Grundlagen der Programmierung

Vorlesung 5 vom 11.11.2004

Sebastian Iwanowski
FH Wedel

Grundlagen der Programmierung

1. Einführung
   - Grundlegende Eigenschaften von Algorithmen und Programmen
2. Logik
   - Aussagenlogik
   - ➔ Prädikatenlogik
3. Programmentwicklung und –verifikation
   - Grundlagen der Programmverifikation
   - Verbundanweisungen
   - Verzweigungen
   - Schleifen
   - Modularisierung
   - Rekursion
4. Entwurf und Analyse von Algorithmen
   - Klassifikation von Algorithmen
   - Programmierung von Algorithmen
   - Bewertung von Algorithmen

Prädikatenlogik

- **Variable**
- **Prädikate**
- **Funktionen**
- **Quantoren**

Der Existenzquantor $\exists \mathbf{x}$ (...) drückt aus, dass es einen Wert für $\mathbf{x}$ gibt, der den dahinter stehenden Ausdruck zu einer wahren Aussage macht.

Der Allquantor $\forall \mathbf{x}$ (...) drückt aus, dass jeder Wert für $\mathbf{x}$ den dahinter stehenden Ausdruck zu einer wahren Aussage macht.

Die Definitionsbereiche für die Variablen dürfen eingeschränkt werden:

Für den Existenzquantor ist das eine Verschärfung,
für den Allquantor eine Abschwächung der Aussage.

Bsp.:

$$\exists\, x \in R\ \exists\, y \in R\ ((2 < x < 4) \wedge (0 < y < 6) \wedge (x + y > 7) \wedge (x \bullet y < 10))$$

$$\forall\, x \in R\ (x \bullet x \geq 0)$$

Prädikatenlogische Formeln

- **Eine prädikatenlogische Formel 1. Stufe ist eine Verknüpfung von endlich vielen Variablen, Funktionen und Prädikaten mit aussagenlogischen Operatoren oder Quantoren, die sich nur auf Variable beziehen.**

Bsp.: $\forall x \, (R(\mathbf{y}, \mathbf{z}) \wedge \exists y \, (\neg P(\mathbf{y}, \mathbf{x}) \vee R(\mathbf{y}, \mathbf{z})))$

Grüne Vorkommen von y und z sind **frei**.
Rote Vorkommen von x, y und z sind **gebunden**.

Geschlossene Formeln:
Formeln, die keine freien Variablen enthalten.

Offene Formeln:
Formeln, die keine gebundenen Variablen enthalten.

Prädikatenlogische Formeln

- **Eine Belegung einer Formel ist eine Zuweisung von *Werten aus festgelegten Definitionsbereichen an die freien Variablen* derart, dass dieselben Variablen immer denselben Wert erhalten.**

- **Eine Formel heißt erfüllbar, wenn es eine Belegung gibt derart, dass die Formel wahr ist.**

☹

  - Das Erfüllbarkeitsproblem ist in der Prädikatenlogik nicht entscheidbar, d.h. kein Algorithmus kann jemals in der Lage sein, von jeder Formel zu entscheiden, ob sie erfüllbar ist oder nicht.

Das allgemeine Problem ist unlösbar !

Prädikatenlogik

Beschreibung der semantischen Eigenschaft einer Formel $F(x_1, x_2, ..., x_k)$ mit prädikatenlogischen Mitteln:

- **F ist erfüllbar:**

  $$\exists\, x_1, x_2, ..., x_k \quad (F(x_1, x_2, ..., x_k) \leftrightarrow \top)$$

- **F ist widersprüchlich:**

  $$\forall\, x_1, x_2, ..., x_k \quad (F(x_1, x_2, ..., x_k) \leftrightarrow \bot)$$

- **F ist gültig (Tautologie):**

  $$\forall\, x_1, x_2, ..., x_k \quad (F(x_1, x_2, ..., x_k) \leftrightarrow \top)$$

- **F ist widerlegbar (falsifizierbar):**

  $$\exists\, x_1, x_2, ..., x_k \quad (F(x_1, x_2, ..., x_k) \leftrightarrow \bot)$$

Was fällt auf ?

Prädikatenlogik

„Rechenregeln“ für die Quantoren:

$\neg\forall x\,(F(x)) \Leftrightarrow \exists x\,(\neg F(x))$

$\neg\exists x\,(F(x)) \Leftrightarrow \forall x\,(\neg F(x))$

Verallgemeinerung von deMorgan

$\forall x\ \forall y\,(F(x,y)) \Leftrightarrow \forall y\ \forall x\,(F(x,y))$

$\exists x\ \exists y\,(F(x,y)) \Leftrightarrow \exists y\ \exists x\,(F(x,y))$

Vertauschung gleicher Quantoren

Was gilt bei der Vertauschung verschiedener Quantoren ? ($\Leftrightarrow$, $\Rightarrow$, $\Leftarrow$, $\not\Leftrightarrow$)

