

Inhaltsverzeichnis

Vorwort	xv
Wie dieses Buch entstanden ist	xv
Danksagungen	xvi
Ein Aufruf zur Unterstützung freier Software	xvii
Einleitung	xix
Was dieses Buch unter „Programmieren“ versteht	xix
Worauf ein angehender Programmierer achten muss	xxi
Wie dieses Buch zu benutzen ist	xxii
Bei Fragen und Kommentaren	xxiv
1. Grundlagen der Programmierung	1
1.1 Der Begriff des Algorithmus	1
1.1.1 Addition als einführendes Beispiel	1
1.1.2 Grundschrirte von Algorithmen	5
1.1.3 Veranschaulichung von Algorithmen durch Flussdiagramme	6
1.2 Elementarer Rechneraufbau	10
1.2.1 Der Prozessor	10
1.2.2 Der Hauptspeicher	10
1.2.3 Die CPU und ihre Maschinensprache	13
1.2.4 Eingabe und Ausgabe	17
1.3 Daten und Operationen auf Daten	19
1.3.1 Nicht-negative ganze Zahlen	20
1.3.2 Negative ganze Zahlen	24
1.3.3 Zahlen mit Nachkommastellen	25
1.3.4 Zeichen und Texte	28
1.3.5 Wahrheitswerte	30
1.3.6 Die Universalität der binären Kodierung	32
1.4 Assembler	33
1.5 Höhere Programmiersprachen	35
1.5.1 Übersetzung von Hochsprachen	36
1.5.2 Syntax und Semantik	37
1.5.3 Einteilung der Programmiersprachen	38
1.5.4 Sprachelemente imperativ-prozeduraler Sprachen	40

1.6	Grundlegende Programme	44
1.6.1	Das Betriebssystem	44
1.6.2	Der Kommandozeilen-Interpreter	49
1.6.3	Editoren	50
1.6.4	Compiler, Assembler, Linker, Lader	51
1.6.5	Interpreter	52
	Literaturhinweise	52
2.	Einfache Programme	55
2.1	Grundlegende Elemente von Programmen	57
2.1.1	„Hallo Welt!“: Das allererste Programm	57
2.1.2	Variable und konstante Objekte	61
2.1.3	Konstanten und Literale	62
2.1.4	Skalare Variablen und Zuweisungen	64
2.1.5	Einfache Ausdrücke	68
2.1.6	Eingabe von skalaren Variablen	73
2.1.7	Benutzung von eingebauten Funktionen	75
2.2	Bedingte Anweisungen	78
2.2.1	Die einfache bedingte Anweisung	79
2.2.2	Bedingungen	82
2.2.3	Die bedingte Anweisung mit sonst-Teil	83
2.2.4	Die allgemeine bedingte Anweisung	85
2.2.5	Schachtelung von Blöcken	87
2.2.6	Ausdrücke und ihre Auswertung	89
2.2.7	Wahrheitswerte	91
2.2.8	Arithmetische und lexikographische Vergleiche	93
2.2.9	Logische Operatoren	95
2.2.10	Übersicht über die Operatoren von Perl	98
2.2.11	Testen von Variablen auf Definiertheit	98
2.3	Schleifen	100
2.3.1	Die kopfgesteuerte Schleife (while-Schleife)	100
2.3.2	Die Zählschleife (for-Schleife)	103
2.3.3	Bedingte Anweisungen in Schleifen	105
2.3.4	Schleifen in Schleifen	110
2.3.5	Durchforsten der Standardeingabe	113
2.4	Arrays	114
2.4.1	Arrays im Gegensatz zu Listen	115
2.4.2	Initialisierung von Arrays und Zugriff auf einzelne Elemente	117
2.4.3	Füllen und Durchwandern von Arrays	121
2.4.4	Suchen in Arrays	122
2.4.5	Ein einfacher Sortieralgorithmus	127
2.4.6	Turing-Maschinen und Berechenbarkeit	132
	Literaturhinweise	134

3. Fortgeschrittenes Programmieren	135
3.1 Hashes	135
3.2 Weitere Kontrollstrukturen	140
3.2.1 Die fußgesteuerte Schleife (do-while-Schleife)	141
3.2.2 Die foreach-Schleife	142
3.2.3 Schleifensprünge und der (nicht zu empfehlende) Gebrauch von Labels	143
3.3 Unterprogramme	147
3.3.1 Eine einfache Funktion	148
3.3.2 Lokale Variablen	151
3.3.3 Argumentenübergabe an Funktionen	153
3.3.4 Eine Prozedur	156
3.3.5 Verschachtelte Funktionsaufrufe	156
3.4 Systematische Fehlersuche	160
3.4.1 Syntaktische Fehler	160
3.4.2 Semantische Fehler	162
3.5 Ein- und Ausgabe	167
3.5.1 Kommandozeilen-Argumente	167
3.5.2 Öffnen von Dateien zum zeilenweisen Lesen und Schreiben	169
3.5.3 Lesen von Verzeichnissen	177
3.5.4 Datei-Test-Operatoren	178
3.5.5 Datei-Operationen	180
3.5.6 Formatierte Ausgabe mit <code>printf</code>	182
3.5.7 Binärdateien	185
3.6 Rekursive Funktionen	191
3.6.1 Rekursive Größen in der Mathematik	191
3.6.2 Rekursive Funktionen in der Informatik	192
3.6.3 Sortieren durch Mischen (Mergesort)	194
Literaturhinweise	199
4. Elemente anspruchsvoller Programme	201
4.1 Referenzen und Zeiger	201
4.1.1 Referenzwerte	202
4.1.2 Referenzieren und Dereferenzieren in Perl	204
4.1.3 Klassische Anwendungsbeispiele für Referenzen	210
4.2 Manipulation einzelner Bits	218
4.2.1 Bitweise Operatoren	219
4.2.2 Arithmetische Schiebe-Operatoren	223
4.3 Reguläre Ausdrücke	226
4.3.1 Formale Sprachen	227
4.3.2 Reguläre Ausdrücke und reguläre Sprachen	229
4.3.3 Endliche Automaten	231
4.3.4 Reguläre Ausdrücke in Perl	233
4.3.5 Mustererkennung und Musterersetzung	238
4.4 Zufallszahlen	241

