \ \{rot, gelb\} \to \{rot, gelb\}$$

Formalismus:

- Verwendung von Sorten: z.B. A und B als Platzhalter für Grundbereiche
- Funktionen/Operationen mit Sorten: $f : A \ x \ B \to B$

Typisch: mehrere Sorten (vgl. Typen in Programmiersprachen)

Sorten und Terme

Schritt 1: Sorten und Strukturen

Definition 1

Sei $\mathcal{S}$ eine endliche Menge von **Sorten** (nur Namen/Symbole für Grundbereiche, keine konkreten Mengen) und seien $S_1, \ldots S_k, S \in \mathcal{S}$. Dann bezeichnet $f : S_1 \; x \; \ldots \; x \; S_k \rightarrow S$ eine **Operation** (der Sorte S) zu $\mathcal{S}$. f ist das **Operatorsymbol** (vgl. Funktionssymbol in PL1); die **Stelligkeit** von f ist k. Eine 0-stellige Operation $a : \quad \rightarrow S$ ist eine **Konstante** der Sorte S. Eine **Struktur** $\mathcal{F}$ ist eine endliche Menge von Operationen zu $\mathcal{S}$.

Beispiel für eine Struktur:

$$
\begin{aligned}
\mathcal{F} = \{ \quad & + : A \; x \; B \rightarrow B, \\
& * : A \; x \; B \rightarrow B, \\
& - : A \rightarrow A, \\
& a : \quad \rightarrow B \qquad \qquad \}
\end{aligned}
$$

$+, *, -, a$ sind Operatorsymbole, a ist ein Konstantensymbol.

Sorten und Terme

Schritt 1: Terme

Definition 2 ((wohlgeformte) Terme)

Sei $\mathcal{S} = \{S_1, \ldots, S_n\}$ eine endliche Menge von *Sorten* und sei $\mathcal{F}$ eine Struktur zu $\mathcal{S}$.
Zu jedem S_j sei eine Mengen von Variablennamen V_j gegeben, $1 \leq j \leq n$.
Die Terme der Sorte S_k werden für $1 \leq k \leq n$ gleichzeitig induktiv definiert durch:

1. Jede Variable $v \in V_k$ ist ein Term der Sorte S_k.
2. Jeder 0-stellige Operator (Konstante) der Sorte S_k ist ein Term der Sorte S_k.
3. Für $1 \leq j \leq r$ sei t_j ein Term der Sorte S_{i_j} und $f : S_{i_1} \ x \ldots \ x \ S_{i_r} \rightarrow S_k$ eine Operation in $\mathcal{F}$. Dann ist $f(t_1, \ldots, t_r)$ ein Term der Sorte S_k.
4. Nur so gebildete Terme sind Terme der Sorte S_k.

Signatur

Definition 3 (Signatur)

Sei $\mathcal{S} = \{S_1, \ldots S_n\}$ eine endliche Menge von Sorten und sei $\mathcal{F}$ eine Struktur zu $\mathcal{S}$. Dann nennen wir $\Sigma = (\{S_1, \ldots, S_n\}, \mathcal{F})$ eine *Signatur*.

Seien V_j Variablensymbole der Sorte S_j für $1 \leq j \leq n$, dann bezeichnet $T_\Sigma(V_1, \ldots, V_n)$ die Menge der wohlgeformten Terme zu Σ mit Variablen in $V_1, \ldots, V_n$.

Die auf Basis einer Struktur gebildeten Terme sind nur dann korrekt, wenn sie alle Sortenrestriktionen einhalten.

Beispiel:
Sei $\mathcal{F} = \{f : S_1 \to S_2, g : S_1 \to S_2\}$. Dann ist $f(g(x))$ kein für die Signatur $\Sigma = (\{S_1, S_2\}, \mathcal{F})$ korrekt gebildeter Term.

Termnotation

Definition 4 (Termnotation)

Ein n-stelliger Term mit dem Operatorsymbol f und den Termen $t_1, \ldots, t_n$ als Operanden wird notiert in

1. **Funktionsform**: $f(t_1, \ldots, t_n)$, wobei die t_i auch in Funktionsform sind.
2. **Präfixform**: $(f \; t_1 \cdots t_n)$, wobei die t_i auch in Präfixform sind.
3. **Postfixform**: $(t_1 \cdots t_n \; f)$, wobei die t_i auch in Postfixform sind.
4. **Infixform**: (nur für max. zweistellige Operatoren)
   $(t_1 \; f \; t_2)$ bzw. $(f \; t_1)$, wobei die t_i auch in Infixform sind.

