

Tabment-Orientierte Programmierung

Klaus Benecke

Karlsruhe DMV 2019

Inhalt

- Rechnen wir zu umständlich?
- Magdeburger Notation für Terme
- bisherige Notation für Terme
- Spezifikation Tabment
- o++o Programme
- Thesen
- weitere methodische Verbesserungen
- Mathelehrplan bis Klasse 10

Rechnen wir zu umständlich? (1)

$$1 + 2$$

$$* 3 = ?$$

$$1$$

$$+ 2 * 3 = ?$$

Rechnen wir zu umständlich? (2)

$$1 + 2$$

$$* 3 + 4 = ?$$

$$\mathbf{1 + 2 * 3 = 9}$$

- Schüler der zweiten Klasse
- viele Taschenrechner
- Smalltalk
- alle Erwachsenen, die eine solche Formel lange nicht zu Gesicht bekamen

Rechnen wir zu umständlich? (3)

ALT	NEU
sum(<TAB! X, Y 1 2 3 4 !TAB>)	<TAB! X, Y 1 2 3 4 !TAB> ++

Rechnen wir zu umständlich? (4)

ALT	NEU
<code>sqrt(abs(cos((pi+3)*2)))</code>	pi+3*2 cos abs sqrt
<code>List.map (function x → float x *. 1.19) [1;2;3]</code>	1 2 3 * 1.19
$\ln(\sin^2(x)) = \ln((\sin(x))^2)$	X sin hoch 2 ln
<code>sqrt(sin(x) + cos(2))</code>	X sin + (2 cos) sqrt

Magdeburger Notation für Terme

- von oben nach unten (MN1)
- von links nach rechts (MN2)
- Klammern zuerst (MN3)
- jede Operation folgt dem ersten Argument (! ist Trennzeichen für weitere Argumente) (MN4)

bisherige Notation für Terme

- von oben nach unten (nicht existent)
- von links nach rechts (bei + - und bei * : Folgen)
- *hoch* von rechts nach links (rechtsassoziativ)
- Klammern zuerst
- binäre Operationen (+,...) folgen dem ersten Argument
- weitere binäre (unäre,...) Operationen präfix (d.h. von rechts nach links: $f(g(x))$)
- Vorrangregel: * : vor + -
- Vorrangregel: hoch vor * : + -
- * wird oft nicht geschrieben (implizit)

Spezifikation Tabment (1)

```
type value =  
  Bar  
  | Int_v   of big_int  
  | Float_v of float  
  | Bool_v  of bool  
  | String_v of string  
  
  | Stringi_v of string  
  
  | Ratio_v of big_int * big_int;;
```

Spezifikation Tabment (2)

```
type coll_sym = Set | Bag | List | Any  
  | S1    (* optionale Werte *)  
  | Set_minus | Bag_minus | List_minus;;  
  (* abwärts sortieren *)
```

```
type name = string;;  
  (* Spaltennamen, Tags *)
```

Spezifikation Tabment (3)

- type **scheme** =
- Empty_s
- | Inj of name
- | Coll_s of coll_sym * scheme
- | Tuple_s of scheme list
- | Alternate_s of scheme list;;

Spezifikation Tabment (4)

type **tabment** =

Empty_t

| El_tab of value

Jeder elementare Wert ist ein Tabment

| Tuple_t of tabment list

| Coll_t of (coll_sym * scheme * (tabment list))

| Tag0 of name * tabment

| Alternate_t of (scheme list) * tabment;;

o++o-Programme (1)

- 2 4 7 ++ = 13
- 1 2 1 3 4 ++: = 2.2
- 32.3 87.4 77.6 * 1.19 = 38.437 104.006 92.344
- 1 .. 100 ++ = 5050
- 1 ... 3 ! 0.0001 sin max = 0.99999999999993
- 0 ... pi ! 0.0001 cos abs *0.0001 ++ = 2.0000073451

o++o-Programme (2)

- geg.: noten.tab
- NAME, (FACH, NOTE1 m) m

Paul	Ma	1	2	1	3	1
------	----	---	---	---	---	---

	Phy	1	3	2	1
--	-----	---	---	---	---

Sophia	Eng	1	2	3	2	1
--------	-----	---	---	---	---	---

	Ma	1	1	2	1
--	----	---	---	---	---

o++o-Programme (3)

```
NAME, (FACH, NOTE1 m) m
Paul  Ma    1 2 1 3 1
      Phy   1 3 2 1
Sophia Eng   1 2 3 2 1
      Ma    1 1 2 1
```

```
noten.tab
gib FACH,(NAME, NOTE1 m) m
```

```
FACH, (NAME, NOTE1 m) m
Eng   Sophia 1 2 3 2 1
Ma    Paul    1 2 1 3 1
      Sophia 1 1 2 1
Phy   Paul    1 3 2 1
```

o++o-Programme (4)

noten.tab

DUR:=NOTEI ++:

NAME,	(FACH,	DUR,	NOTEI	m)	m
Paul	Ma	1.6	1 2 1	3	1
	Phy	1.75	1 3 2	1	
Sophia	Eng	1.8	1 2 3	2	1
	Ma	1.25	1 1 2	1	

0++0-Programme (5)

- 1/2 3/4 5/10 6/2 3 ++

= 31/4

- 1/2 3/4 5/10 6/2 3

gib RATIO, RATIOb m

=

1/2 1/2

5/10

3/4 3/4

6/2 6/2

3/1

weitere methodische Verbesserungen

ALT	NEU
Stadt="Karlsruhe"	STADT=Karlsruhe
in Ocaml abstrakt	{1 2 1}
M(FACH,LEHRER?,L(NOTE))	FACH,LEHRER?,NOTE m

Thesen

- Jeder soll neben Lesen, Schreiben und Rechnen auch Programmieren lernen.
 - (Endnutzer-) Programmiersprachen sollten im Mathe unterrichtet werden.
 - hinter jedem Klick sollte ein lesbares Programm stehen
 - Merkfähigkeit und Universalität der Konzepte
 - Symbolik möglichst aus Mathematik übernehmen ({)
 - mengenweises Rechnen
 - strukturierte Tabellen und Dokumente

Mathelehrplan bis Klasse 10 (1) ?

könnte entfallen	müsste hinzukommen
Tabellenkalkulation	Anfragen an Datenbanken, Dokumente, Wikipedia,...
Nullstellen quadratischer Polynome	näherungsweise Nullstellenbestimmung beliebiger Funktionen
Vorrangregeln	
	Anwendungen der Differential- und Integralrechnung (max, min, Fläche unter Kurve)

Mathelehrplan bis Klasse 10 (2) ?

könnte entfallen	müsste hinzukommen
Divisionsalgorithmus für Dezimalzahlen	Selektionsalgorithmus, mengenweise Berechnungen
	ein Sortier- Umstrukturierungsalgorithmus
	Unterschiede: Liste, Menge, Bag, Tupel
	Prozentrechnung reduzieren: * 0.19 oder * 1.19
Taschenrechner, grafikfähiger Taschenrechner	erweiterter (o++o)-Taschenrechner

**Vielen Dank für Ihre
Aufmerksamkeit!**