4.5	Elementare Datenstrukturen	243
4.5.1	Arrays und Hashes im Vergleich	244
4.5.2	Verbundtypen	245
4.5.3	Zwei- und mehrdimensionale Arrays	247
4.5.4	Verkettete Listen	251
4.5.5	Stapel	254
4.5.6	Schlangen	257
4.5.7	Bäume	258
4.6	Benchmarking und Profiling	262
4.7	Bibliotheken	264
4.7.1	Wiederverwendung von Code	264
4.7.2	Selbstgeschriebene Module in Perl	266
4.7.3	Vordefinierte Module in Perl	271
	Literaturhinweise	273
5.	Lösungen zu ausgewählten Übungen	275
5.1	Lösungen zu Übungen aus Kapitel 1	275
5.2	Lösungen zu Übungen aus Kapitel 2	280
5.3	Lösungen zu Übungen aus Kapitel 3	322
5.4	Lösungen zu Übungen aus Kapitel 4	346
A.	Installation von Perl	361
A.1	Bezugsquellen	361
A.2	Installation	362
A.2.1	Vorkompilierte Binaries	362
A.2.2	Perl selbst kompilieren	363
A.3	Eine UNIX-Umgebung für MS-Windows	364
B.	Dokumentation zu Perl	365
B.1	Online-Dokumentation	365
B.1.1	Die Manual-Seiten	365
B.1.2	Informationen aus dem Internet	366
B.2	Dokumentation in gedruckter Form	367
C.	Installation der Kursunterlagen	369
C.1	Bezugsquellen	369
C.2	Installation	369

Abbildungsverzeichnis

1.1	Schriftliche Addition	1
1.2	Algorithmus zur Addition zweier gleichlanger Zahlen	3
1.3	Programmablaufpläne nach DIN 66001	7
1.4	Programmablaufplan zur Bestimmung des Maximums zweier Zahlen	8
1.5	Schematischer Aufbau eines Prozessors	11
1.6	Ein Speicherbaustein	12
1.7	Eine Speicherzelle aus 8 Bits	13
1.8	Peripherie-Geräte	17
1.9	Fließkommadarstellung nach IEEE 754, einfache Präzision	27
1.10	Hardware-Software-Hierarchie	46
1.11	Ein Ausschnitt aus einem UNIX-Dateibaum	48
1.12	Der Editor Emacs	51
2.1	Drei skalare Variablen	67
2.2	Zwei Iterationen von binärer Suche	108
2.3	Zwei geschachtelte for-Schleifen	111
2.4	Ein Array mit 6 Einträgen	116
2.5	Eine (verkettete) Liste mit 6 Elementen	116
2.6	Elemente und Indizes eines Perl-Arrays	120
2.7	Ein Array nach einem zyklischen Links-Shift	125
2.8	Erzeugen von Paarungen	126
2.9	Das Josephus-Problem	127
2.10	Sortieren durch Minimumsuche benötigt $1/2n^2$ Vergleiche	129
2.11	Identität und Logarithmusfunktion	130
2.12	Zeitkomplexität verschiedener Sortieralgorithmen	131
2.13	Eine Turing-Maschine	133
3.1	Ein Hash mit 5 Schlüssel-Werte-Paaren	136
3.2	Eine Hash-Tafel	138
3.3	Der Aufrufbaum der Funktion \max_3	157
3.4	Eine Textdatei und ein zugehöriges Filehandle	170
3.5	Top-Down- und Bottom-Up-Programmierung	176
3.6	Rekursive Berechnung von $4!$	193
3.7	Mischen zweier Teilarrays	195
3.8	Teilen und Mischen bei Mergesort	195

3.9	Die Türme von Hanoi	198
4.1	Eine Landkarte mit Zeigern	202
4.2	Eine Variable x und eine darauf verweisende Zeigervariable y	203
4.3	Referenzen auf Hashes	206
4.4	Referenzen auf Skalare, Arrays und Hashes	209
4.5	Ein Schachbrett als zweidimensionales Array	217
4.6	Bitweises Oder von 130 mit 7	219
4.7	Bitweises Und von 130 mit 7	220
4.8	Bitweises exklusives Oder von 130 mit 7	221
4.9	Bitweises Komplementieren von 7	222
4.10	Arithmetisches Schieben um eine Stelle nach rechts bzw. links	223
4.11	Ein endlicher Automat	232
4.12	Ein unendlicher Automat	233
4.13	Ein eindimensionales und ein zweidimensionales Array	248
4.14	Conways Spiel des Lebens	250
4.15	Einfügen in eine einfach verkettete Liste	251
4.16	Eine doppelt verkettete, zirkuläre Liste	252
4.17	Ein Stapel zur Auswertung von Postfix-Ausdrücken	254
4.18	Eine Schlange	257
4.19	Ein Baum	259
4.20	Ein vollständiger binärer Baum	261
5.1	Das exklusive Oder	280