Beispiele:

1. $+\,(*\,(x, 3), -\,(2, 3))$ Funktionsform
2. $(\,+\,(\,*\,x\;3)\,(\,-\,2\;3))$ Präfixform
3. $(\,(\,x\;3\;*)\,(\,2\;3\;-)\,+)$ Postfixform
4. $(\,(\,x\;*\;3)\;+\;(\,2\;-\;3))$ Infixform

Termnotation

Beispiele:

$x + y$ Infixform
$+ \ x \ y$ Präfixform
$x \ y \ +$ Postfixform
$+(x, y)$ Funktionsform

$x \wedge y$ Infixform
$\wedge \ x \ y$ Präfixform
$x \ y \ \wedge$ Postfixform
$\wedge(x, y)$ Funktionsform

$f \ x \ y \ z$ Präfixform
$x \ y \ z \ f$ Postfixform
$f(x, y, z)$ Funktionsform

Termnotation

Beispiele:

$* \; x \; (+ \; y \; z)$ Präfixform

$x \; (y \; z \; +) \; *$ Postfixform

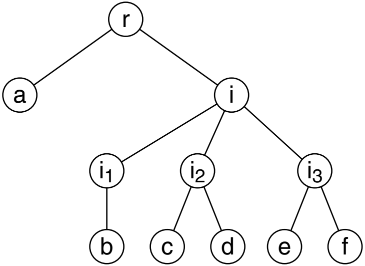
$*(x, +(y, z))$ Funktionsform

$x \; * \; (\; y \; + \; z \;)$ Infixform

In allen Notationen ist die Einsparung von Klammern nur bei festgelegten Einsparungsregeln möglich: z.B. Operatorpräzedenz (Punktrechnung vor Strichrechnung) oder Linksklammerung (bei assoziativen Operatoren).

Bäume als spezielle Graphen

Beispiel:



r ist die Wurzel des Baumes.

i_1, i_2 und i_3 sind die Nachfolger des Knotens i.

Jeder Knoten hat nur endlich viele oder keinen Nachfolger.

Knoten ohne Nachfolger werden Blätter genannt.

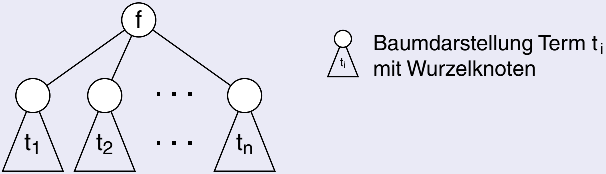
Alle Knoten außer den Blättern sind innere Knoten.

Terme und Baumdarstellung

Definition 5 (Baumdarstellung)

Die Baumdarstellung von Termen wird induktiv definiert:

1. Ist der Term eine Variable x, dann besteht der Baum nur aus einem Knoten mit Namen x: x
2. Ist der Term eine Konstante a, dann besteht der Baum nur aus einem Knoten mit Namen a: a
3. Ein n-stelliger Term $f(t_1, \ldots, t_n)$ mit dem Operatorsymbol f und den Untertermen $t_1, \ldots, t_n$ wird als Baum dargestellt durch

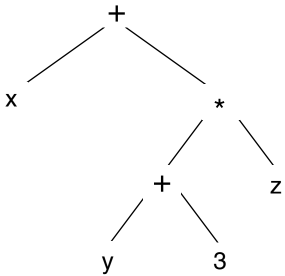


Achtung: Die Nachfolger eines Knotens werden von links nach rechts gelesen.

Terme und Baumdarstellung

Beispiel:

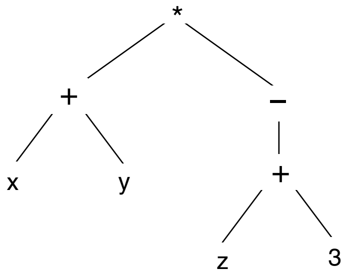
Der nachfolgende Baum repräsentiert den Term
Funktionsform $+(x, *(+(y, 3), z))$



Terme und Baumdarstellung

Beispiel:

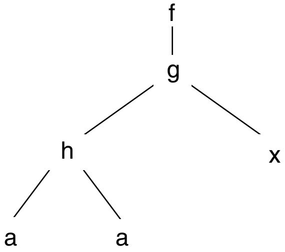
Der nachfolgende Baum repräsentiert den Term
Term: $(x + y) * -(z + 3)$
Funktionsform $*((+(x, y)), -(+(z, 3)))$



Terme und Baumdarstellung

Beispiel:

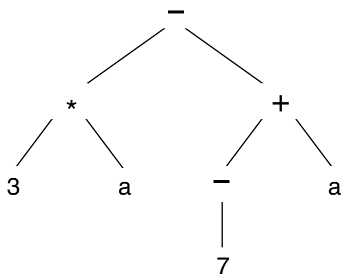
Der nachfolgende Baum repräsentiert den (in Funktionsform angegebenen) Term $f(g(h(a,a),x))$.



