

Inhaltsverzeichnis

Einleitung	1
Was dieses Buch unter „Programmieren“ versteht	1
Worauf ein angehender Programmierer achten muss	3
Wie dieses Buch zu benutzen ist	4
Bei Fragen und Kommentaren	6
1. Grundlagen der Programmierung	9
1.1 Der Begriff des Algorithmus	9
1.1.1 Addition als einführendes Beispiel	9
1.1.2 Grundschrirte von Algorithmen	13
1.1.3 Veranschaulichung von Algorithmen durch Flussdiagramme	14
1.2 Elementarer Rechneraufbau	18
1.2.1 Der Prozessor	18
1.2.2 Der Hauptspeicher	18
1.2.3 Die CPU und ihre Maschinensprache	21
1.2.4 Eingabe und Ausgabe	25
1.3 Daten und Operationen auf Daten	27
1.3.1 Nicht negative ganze Zahlen	28
1.3.2 Negative ganze Zahlen	32
1.3.3 Zahlen mit Nachkommastellen	33
1.3.4 Zeichen und Texte	36
1.3.5 Wahrheitswerte	38
1.3.6 Die Universalität der binären Kodierung	40
1.4 Assembler	40
1.5 Höhere Programmiersprachen	42
1.5.1 Übersetzung von Hochsprachen	44
1.5.2 Syntax und Semantik	45
1.5.3 Einteilung der Programmiersprachen	46
1.5.4 Sprachelemente imperativ-prozeduraler Sprachen	48
1.6 Grundlegende Programme	52
1.6.1 Das Betriebssystem	52
1.6.2 Der Kommandozeilen-Interpreter	57
1.6.3 Editoren	57
1.6.4 Compiler, Assembler, Linker, Lader	59

1.6.5	Interpreter	59
	Literaturhinweise	60
2.	Einfache Programme	61
2.1	Grundlegende Elemente von Programmen	63
2.1.1	„Hallo Welt!“, Das allererste Programm	63
2.1.2	Variable und konstante Objekte	67
2.1.3	Konstanten und Literale	68
2.1.4	Skalare Variablen und Zuweisungen	70
2.1.5	Einfache Ausdrücke	74
2.1.6	Eingabe von skalaren Variablen	80
2.1.7	Benutzung von eingebauten Funktionen	81
2.2	Bedingte Anweisungen	84
2.2.1	Die einfache bedingte Anweisung	85
2.2.2	Bedingungen	88
2.2.3	Die bedingte Anweisung mit sonst-Teil	89
2.2.4	Die bedingte Anweisung mit Mehrfachverzweigung	91
2.2.5	Schachtelung von Blöcken	93
2.2.6	Ausdrücke und ihre Auswertung	95
2.2.7	Wahrheitswerte	97
2.2.8	Arithmetische und lexikographische Vergleiche	99
2.2.9	Logische Operatoren	101
2.2.10	Übersicht über die Operatoren von Perl	104
2.2.11	Testen von Variablen auf Definiertheit	104
2.3	Schleifen	106
2.3.1	Die kopfgesteuerte Schleife (while-Schleife)	106
2.3.2	Die Zählschleife (for-Schleife)	109
2.3.3	Bedingte Anweisungen in Schleifen	111
2.3.4	Schleifen in Schleifen	115
2.3.5	Durchforsten der Standardeingabe	118
2.4	Arrays	120
2.4.1	Arrays im Gegensatz zu Listen	120
2.4.2	Initialisierung von Arrays und Zugriff auf einzelne Elemente	122
2.4.3	Füllen und Durchwandern von Arrays	127
2.4.4	Suchen in Arrays	127
2.4.5	Ein einfacher Sortieralgorithmus	133
2.4.6	Turing-Maschinen und Berechenbarkeit	137
	Literaturhinweise	139
3.	Fortgeschrittenes Programmieren	141
3.1	Hashes	141
3.2	Weitere Kontrollstrukturen	146
3.2.1	Die fußgesteuerte Schleife (do-while-Schleife)	147
3.2.2	Die foreach-Schleife	148

