

Inhaltsverzeichnis

Vorwort	9	
Software	9	
Danksagung	10	
Kapitel 1	Einführung	11
1.1	Programmiersprachen	11
1.2	Werte, Funktionen und Typen	13
Kapitel 2	Programmierumgebung	21
2.1	Die Betriebssystemfamilie Unix	21
2.2	Haskell-Programme ausführen	23
Kapitel 3	Grundlegende Kontrollstrukturen und Typen	25
3.1	Programme werden aus Modulen gebildet	25
3.2	Verzweigungen im Programmablauf	26
3.3	Zuweisungen	28
3.4	Tupel: Kombinationen verschiedener Datenelemente	29
3.5	Listen: Viele Werte desselben Typs	32
3.6	Layout	34
Kapitel 4	Rekursion	37
4.1	Rekursion über Zahlen	37
4.2	Partielle Funktionen	41
4.3	Rekursion und Listen	42
4.4	Polymorphie und Überladen von Operationen	47
Kapitel 5	Listen und Funktionen höherer Ordnung	51
5.1	Daten aus einer Liste extrahieren	51
5.2	Listenelemente aktualisieren	57
5.3	Ein kurzer Blick auf das Sortieren	58
5.4	Funktionen höherer Ordnung	58
Kapitel 6	Die Unix-Programmierumgebung	65
6.1	Unix	65
6.2	Unix – kleine Tools und wie man sie kombiniert	69
6.3	Umleitung und Pipeline	72
6.4	Interpreter und Compiler	73
6.5	Shell-Skripte	75
6.6	Ein Autotest-Skript	77
6.7	Skripte für den automatischen Test	84

Kapitel 7	Eingabe und Ausgabe	87
	7.1 Noch einmal: Compiler versus Interpreter	87
	7.2 Eingabe-/Ausgabe-Aktionen	89
	7.3 Datei-Eingabe/-Ausgabe	97
	7.4 Beispiele	97
Kapitel 8	Benutzerdefinierte Datentypen	103
	8.1 Parametrische Typsynonyme	103
	8.2 Aufzählungstypen	104
	8.3 Parametrisierte Alternativen	106
	8.4 Rekursive Datentypdefinitionen	109
	8.5 Intermezzo: Die Bedeutung der verschiedenen Syntaxvarianten	111
	8.6 Typklassen für benutzerdefinierte Datentypen	113
Kapitel 9	Formale Überlegungen	117
	9.1 Programmkalkulation	118
	9.2 Mit Lemmata und Generalisierung arbeiten	127
	9.3 Programmherleitung	131
Kapitel 10	Modularisierung und Programmdekomposition	137
	10.1 Modulares Design	138
	10.2 Geltungsbereich und Sichtbarkeit von Namen	143
Kapitel 11	Software-Komplexität	149
	11.1 Analyse der Komplexität	150
	11.2 Die O -Notation	156
	11.3 Effizientere Sortieralgorithmen	158
	11.4 Rekurrenzrelationen auflösen	164
Kapitel 12	Baumstrukturen	165
	12.1 Datenstrukturen und Effizienz	165
	12.2 AVL-Bäume	170
	12.3 Trie-Strukturen	175
Kapitel 13	Arithmetische Ausdrücke auswerten	179
	13.1 Geschlossene Ausdrücke auswerten	179
	13.2 Ausdrücke mit Variablen	182
	13.3 Funktionsdefinitionen einfügen	185

Anhang A	Gemeinsame Eigenschaften	189
	A.1 Einige grundlegende Eigenschaften	189
Anhang B	Zusätzlicher Programmcode	191
	B.1 Der Quellcode für den Ausdrucksparser	191
	Sachregister	195