Termnotation und Baumdarstellung

Beispiel:

Der nachfolgende Baum repräsentiert den Term $(3 * a) - (-7 + a)$ (in Infixdarstellung).



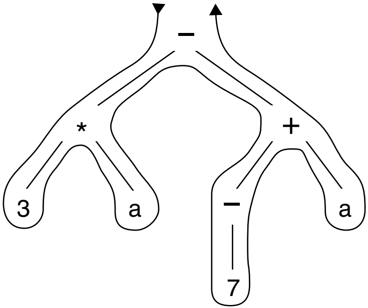
Der Baum repräsentiert die eindeutige Struktur des Terms.

Termnotation und Baumdarstellung

Beispiel: (Fortsetzung)

Wie können die möglichen Termformen an dem Baum abgelesen werden?

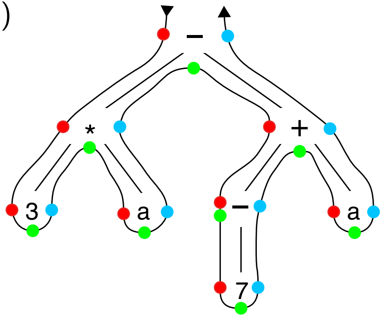
Führe einen Durchlauf durch den Baum durch (Tiefensuche-Durchlauf)



Gebe unter festgelegten Bedingungen die Knotensymbole aus!

Termnotation und Baumdarstellung

Beispiel: (Fortsetzung)



Aus einem Durchlauf des Baumes in Pfeilrichtung erzeugt man eine

Präfixform: Ausgabe Symbol beim ersten Besuch eines Knotens,
Postfixform: Ausgabe Symbol beim letzten Besuch eines Knotens,
Infixform: Ausgabe Symbol beim zweiten Besuch eines Knotens.

Für Knoten mit nur einem Nachfolger fallen erster und zweiter Besuch zusammen, für Blätter zählt das Erreichen als erster, zweiter und letzter Besuch. Bei inneren Knoten werden vor dem ersten und nach dem letzten Besuch des Knotens die entsprechenden Klammern ausgegeben.

Substitution und Unifikation

Als **Substitution** bezeichnet man die Ersetzung einer Variable in einem Term durch einen Term passender Sorte. Entsprechende Umformungen sind aus dem Arbeiten mit Formeln bekannt.

Die Frage, ob zwei Terme durch Substitution gleich gemacht werden können, führt auf den Begriff der **Unifikation**.

Substitution

Definition 6 (Einfache Substitution)

Eine einfache Substitution $\sigma = [v/t]$ ist ein Paar aus einer Variablen v und einem Term t zur Signatur Σ. Dabei gehören v und t derselben Sorte an.

Definition 7 (Anwendung einer Substitution)

Für die Anwendung einer Substitution $[v/t]$ auf einen Term u schreibt man $u[v/t]$. Das Ergebnis einer solchen Anwendung ist ein Term, der durch eine der folgenden Fälle bestimmt wird:

1. $u[v/t] = t$, falls u die zu ersetzende Variable v ist.
2. $u[v/t] = u$, falls u eine Konstante oder eine andere Variable als v ist.
3. $u[v/t] = f(u_1[v/t], u_2[v/t], \dots, u_n[v/t])$, falls $u = f(u_1, \dots, u_n)$.

Substitution

Beispiel: Betrachte den Term $x + x - 1$, den man auch in Funktionsform als

$$-(+(x, x), 1)$$

schreiben kann. Anwendung der Substitution $\sigma = [x/2 * b]$ bzw. $\sigma = [x/ * (2, b)]$ auf diesen Term ergibt

$$\begin{aligned} -(+(x, x), 1)\,[x/ * (2, b)] &= -(+(x, x)\,[x/ * (2, b)], 1\,[x/ * (2, b)]) \\ &= -(+(x, x)\,[x/ * (2, b)], 1) \\ &= -(+(x\,[x/ * (2, b)], x\,[x/ * (2, b)]), 1) \\ &= -(+(*(2, b), *(2, b)), 1) \end{aligned}$$

In Infixnotation entspricht das Ergebnis also dem Term

$$(2 * b) + (2 * b) - 1\,.$$

Substitution

Wichtig zu bemerken ist, dass in der Anwendung einer Substitution $[v/t]$ auf einen Term u

- gleichzeitig **alle** Vorkommen von v in u durch t ersetzt werden, und
- eine solche Ersetzung nur **einmal** erfolgt (also nicht rekursiv fortgesetzt wird, falls v erneut in t vorkommt).

