

Inhaltsverzeichnis

Geleitwort	xi
Vorwort	xvii
Danksagung	xx
1 Konstruktion von Abstraktionen mit Prozeduren	1
1.1 Die Programmelemente	4
1.1.1 Ausdrücke	5
1.1.2 Namen und Umgebungen	8
1.1.3 Auswertung von Kombinationen	10
1.1.4 Zusammengesetzte Prozeduren	13
1.1.5 Das Substitutionsmodell für Prozeduranwendungen . .	15
1.1.6 Bedingte Ausdrücke und Prädikate	18
1.1.7 Beispiel: Berechnung der Quadratwurzel nach dem Newtonschen Iterationsverfahren	22
1.1.8 Prozeduren als "Black-box"-Abstraktionen	27
1.2 Prozeduren und Prozesse	33
1.2.1 Lineare Rekursion und Iteration	35
1.2.2 Baumrekursion	40
1.2.3 Größenordnungen	45
1.2.4 Potenzrechnung	47
1.2.5 Der größte gemeinsame Teiler	50
1.2.6 Beispiel: Primzahlen	52
1.3 Abstraktionen mit Prozeduren höherer Ordnung	60
1.3.1 Prozeduren als Parameter	61
1.3.2 Konstruktion von Prozeduren mit <code>lambda</code>	65
1.3.3 Prozeduren als allgemeine Methode	70
1.3.4 Prozeduren als Ergebnis	77

2	Konstruktion von Abstraktionen mit Daten	83
2.1	Einführung in die Datenabstraktion	87
2.1.1	Beispiel: Arithmetische Operatoren für rationale Zahlen	87
2.1.2	Abstraktionsbarrieren	92
2.1.3	Was sind eigentlich Daten?	95
2.1.4	Beispiel: Arithmetik mit Intervallen	99
2.2	Datenhierarchien	104
2.2.1	Darstellung von Sequenzen	106
2.2.2	Darstellung von Baumstrukturen	113
2.2.3	Symbole und die Notwendigkeit des <code>quote</code>	118
2.2.4	Beispiel: Symbolisches Differenzieren	123
2.2.5	Beispiel: Darstellung von Mengen	129
2.2.6	Beispiel: Huffman-Bäume	140
2.3	Mehrfachdarstellungen abstrakter Daten	150
2.3.1	Darstellungen komplexer Zahlen	153
2.3.2	Manifeste Typen	157
2.3.3	Datengesteuerte Programmierung	162
2.4	Systeme mit generischen Operatoren	169
2.4.1	Generische Arithmetikoperatoren	170
2.4.2	Kombination von Operanden verschiedenen Typs . . .	175
2.4.3	Beispiel: Symbolische Algebra	183
3	Modularität, Objekte und Zustände	199
3.1	Zuweisungen und lokale Zustände	200
3.1.1	Lokale Zustandsvariablen	201
3.1.2	Der Preis der Zuweisung	209
3.1.3	Die Vorzüge der Zuweisung	214
3.2	Das Umgebungsmodell der Auswertung	220
3.2.1	Die Auswertungsregeln	221
3.2.2	Auswertung einfacher Prozeduren	225
3.2.3	Rahmen als Ort, an dem sich der lokale Zustand befindet	228
3.2.4	Interne Definitionen	234
3.3	Modelle mit veränderbaren Daten	237
3.3.1	Veränderbare Listenstrukturen	238
3.3.2	Darstellung von Warteschlangen	249
3.3.3	Darstellung von Tabellen	256
3.3.4	Ein Simulator für digitale Schaltkreise	263
3.3.5	Propagierung von Beschränkungen	276

3.4	Datenströme	290
3.4.1	Datenströme als Standardschnittstellen	290
3.4.2	Prozeduren höherer Ordnung für Datenströme	296
3.4.3	Implementierung von Datenströmen	309
3.4.4	Datenströme unendlicher Länge	318
3.4.5	Datenströme und verzögerte Auswertung	330
3.4.6	Mit Datenströmen Modelle von lokalen Zuständen bilden	343
4	Metalinguistische Abstraktion	351
4.1	Der metazirkuläre Evaluator	355
4.1.1	Das Kernstück des Evaluators	357
4.1.2	Darstellung von Ausdrücken	362
4.1.3	Bearbeitung von Umgebungen	366
4.1.4	Der Evaluator als Lisp-Programm	371
4.1.5	Behandlung von Ausdrücken als Programm	374
4.2	Variationen zu einem Schema	378
4.2.1	Auswertung in normaler Reihenfolge	379
4.2.2	Andere Arten der Bindung	383
4.3	Module	390
4.3.1	Verwendung von Umgebungen zum Erzeugen von Modulen	391
4.3.2	Module in einem System für generische Arithmetik	395
4.4	Logikprogrammierung	403
4.4.1	Deduktiver Informationsabruf	406
4.4.2	Wie das Anfragesystem funktioniert	418
4.4.3	Ist Logikprogrammierung Mathematische Logik?	428
4.5	Implementierung des Anfragesystems	435
4.5.1	Die Treiberschleife und Instantiierung	435
4.5.2	Der Evaluator	437
4.5.3	Finden von Aussagen durch Mustervergleich	441
4.5.4	Regeln und Unifikation	444
4.5.5	Die Datenbank	448
4.5.6	Weitere nützliche Prozeduren	452
5	Rechnen mit Registermaschinen	461
5.1	Entwurf von Registermaschinen	463
5.1.1	Eine Sprache zur Beschreibung von Registermaschinen	466
5.1.2	Abstraktion beim Entwurf von Maschinen	471
5.1.3	Unterprogramme	475

5.1.4	Implementierung der Rekursion mit einem Keller . . .	480
5.1.5	Ein Simulator für Registermaschinen	488
5.2	Der Evaluator mit expliziter Steuerung	505
5.2.1	Der Kern des Evaluators mit expliziter Steuerung . . .	507
5.2.2	Auswerten von Sequenzen und Endrekursion	515
5.2.3	Bedingte Ausdrücke und andere Sonderformen	519
5.2.4	Laufenlassen des Evaluators	522
5.2.5	Interne Definitionen	527
5.3	Übersetzen	532
5.3.1	Die Struktur des Übersetzers	535
5.3.2	Übersetzen von Ausdrücken	539
5.3.3	Datenstrukturen des Übersetzers	555
5.3.4	Elementare Codegeneratoren	560
5.3.5	Ein Beispiel für übersetzten Code	569
5.3.6	Die Schnittstelle zwischen übersetztem Code und dem Evaluator	576
5.3.7	Lexikalische Adressierung	582
5.4	Speicherzuteilung und Speicherbereinigung	589
5.4.1	Speicher als Vektoren	590
5.4.2	Als ob der Speicher unendlich wäre	595
	Anhang zur deutschen Ausgabe	605
	Literaturverzeichnis	607
	Stichwortverzeichnis	613