

Inhaltsverzeichnis

Vorwort	7
1 Elemente funktionaler Programmierung	15
1.1 Werte und Ausdrücke	15
1.2 Tupel, Records und Listen	21
1.3 Variablen und Definitionen	25
1.4 Funktionen	31
1.5 Typen und Polymorphismus	37
1.6 Datentypen und Pattern Matching	42
1.7 Funktionen höherer Ordnung	47
1.8 Literaturhinweise	52
2 Verzögerte Auswertung	53
2.1 Parameterübergabe und Auswertungsstrategien	53
2.2 Unendliche Datenstrukturen	57
2.3 Literaturhinweise	59
3 Verifikation und Programm-Transformation	61
3.1 Induktion	61
3.1.1 Korrektheit rekursiv definierter Funktionen	62
3.1.2 Strukturelle Induktion auf Datentypen	66
3.1.3 Wohlfundierte Relationen und wohlfundierte Induktion	71
3.2 Optimierung mit dem Unfold/Fold-Verfahren	75
3.3 Der Bird/Meertens-Formalismus	83
3.3.1 Beispiel: Das Auswahlproblem	83
3.3.2 Eigenschaften von map und foldl	86
3.3.3 Berechnung von Listensegmenten mittels scanl	88
3.3.4 Optimale Berechnung der maximalen Teilsumme	91
3.4 Literaturhinweise	93
4 Der Lambda-Kalkül	95
4.1 Die Syntax des Lambda-Kalküls	95
4.2 Reduktion und Normalformen	99
4.3 Repräsentation von Datentypen	107

4.4	Rekursion	109
4.5	Ausdrucksstärke des Lambda-Kalküls	112
4.6	Literaturhinweise	115
5	Typisierung und Typinferenz	117
5.1	Einführung in Typsysteme	117
5.2	Ein polymorphes Typinferenzsystem	120
5.3	Automatische Typinferenz	128
5.4	Literaturhinweise	133
6	Implementierungstechniken	135
6.1	Implementierung von Programmiersprachen	135
6.2	Ein Interpreter für Mini-ML	137
6.2.1	Interne Repräsentation von Programmen	138
6.2.2	Die Interpreterfunktion <code>eval</code>	140
6.3	Ein Compiler für Mini-ML	147
6.3.1	Die SECD-Maschine	148
6.3.2	Übersetzung von Mini-ML in SECD-Code	153
6.3.3	Behandlung von Rekursion	155
6.3.4	Implementierung der SECD-Maschine in ML	159
6.4	Literaturhinweise	163
	Lösungen zu den Aufgaben	165
	Literatur	179
	Index	185