Beispiel:

$$(x + y)\,[y/y * y] = (x + y * y)$$

Substitution

Definition 8 (Mehrfache Substitution)

In einer mehrfachen Substitution $[v_1/t_1, v_2/t_2, \ldots, v_n/t_n]$ mit $n \geq 1$ müssen alle Variablen v_i paarweise verschieden sein. Die Paare v_i/t_i müssen jeweils zur selben Sorte S_i gehören. Angewandt wird σ auf einen Term t dann wie folgt:

1. $u\,\sigma = t_i$, falls $u = v_i$ für ein $i \in \{1, \ldots, n\}$.
2. $u\,\sigma = u$, falls u eine Konstante oder eine Variable ist, die nicht in $\{v_1, \ldots, v_n\}$ vorkommt.
3. $u\,\sigma = f(u_1\,\sigma, u_2\,\sigma, \ldots, u_k\,\sigma)$, falls $u = f(u_1, \ldots, u_k)$.

Wiederum werden alle Unterterme unabhängig voneinander und nur einmal substituiert.

Beispiel: $(x + y)[x/2 * b, y/(3 + x)] = (2 * b + (3 + x))$

Hintereinanderausführung

Da die Anwendung einer Substitution wieder einen Term liefert, kann man mehrere Substitutionen hintereinander ausführen. Die Hintereinanderausführung liefert im Allgemeinen ein anderes Ergebnis als die mehrfache Substitution mit den gleichen Variablen/Term-Paaren.

Beispiel:

$$\begin{aligned}(x+y)[x/y*x][y/3][x/a] &= (y*x+y)[y/3][x/a]\\ &= (3*x+3)[x/a]\\ &= (3*a+3)\end{aligned}$$

Hintereinanderausführung

Für zwei hintereinander auszuführende Substitution σ_1 und σ_2 lässt sich eine Substitution $\sigma = \sigma_1\sigma_2$ bestimmen mit gleicher Wirkung bestimmen, sodass also

$$(u\,\sigma_1)\,\sigma_2 = u\,\sigma$$

für alle Terme u. Allgemein gilt:

$$(u\,\sigma_1)\,\sigma_2 = u\,(\sigma_1\,\sigma_2)$$

Die Hintereinanderausführung von $[x_1/t_1, \ldots, x_n/t_n]$ und $[y/r]$ hat die gleiche Wirkung wie

- $[x_1/t_1[y/r], \ldots, x_n/t_n[y/r]]$ falls $y \in \{x_1, \ldots, x_n\}$,
- $[x_1/t_1[y/r], \ldots, x_n/t_n[y/r], y/r]$ falls $y \notin \{x_1, \ldots, x_n\}$.

Hintereinanderausführung

Beispiel:

$$
\begin{aligned}
[x/y*x][y/3][x/a] &= [x/y*x[y/3],[y/3]][x/a] \\
&= [x/3*x, y/3][x/a] \\
&= [x/3*x[x/a], y/3[x/a]] \\
&= [x/3*a, y/3]
\end{aligned}
$$

Wir bezeichnen die **leere Substitution** mit $[\,]$. Für alle Terme t gilt $t[\,] = t$, und für alle Variablen v gilt $[v/v] = [\,]$.

Die umfasst-Relation

Definition 9

Ein Term s umfasst einen Term t, wenn es eine Substitution σ gibt mit $s\,\sigma = t$.

Die umfasst-Relation definiert eine Quasiordnung auf der Menge der Terme, denn sie ist

- reflexiv,
- transitiv,
- aber nicht antisymmetrisch.

Unifikation

Definition 10 (Unifikation)

Zwei Terme s und t sind **unifizierbar**, wenn es eine Substitution σ gibt mit $s\,\sigma = t\,\sigma$. In diesem Fall ist σ eine **Unifikator** von s und t.