3.2.3	Schleifensprünge und der (nicht zu empfehlende) Gebrauch von Labels	149
3.3	Unterprogramme	154
3.3.1	Eine einfache Funktion	155
3.3.2	Lokale Variablen	158
3.3.3	Argumentenübergabe an Funktionen	160
3.3.4	Eine Prozedur	162
3.3.5	Verschachtelte Funktionsaufrufe	163
3.4	Systematische Fehlersuche	167
3.4.1	Syntaktische Fehler	167
3.4.2	Semantische Fehler	169
3.5	Ein- und Ausgabe	174
3.5.1	Kommandozeilen-Argumente	174
3.5.2	Öffnen von Dateien zum zeilenweisen Lesen und Schreiben	176
3.5.3	Lesen von Verzeichnissen	183
3.5.4	Dateitest-Operatoren	185
3.5.5	Datei-Operationen	187
3.5.6	Formatierte Ausgabe mit <code>printf</code>	189
3.5.7	Binärdateien	192
3.6	Rekursive Funktionen	197
3.6.1	Rekursive Größen in der Mathematik	197
3.6.2	Rekursive Funktionen in der Informatik	199
3.6.3	Sortieren durch Mischen (Mergesort)	201
	Literaturhinweise	205
4.	Elemente anspruchsvoller Programme	207
4.1	Referenzen und Zeiger	207
4.1.1	Referenzwerte	208
4.1.2	Referenzieren und Dereferenzieren in Perl	210
4.1.3	Klassische Anwendungsbeispiele für Referenzen	216
4.2	Manipulation einzelner Bits	224
4.2.1	Bitweise Operatoren	224
4.2.2	Bitweise Schiebe-Operatoren	228
4.3	Reguläre Ausdrücke	232
4.3.1	Formale Sprachen	233
4.3.2	Reguläre Ausdrücke und reguläre Sprachen	235
4.3.3	Endliche Automaten	237
4.3.4	Reguläre Ausdrücke in Perl	239
4.3.5	Mustererkennung und Musterersetzung	244
4.4	Zufallszahlen	247
4.5	Elementare Datenstrukturen	249
4.5.1	Arrays und Hashes im Vergleich	250
4.5.2	Verbundtypen	251
4.5.3	Zwei- und mehrdimensionale Arrays	253
4.5.4	Verkettete Listen	257

4.5.5	Stapel	260
4.5.6	Schlangen	263
4.5.7	Bäume	264
4.6	Benchmarking und Profiling	268
4.7	Bibliotheken	270
4.7.1	Wiederverwendung von Code	270
4.7.2	Selbst geschriebene Module in Perl	272
4.7.3	Vordefinierte Module in Perl	278
	Literaturhinweise	280
5.	Lösungen zu ausgewählten Übungen	283
5.1	Lösungen zu Übungen aus Kapitel 1	284
5.2	Lösungen zu Übungen aus Kapitel 2	289
5.3	Lösungen zu Übungen aus Kapitel 3	331
5.4	Lösungen zu Übungen aus Kapitel 4	355
A.	Installation von Perl	369
A.1	Bezugsquellen	369
A.2	Installation	370
A.2.1	Vorkompilierte Binaries	370
A.2.2	Perl selbst kompilieren	371
A.3	Eine UNIX-Umgebung für MS-Windows	372
B.	Dokumentation zu Perl	373
B.1	Online-Dokumentation	373
B.1.1	Die Manualseiten	373
B.1.2	Informationen aus dem Internet	374
B.2	Dokumentation in gedruckter Form	375
C.	Installation der Kursunterlagen	377
C.1	Bezugsquellen	377
C.2	Installation	377
	Abbildungsverzeichnis	379
	Tabellenverzeichnis	381
	Literaturverzeichnis	383
	Index	385