$\exists x\ \forall y\,(F(x,y))$ $\qquad$ $\forall y\ \exists x\,(F(x,y))$

$\exists y\ \forall x\,(F(x,y))$ $\qquad$ $\forall x\ \exists y\,(F(x,y))$

Prädikatenlogik

„Rechenregeln“ für die Quantoren:

Was gilt bei Hineinziehen von Quantoren in $\wedge$ oder $\vee$? ($\Leftrightarrow, \Rightarrow, \Leftarrow, \not\Leftrightarrow$)

$\forall x\, (F(x)) \vee \forall x\, (G(x))$ $\quad\quad$ $\forall x\, (F(x) \vee G(x))$

$\forall x\, (F(x)) \wedge \forall x\, (G(x))$ $\quad\quad$ $\forall x\, (F(x) \wedge G(x))$

$\exists x\, (F(x)) \vee \exists x\, (G(x))$ $\quad\quad$ $\exists x\, (F(x) \vee G(x))$

$\exists x\, (F(x)) \wedge \exists x\, (G(x))$ $\quad\quad$ $\exists x\, (F(x) \wedge G(x))$

Negation	von Prädikaten mit Quantoren
Aussage	*„alle Aldi-PCs sind schlecht gebaut“*
Frage	Welche der folgenden Aussagen ist die Negation dieser Aussage
(1)	Alle nicht-Aldi-PCs sind gut gebaut
(2)	Alle Aldi-PCs sind gut gebaut
(3)	Einige Aldi-PCs sind gut gebaut
(4)	Einige Aldi-PCs sind schlecht gebaut
(5)	Einige nicht-Aldi-PCs sind schlecht gebaut
(6)	Einige nicht-Aldi-PCs sind gut gebaut

Beispiel

- Es seien Animal, Vegetarian, Sheep, Cow, und Mad_cow einstellige Prädikatensymbole. Die Symbole EATS und PART_OF seien zweistellige Prädikatensymbole, und es seien x, y, z Variablen. Wir betrachten folgende Menge T von Formeln:
- $$
\begin{aligned}
T := \{ & \forall x\, (Sheep(x) \rightarrow \\
& \qquad (Animal(x) \land Vegetarian(x))), \\
& \forall x\, (Cow(x) \rightarrow \\
& \qquad (Animal(x) \land Vegetarian(x))), \\
& \forall x\, (Vegetarian(x) \rightarrow \\
& \qquad (\neg\exists y\, (EATS(x,y) \land Animal(y)) \land \\
& \qquad \forall y\, (EATS(x,y) \rightarrow \\
& \qquad\qquad (\neg\exists z\, (PART_OF(y,z) \land Animal(z)))))\ \}
\end{aligned}
$$

Beispiel (2)

- Annahme: Formeln seien Grundlage für die Spezifikation von Programmen für ein großes Softwareprojekt für das Landwirtschaftsministerium dar.
- Frage: Welche Kühe (eines bestimmten Bestandes, der als Parameter eingeht) sind beim Verzehr gesundheitsgefährdend?
- Hierzu wird folgende Formel verwendet:

$$
\begin{aligned}
f := \forall x\ (\mathrm{Mad\_cow}(x) \;\Leftrightarrow\; & \\
& (\mathrm{Cow}(x) \wedge \exists y\ (\mathrm{EATS}(x, y) \wedge \\
& \qquad \exists z\ \mathrm{PART\_OF}(y, z) \wedge \mathrm{Sheep}(z))))
\end{aligned}
$$

Beispiel (3)

- Zu erstellen: Programm, das die Menge M aller x berechnen soll, für die Mad_cow(x) gilt
- Nach eingehender Beratung mit Ihren Mitarbeitern kommen Sie zu dem Schluß, daß Sie das Projekt, obwohl es lukrativ sein mag, aus Gewissensgründen nicht annehmen werden, da die Menge M immer leer sein wird.
- Sie zeigen, daß die Formel □x Mad_cow(x) unerfüllbar bzgl. $T \cup \{f\}$ ist

Prädikatenlogik

Arithmetische Vergleichsprädikate:

Präfix-Notation	Infix-Notation	Postfix-Notation
(Standard in Prädikatenlogik)	(Standard in Arithmetik)	(Standard auf alten Taschenrechnern ohne Klammern)
`lessThan (x,y)`	`x < y`	`x, y, <`
`equal (x,y)`	`x = y`	`x, y, =`

Mit diesen beiden Prädikaten lassen sich alle anderen Vergleichsprädikate bilden:

$x \leq y$

$x \geq y$

$x \neq y$

$x > y$

Wie drückt man mit diesen Prädikaten aus, dass eine Zahl x zwischen y und z liegt ?

Prädikatenlogik

Arithmetische Vergleichsprädikate:

Bilde das Gegenteil von:

1) $(x>0) \lor ((y+x)\leq 0)$

2) $\forall y<0\ ((x>0) \lor ((y+x)\leq 0))$

Beim nächsten Mal:

Programmentwicklung und -verifikation