Beispiel:
$s = x + 3$ und $t = 2 + y$ sind unifizierbar mit $\sigma = [x/2, y/3]$.
$s = x + x$ und $t = 2 + 3$ sind nicht unifizierbar.

Unifikatoren sind nicht eindeutig. Beispielsweise besitzen $s = (x + y)$ und $t = (2 + z)$ u.a. die folgenden Unifikatoren:

$$\sigma_1 = [x/2, y/z] \qquad \sigma_2 = [x/2, z/y]$$
$$\sigma_3 = [x/2, y/1, z/1] \qquad \sigma_4 = [x/2, y/2, z/2]$$

Unifikation

Definition 11 (Allgemeinster Unifikator)

Ein Unifikator σ_a heißt allgemeinster Unifikator der Terme s und t, wenn es zu jedem Unifikator σ_i von s und t eine Substitution τ_i gibt mit $\sigma_a \tau_i = \sigma_i$.

Algorithmus zum Finden eines allgemeinsten Unifikators von s und t:

1. Setze $\sigma = [\,]$.
2. Solange $s\,\sigma \neq t\,\sigma$ wiederhole Schritte 3 und 4:
3. Setze $(u, v) = \text{Abweichungspaar}(s\,\sigma, t\,\sigma)$.
4. Falls
    - u ist eine Variable x, die nicht in v vorkommt, dann setze $\sigma = \sigma[x/v]$;
    - v ist eine Variable x, die nicht in u vorkommt, dann setze $\sigma = \sigma[x/u]$;
    - sonst sind die Terme nicht unifizierbar (Abbruch des Algorithmus).
5. Bei Erfolg gilt $s\,\sigma = t\,\sigma$ und σ ist ein allgemeinster Unifikator.

Unifikation

Bestimmung von $(u, v) = \text{Abweichungspaar}(s, t)$ für $s \neq t$:

$(u, v) = (s, t)$ falls

- $s = f(\dots)$ und $t = g(\dots)$ und $f \neq g$
- $s = f(\dots)$ und $t = x$
- $s = x$ und $t = f(\dots)$
- $s = x$ und $t = y$ und $x \neq y$

Ansonsten ist $(u, v) = \text{Abweichungspaar}(t_i, t_i')$, wobei i der kleinste Index ist mit

$$s = f(t_1, \dots, t_{i-1}, t_i, \dots) \text{ und } t = f(t_1, \dots, t_{i-1}, t_i', \dots)$$

und $t_i \neq t_i'$.

Unifikation

Beispiel:

$$s[\,] = +(x, y)$$
$$t[\,] = +(z, *(2, x))$$

Das Abweichungspaar (x, z) liefert $[x/z]$:

$$s[\,][x/z] = +(z, y)$$
$$t[\,][x/z] = +(z, *(2, z))$$

Das Abweichungspaar $(y, *(2, z))$ liefert $[y/*(2, z)]$:

$$s[\,][x/1][y/*(2, z)] = +(z, *(2, z))$$
$$t[\,][x/1][y/*(2, z)] = +(z, *(2, z))$$

Ein allgemeinster Unifikator ist $[x/z][y/*(2, z)] = [x/z, y/*(2, z)]$.

Terme und Algebren

Übersicht:

1. Teil 1: Sorten und Terme
2. **Teil 2: Algebren**

Motivation

Eine **Algebra** besteht aus

- Einer Menge grundlegender Elemente (Trägermenge),
- Operationen über dieser Menge,
- Gesetze, die die Operationen zueinander in Bezug setzen.

Hiermit lassen sich **formale Kalküle** formulieren, wie z.B.

- Algebra der Mengenoperationen,
- Boolesche Algebra,
- Arithmetik ganzer Zahlen,
- veränderliche Datenstrukturen wie Keller und Listen,
- Abläufe in dynamischen Systemen.

Motivation

Idee:
Formulierung von Eigenschaften für Term-Umformung und Vereinfachung.

Gegeben sei eine Signatur $\Sigma = (\{A\}, \mathcal{F})$ und eine Variablenmenge $V = \{x_1, \ldots, x_n\}$ der Sorte A mit $\mathcal{F} = \{f : A \text{ x } A \to A,\ h : A \text{ x } A \to A\}$.

Terme werden umgeformt anhand von Regeln.

Ersetzungsregel: $h(f(x, y), z) \to f(h(x, z), h(y, z))$

Term $h(f(x, y), z)$ kann ersetzt werden durch den Term $f(h(x, z), h(y, z))$

Beispiel: Aus $h(h(f(x_1, x_2), x_3), x_4))$ wird $h(f(h(x_1, x_3), h(x_2, x_3)), x_4)$.

