
Inhaltsverzeichnis

1	Einleitung	1
1.1	Höhere Programmiersprachen	1
1.2	Implementierung von Programmiersprachen	2
1.2.1	Interpreter	2
1.2.2	Übersetzer	3
1.2.3	Reale und virtuelle Maschinen	4
1.2.4	Kombinationen von Übersetzung und Interpretation	5
1.3	Allgemeine Literaturhinweise	6
2	Imperative Programmiersprachen	7
2.1	Sprachkonzepte und ihre Übersetzung	7
2.2	Die Architektur der C-Maschine	8
2.3	Einfache Ausdrücke und Wertzuweisungen	10
2.4	Anweisungen und Anweisungsfolgen	15
2.5	Bedingte und iterative Anweisungen	16
2.6	Speicherbelegung für Variablen einfachen Typs	22
2.7	Speicherbelegung für Felder und Strukturen	23
2.8	Zeiger und dynamische Speicherbelegung	28
2.9	Funktionen	34
2.9.1	Die Speicherorganisation der C-Maschine	37
2.9.2	Der Umgang mit lokalen Variablen	39
2.9.3	Betreten und Verlassen von Funktionen	42
2.10	Übersetzung ganzer Programme	48
2.11	Aufgaben	51
2.12	Liste der Register der CMA	57
2.13	Liste der Codeerzeugungsfunktionen der CMA	57
2.14	Liste der CMA-Instruktionen	58
2.15	Literaturhinweise	58

3	Funktionale Programmiersprachen	59
3.1	Sprachtyp und einleitende Beispiele	59
3.2	Eine einfache funktionale Programmiersprache	62
3.3	Die Architektur der MaMa	66
3.4	Die Übersetzung einfacher Ausdrücke	68
3.5	Der Zugriff auf Variablen	71
3.6	<i>let</i> -Ausdrücke	75
3.7	Funktionsdefinitionen	76
3.8	Funktionsanwendungen	79
3.9	Unter- und Überversorgung mit Argumenten	82
3.10	Rekursive Variablendefinitionen	86
3.11	Abschlüsse und ihre Auswertung	89
3.12	Optimierungen I: Globale Variablen	92
3.13	Optimierungen II: Abschlüsse	93
3.14	Die Übersetzung eines Programmausdrucks	94
3.15	Strukturierte Daten	95
3.15.1	Tupel	95
3.15.2	Listen	97
3.15.3	Abschlüsse für Tupel und Listen	100
3.16	Optimierungen III: Letzte Aufrufe	101
3.17	Aufgaben	102
3.18	Liste der Register der MaMa	106
3.19	Liste der Codeerzeugungsfunktionen der MaMa	107
3.20	Liste der MaMa-Instruktionen	107
3.21	Literaturhinweise	107
4	Logische Programmiersprachen	109
4.1	Die Sprache ProL	109
4.2	Die Architektur der WiM	112
4.3	Anlegen von Termen in der Halde	114
4.4	Die Übersetzung von Literalen	118
4.5	Unifikation	120
4.6	Klauseln	130
4.7	Die Übersetzung von Prädikaten	131
4.7.1	Backtracking	132
4.7.2	Zusammenfügung	134
4.8	Die Endbehandlung von Klauseln	135
4.9	Anfragen und Programme	137
4.10	Optimierung I: Letzte Ziele	140
4.11	Optimierung II: Verkleinerung von Kellerrahmen	143
4.12	Optimierung III: Klausel-Indizierung	144
4.13	Erweiterung: Der Cut Operator	147
4.14	Exkurs: Speicherbereinigung	149
4.15	Aufgaben	154
4.16	Liste der Register der WiM	157

4.17	Liste der Codeerzeugungsfunktionen der WiM	157
4.18	Liste der WiM-Instruktionen	157
4.19	Literaturhinweise	158
5	Objektorientierte Programmiersprachen	159
5.1	Konzepte objektorientierter Sprachen	159
5.1.1	Objekte	159
5.1.2	Objektklassen	160
5.1.3	Vererbung	161
5.1.4	Generizität	162
5.1.5	Informationskapselung	163
5.1.6	Zusammenfassung	163
5.2	Eine objektorientierte Erweiterung von C	164
5.3	Die Speicherorganisation für Objekte	165
5.4	Methodenaufrufe	168
5.5	Die Definition von Methoden	171
5.6	Die Verwendung von Konstruktoren	171
5.7	Die Definition von Konstruktoren	173
5.8	Ausblick: Mehrfache Beerbung	175
5.9	Aufgaben	177
5.10	Liste zusätzlicher Register	182
5.11	CMA-Instruktionen für Objekte	182
5.12	Literaturhinweise	182
	Literaturverzeichnis	183
	Sachverzeichnis	187



<http://www.springer.com/978-3-540-49596-3>

Übersetzerbau

Virtuelle Maschinen

Wilhelm, R.; Seidl, H.

2007, XIII, 192 S., Softcover

ISBN: 978-3-540-49596-3