

Inhaltsverzeichnis

1. Imperative Programmierung	1
1.1 Maschine und Sprache	2
1.2 Ziele und Wege	12
Literatur zu Kapitel 1	16
2. Qualität und Stil "im Kleinen"	17
2.1 Programm-Qualität	18
2.1.1 Korrektheit	19
2.1.2 Integrierbarkeit	24
2.1.3 Wartbarkeit	25
2.1.4 Effizienz	27
2.2 Programmierstil	30
2.2.1 Dokumentation und Selbstdokumentation	30
2.2.2 Aspekte der strukturellen Komplexität	42
2.3 Einfache Maßnahmen zur Verbesserung der Effizienz	50
2.3.1 Allgemeine Überlegungen	51
2.3.2 Spezielle Regeln	52
Literatur zu Kapitel 2	61
3. Schrittweise Verfeinerung	63
3.1 Schrittweise Verfeinerung als Entwurfstechnik	64
3.2 Zwei Beispiele	66
3.2.1 Folgedatum	67
3.2.2 Acht Damen	80
3.3 Diagramm-Techniken zur Unterstützung "Schrittweiser Verfeinerung"	96
3.3.1 Nassi-Shneiderman-Diagramme	99
3.3.2 Klammerdiagramme	104
Literatur zu Kapitel 3	110
4. Programmieren durch Beweisen	111
4.1 Prädikate	112
4.1.1 Speicher und Variable	112
4.1.2 Operationen mit Prädikaten	114
4.1.3 Prädikate als Kommentare	120
4.2 Semantik	124
4.2.1 Schwächste Vorbedingungen	124
4.2.2 Zuweisungen und Sequenzen	129
4.2.3 Selektionen	131
4.2.4 Iterationen	135

4.3 Zielorientierte Programmentwicklung	144
4.3.1 Konstruktion von Selektions-Anweisungen	145
4.3.2 Konstruktion von Iterations-Anweisungen	148
4.3.3 Zur Auffindung von Schleifen-Invarianten	157
Literatur zu Kapitel 4	164
5. Datenstrukturierter Programm-Entwurf	165
5.1 Objektstruktur und Programmstruktur	169
5.1.1 Strukturbeschreibungen	169
5.1.2 Produktion von Datenobjekten	179
5.1.3 Analyse von Datenobjekten	185
5.2 Programm-Konstruktion aus Input und Output	199
5.2.1 Das Verfahren in seiner einfachsten Form	200
5.2.2 Verarbeitung mehrerer Input-Ströme	209
5.2.3 Strukturkonflikte und ihre Auflösung	222
Literatur zu Kapitel 5	227
6. Aspekte der Programmierung im Großen	228
6.1 Softwaresysteme	229
6.2 Qualität "im Großen" und Software-Engineering	234
6.2.1 Korrektheit und die Folgen	234
6.2.2 Integrierbarkeit und die Folgen	241
6.2.3 Wartbarkeit und die Folgen	245
6.2.4 Benutzungsfreundlichkeit	253
6.2.5 Software-Engineering	258
6.3 Systemanalyse und Systemspezifikation	262
6.3.1 Realwelt-Modelle für die Software-Spezifikation	264
6.3.2 Petrinetze	272
6.3.3 "Jackson System Development"	284
6.3.4 Hierarchische Systemmodelle	305
6.4 Systementwurf und Systemimplementierung	317
6.4.1 Ziele und Techniken	321
6.4.2 Modularisierung	330
6.4.3 Abstrakte Datentypen	357
6.4.4 Objekte, Klassen und Vererbung	379
6.4.5 Parallele Prozesse	424
Literatur zu Kapitel 6	447
Stichwortverzeichnis	451