Die Ersetzungsregeln werden auch als Axiome bezeichnet.

Konkrete Algebren

Gegeben sei eine Signatur $\Sigma = (\{A\}, \mathcal{F})$ und eine Variablenmenge $V = \{x_1, \ldots, x_n\}$ der Sorte A mit $\mathcal{F} = \{f : A \; x \; A \to A, \; h : A \; x \; A \to A\}$

Ersetzungsregel (Axiom): $h(f(x, y), z) \to f(h(x, z), h(y, z))$

Konkrete Algebra: (vgl. Interpretation in der Prädikatenlogik)
- Zuordnung von konkreten Sorten für Sortennamen
- Zuordnung von konkreten Operationen für Operatornamen

Beispiel:

1. Konkreter Grundbereich: A wird zugeordnet $\mathbb{N}$
2. Konkrete Funktionen: f wird zugeordnet $+ : \mathbb{N} \; x \; \mathbb{N} \to \mathbb{N}$
   h wird zugeordnet $* : \mathbb{N} \; x \; \mathbb{N} \to \mathbb{N}$

Nach der Ersetzung der Operatorsymbole durch konkrete Funktionen gilt für das Axiom die Gleichheit. Die konkrete Algebra ist ein Modell.

$h(f(x, y), z) \to f(h(x, z), h(y, z)$

$*(+(x, y), z) = +(*(x, z), *(y, z))$ (Assoziativgesetz)

Axiome

Definition 12

Sei $\Sigma = (\mathcal{S}, \mathcal{F})$ eine Signatur. Ein Axiom hat die Form $t_1 \to t_2$, wobei die Terme t_1 und t_2 wohlgeformte Terme der gleichen Sorte sein müssen.

Axiome

Definition 12

Sei $\Sigma = (\mathcal{S}, \mathcal{F})$ eine Signatur. Ein Axiom hat die Form $t_1 \rightarrow t_2$, wobei die Terme t_1 und t_2 wohlgeformte Terme der gleichen Sorte sein müssen.

Signatur $\Sigma = (\{BOOL\}, \mathcal{F})$.

$$
\begin{array}{lrl}
\mathcal{F} = \{ & true : & \rightarrow BOOL \\
& false : & \rightarrow BOOL \\
& f_\wedge : & BOOL \text{ x } BOOL \rightarrow BOOL \\
& f_\vee : & BOOL \text{ x } BOOL \rightarrow BOOL \\
& f_\neg : & BOOL \rightarrow BOOL \qquad \}
\end{array}
$$

Axiome: Seien x und y Variable der Sorte $BOOL$.

$Q_1 : f_\neg(true) \rightarrow false$

$Q_2 : f_\neg(false) \rightarrow true$

$Q_3 : f_\wedge(true, x) \rightarrow x$

$Q_4 : f_\wedge(false, x) \rightarrow false$

$Q_5 : f_\vee(x, y) \rightarrow f_\neg(f_\wedge(f_\neg(x), f_\neg(y)))$

Abstrakte Algebra

Definition 13

Sei $\Sigma = (\mathcal{S}, \mathcal{F})$ eine Signatur mit zugehörigen Variablenmengen und Q eine endliche Menge von Axiomen, dann wird $((\mathcal{S}, \mathcal{F}), Q)$ als **abstrakte Algebra** bezeichnet. Alternativ wird auch für die Menge τ der wohlgeformten Terme der Signatur $(\mathcal{S}, \mathcal{F})$ die abstrakte Algebra durch $(\tau, (\mathcal{S}, \mathcal{F}), Q)$ angegeben.

Mit Axiomen umformen

Definition 14

Mit Axiomen umformen bedeutet, einen Term s_1 unter Verwendung eines Axioms $t_1 \to t_2$ in einen Term s_2 umzuformen. Wir schreiben $s_1 \to s_2$, wenn gilt:

- s_1 und s_2 stimmen bis auf einen Unterterm an einer entsprechenden Position überein, d.h. es gibt einen Term s, in dem die Variable v genau einmal vorkommt, und $s_1 = s[v/r_1]$, $s_2 = s[v/r_2]$.
- Es gibt eine Substitution σ mit $t_1\sigma = r_1$ und $t_2\sigma = r_2$.

Ein Term s ist in t umformbar, wenn es eine Folge von Termen $s = s_0, s_1, \ldots, s_n = t$ mit $s_{i-1} \to s_i$ für alle $i = 1, \ldots, n$ gibt; wir schreiben dann $s \to t$.

$\to$ ist offensichtlich transitiv. Die Axiome sollten so gewählt werden, dass $\to$ auch irreflexiv ist. In diesem Fall ist $\to$ eine Halbordnung.

Mit Axiomen umformen

Beispiel:

$s_1 = f_\neg(f_\wedge(true, f_\vee(x, y)))$ Axiom: $Q_3 : f_\wedge(true, x) \rightarrow x$

$r_1 = f_\wedge(true, f_\vee(x, y)) = t_1\sigma = f_\wedge(true, x)[x/f_\vee(x, y)]$
$r_2 = t_2\sigma = x[x/f_\vee(x, y)]) = f_\vee(x, y)$

Also sei $s_2 = f_\neg(f_\vee(x, y))$, dann ist s_1 nach s_2 umformbar.

Konkrete Algebra

Definition 15

Sei $((\mathcal{S}, \mathcal{F}), Q)$ mit $\mathcal{S} = \{S_1, \ldots, S_n\}$ und den zugehörigen Variablenmengen V_i. Eine **konkrete Algebra** (Ω, F, Q) besteht aus:

1. Jeder Sorte S_i ordnen wir einen nicht-leeren Grundbereich ω_i zu.
2. Ω ist die Menge der Grundbereiche ω_i inklusive der Zuordnung.
3. Jeder Operation $f : S_1\ x \ldots x\ S_r \rightarrow S_k$ ordnen wir eine konkrete Funktion $f^* : \omega_1\ x \ldots, x\ \omega_r \rightarrow \omega_k$ zu.
4. F ist die Menge der konkreten Funktionen.
5. Der Einfachheit halber seien die Variablen der Grundbereiche ω_i gerade die Variablen der Sorte S_i.

Sprechweise: Die konkrete Algebra wird auch als Modell der abstrakten Algebra bezeichnet.

Modell, abstrakte Algebra

Definition 16

Sei $((\mathcal{S}, \mathcal{F}), Q)$ eine abstrakte Algebra und sei (Ω, F, Q) eine (passende) zugehörige konkrete Algebra.
(Ω, F, Q) ist ein Modell der abstrakten Algebra genau dann, wenn für alle Axiome $t_1 \rightarrow t_2$ aus Q gilt: Nach Ersetzung aller Operationen in t_1 und t_2 durch die zugehörigen konkreten Funktionen, die Ergebnisse der Ersetzung seien t_1^* bzw. t_2^*, muss $t_1^* = t_2^*$ gelten.
(Für die konkrete Algebra gilt im Fall von Axiomen die Gleichheit der linken und rechten Seite der Axiome.)

Algebren

Beispiel:

Signatur $\Sigma = (\{A\}, \mathcal{F})$ und Variablenmenge $V = \{x_1, \ldots, x_n\}$ mit
$\mathcal{F} = \{f : A \ x \ A \rightarrow A, h : A \ x \ A \rightarrow A\}$ sowie Axiome
$\{h(f(x, y), z) \rightarrow f(h(x, z), h(y, z))\}$

Konkrete Algebra:
$(\mathbb{N}, \{+ : \mathbb{N} x \mathbb{N} \rightarrow \mathbb{N}, \ * : \mathbb{N} x \mathbb{N} \rightarrow \mathbb{N}\}, \{*(+(x, y), z) \rightarrow +(*(x, z), *(y, z))\})$

Grundbereich: $\mathbb{N}$
Funktionen: f wird die Addition $+$ zugeordnet,
h wird die Multiplikation $*$ zugeordnet
Das einzige Axiom ist $h(f(x, y), z) \rightarrow f(h(x, z)h(y, z))$

Dann gilt für das Axiom

$*(+(x, y), z) = +(*(x, z), *(y, z))$ (Assoziativgesetz)

Algebren

Beispiel: Abstrakte Algebra *BOOL*

Signatur $\Sigma = (\{BOOL\}, \mathcal{F})$.

$$
\begin{aligned}
\mathcal{F} = \{ \quad & true : \quad \rightarrow BOOL \\
& false : \quad \rightarrow BOOL \\
& f_\wedge : BOOL \ x \ BOOL \rightarrow BOOL \\
& f_\vee : BOOL \ x \ BOOL \rightarrow BOOL \\
& f_\neg : BOOL \rightarrow BOOL \qquad \}
\end{aligned}
$$

Axiome: Seien x und y Variable der Sorte *BOOL*.

$Q_1 : f_\neg(true) \rightarrow false$
$Q_2 : f_\neg(false) \rightarrow true$
$Q_3 : f_\wedge(true, x) \rightarrow x$
$Q_4 : f_\wedge(false, x) \rightarrow false$
$Q_5 : f_\vee(x, y) \rightarrow f_\neg(f_\wedge(f_\neg(x), f_\neg(y)))$

Algebren

Beispiel: Konkrete Algebra für *BOOL*

Signatur $\Sigma = (\{BOOL\}, \mathcal{F})$. $\qquad \omega := Bool := \{0,1\}$

$$
\begin{array}{lll}
\mathcal{F} = \{ & true : \; \rightarrow BOOL & true^* := 1 \\
 & false : \; \rightarrow BOOL & false^* := 0 \\
 & f_\wedge : BOOL \text{ x } BOOL \rightarrow BOOL & f_\wedge^*(v_1, v_2) := min(v_1, v_2) \\
 & f_\vee : BOOL \text{ x } BOOL \rightarrow BOOL & f_\vee^*(v_1, v_2) := max(v_1, v_2) \\
 & f_\neg : BOOL \rightarrow BOOL \qquad \} & f_\neg^*(0) := 1, f_\neg^*(1) := 0
\end{array}
$$

Axiome: Seien x und y Variable der Sorte *BOOL*.

$Q_1 : f_\neg(true) \rightarrow false$
$Q_2 : f_\neg(false) \rightarrow true$
$Q_3 : f_\wedge(true, x) \rightarrow x$
$Q_4 : f_\wedge(false, x) \rightarrow false$
$Q_5 : f_\vee(x, y) \rightarrow f_\neg(f_\wedge(f_\neg(x), f_\neg(y)))$

Für die Axiome gilt die Gleichheit zwischen linkem und rechtem Term.

Algebren

Beispiel: Alternative konkrete Algebra für *BOOL*

Für die abstrakte Algebra aus der vorherigen Folie ist die folgende konkrete Algebra ein Modell:

1. Der Grundbereich sei $\omega = \{\emptyset, \{1\}\}$ (leere Menge und Einermenge)
2. Konstante $\emptyset$ für *false*
3. Konstante $\{1\}$ für *true*
4. Mengendurchschnitt $\cap$ für $f_\wedge$
5. Mengenvereinigung $\cup$ für $f_\vee$
6. Mengenkomplement bzgl. $\{1\}$ für $f_\neg$

Algebren

Beispiel: Abstrakte Algebra für *Keller*

Signatur $\Sigma = (S, \mathcal{F})$
Sorten $\mathcal{S} = \{Keller, Element, BOOL\}$

$$
\begin{aligned}
\mathcal{F} = \{ \quad createStack &: \quad \rightarrow Keller \\
push &: Keller \text{ x } Element \rightarrow Keller \\
pop &: Keller \rightarrow Keller \\
top &: Keller \rightarrow Element \\
empty &: Keller \rightarrow BOOL\}
\end{aligned}
$$

Axiome:
$K_1 : empty(createStack) \rightarrow true$
$K_2 : empty(push(k, e)) \rightarrow false$
$K_3 : pop(push(k, e)) \rightarrow k$
$K_4 : top(push(k, e)) \rightarrow e$

Algebren

Beispiel: Konkrete Algebra als Modell für *Keller*

Grundbereiche:

BOOL	ist	$Bool = \{0, 1\}$
Element	ist	$\mathbb{N}_0$
Keller	ist	$\mathbb{N}_0^*$, Menge der endlichen Folgen natürlicher Zahlen

Funktionen:

createStack	$newSeries: \ \rightarrow \mathbb{N}_0^*$	$newSeries() = ()$
push	$append: \mathbb{N}_0^* \times \mathbb{N}_0 \rightarrow \mathbb{N}_0^*$	$append((a_1, \ldots, a_n), x) = (a_1, \ldots, a_n, x)$
pop	$remove: \mathbb{N}_0^* \rightarrow \mathbb{N}_0^*$	$remove((a_1, \ldots, a_n, x)) = (a_1, \ldots, a_n)$
top	$last: \mathbb{N}_0^* \rightarrow \mathbb{N}_0$	$last((a_1, \ldots, a_n, x)) = x$
empty	$noElem: \mathbb{N}_0^* \rightarrow \{0, 1\}$	$noElem(f) = 1$ gdw. f ist leere Folge.

Es bleibt noch noch zu zeigen, dass die Gleichheit für die Axiome gilt.