
Inhaltsverzeichnis

| | |
|---|-----------|
| Eine Einleitung in Frage und Antwort | V |
| 1 Vorbereitungen | 1 |
| 1.1 Modellierung und Algorithmen | 1 |
| 1.2 Komplexität und \mathcal{O} -Notation | 11 |
| 1.3 Kondition eines Problems | 16 |
| 1.4 Rechnerarithmetik | 18 |
| 1.4.1 Zahldarstellung | 18 |
| 1.4.2 Rundung und Gleitpunktrechnung | 20 |
| 1.4.3 Binäre Realisierung | 23 |
| 1.5 Stabilität | 26 |
| 1.6 Vom Problem zum Programm – und zurück | 28 |
| 1.7 Kontrollfragen zu Kapitel 1 | 30 |
| 1.8 Übungsaufgaben zu Kapitel 1 | 32 |
| 2 Elementare C-Programmierung | 35 |
| 2.1 Editieren und Übersetzen am Beispiel: „Hallo Welt!“ | 36 |
| 2.2 Datentypen | 38 |
| 2.3 Variablen und ihre Deklaration | 40 |
| 2.4 Konstanten | 42 |
| 2.5 Operatoren | 43 |
| 2.5.1 Allgemeines zu Operatoren | 43 |
| 2.5.2 Zuweisungsoperator und schreibgeschützte Variablen ... | 43 |
| 2.5.3 Arithmetische Operatoren | 45 |
| 2.5.4 Arithmetische Zuweisungsoperatoren | 47 |
| 2.5.5 Inkrement- und Dekrementoperatoren | 47 |
| 2.5.6 Vergleichende und logische Operatoren | 48 |
| 2.5.7 Typumwandlung durch <i>Casts</i> | 51 |
| 2.6 Einfache Ein- und Ausgabe | 51 |
| 2.6.1 Ausgabe mit <code>printf()</code> | 52 |
| 2.6.2 Eingabe mittels <code>scanf()</code> | 54 |

| | | |
|----------|--|------------|
| 2.7 | Programmflusskontrolle | 56 |
| 2.7.1 | Anweisungsblöcke | 56 |
| 2.7.2 | Bedingte Ausführung | 57 |
| 2.7.3 | Wiederholte Ausführung | 60 |
| 2.8 | Felder | 65 |
| 2.9 | Beispiel: Der Euklidische Algorithmus | 67 |
| 2.10 | Kontrollfragen zu Kapitel 2 | 69 |
| 2.11 | Übungsaufgaben zu Kapitel 2 | 75 |
| 3 | Funktionen | 77 |
| 3.1 | Deklaration und Definition von Funktionen | 78 |
| 3.2 | Call by Value | 83 |
| 3.3 | Mathematische Funktionen | 85 |
| 3.4 | Gültigkeit von Variablen | 88 |
| 3.5 | Beispiel: Entfernungsmessung durch Peilung | 92 |
| 3.6 | Kontrollfragen zu Kapitel 3 | 95 |
| 3.7 | Übungsaufgaben zu Kapitel 3 | 99 |
| 4 | Zeiger und ihre Anwendungen | 103 |
| 4.1 | Zeiger | 104 |
| 4.1.1 | Elementare Operationen mit Zeigern | 104 |
| 4.1.2 | Der Datentyp <code>void*</code> | 106 |
| 4.2 | Call by Reference | 107 |
| 4.3 | Zeiger und Felder: Zeigerarithmetik | 109 |
| 4.4 | Dynamische Speicherverwaltung | 113 |
| 4.5 | Dynamische Implementierung von Matrizen | 118 |
| 4.5.1 | Implementierung über Doppelzeiger | 118 |
| 4.5.2 | Implementierung durch Indextransformation | 120 |
| 4.6 | Typische Fehlerquellen | 122 |
| 4.7 | Kontrollfragen zu Kapitel 4 | 125 |
| 4.8 | Übungsaufgaben zu Kapitel 4 | 129 |
| 5 | Numerisches Zwischenspiel | 131 |
| 5.1 | Nullstellenbestimmung | 132 |
| 5.2 | Interpolation | 137 |
| 5.3 | Numerische Integration | 142 |
| 5.4 | Kontrollfragen zu Kapitel 5 | 145 |
| 5.5 | Übungsaufgaben zu Kapitel 5 | 147 |
| 6 | Zeichen und Strings | 149 |
| 6.1 | Zeichen | 149 |
| 6.2 | Strings | 153 |
| 6.2.1 | Initialisierung und Terminierung | 153 |
| 6.2.2 | Bibliotheksfunktionen für Strings | 154 |
| 6.3 | Beispiel: Einfache Kryptographie | 158 |

| | | |
|-----------|--|------------|
| 6.4 | Kontrollfragen zu Kapitel 6 | 161 |
| 6.5 | Übungsaufgaben zu Kapitel 6 | 162 |
| 7 | Fortgeschrittene Ein- und Ausgabe | 163 |
| 7.1 | Arbeiten mit Dateien | 163 |
| 7.1.1 | ASCII-Format | 165 |
| 7.1.2 | Binäre Ein- und Ausgabe | 168 |
| 7.2 | Kommandozeilenargumente | 171 |
| 7.3 | Beispiel: Unbeschränktes Bakterienwachstum | 174 |
| 7.4 | Kontrollfragen zu Kapitel 7 | 177 |
| 7.5 | Übungsaufgaben zu Kapitel 7 | 178 |
| 8 | Fortgeschrittene Datentypen | 181 |
| 8.1 | Strukturen | 181 |
| 8.2 | Anwendungsbeispiele für Strukturen | 187 |
| 8.2.1 | Zeitmessung | 187 |
| 8.2.2 | Einfach verkettete Listen | 188 |
| 8.3 | Benennung eigener Datentypen mit <code>typedef</code> | 192 |
| 8.4 | Zeiger auf Funktionen | 193 |
| 8.5 | Beispiele für zusammengesetzte Deklarationen | 195 |
| 8.6 | Weitere Datentypen: <code>enum</code> und <code>union</code> | 197 |
| 8.7 | Kontrollfragen zu Kapitel 8 | 198 |
| 8.8 | Übungsaufgaben zu Kapitel 8 | 200 |
| 9 | Rekursion | 203 |
| 9.1 | Rekursive Programmierung | 203 |
| 9.2 | Effiziente Such- und Sortieralgorithmen | 206 |
| 9.3 | Kontrollfragen zu Kapitel 9 | 215 |
| 9.4 | Übungsaufgaben zu Kapitel 9 | 217 |
| 10 | Speicher- und laufzeiteffiziente Datenstrukturen | 219 |
| 10.1 | Symmetrische Matrizen | 219 |
| 10.2 | Das dyadische Produkt | 221 |
| 10.3 | Dünn besetzte Matrizen | 223 |
| 10.3.1 | Bandmatrizen | 223 |
| 10.3.2 | Unstrukturierte dünn besetzte Matrizen | 225 |
| 10.4 | Permutationen und Indexfelder | 228 |
| 10.5 | Kontrollfragen zu Kapitel 10 | 230 |
| 10.6 | Übungsaufgaben zu Kapitel 10 | 231 |
| 11 | Mehrdateiprojekte, Bibliotheken und Makefiles | 233 |
| 11.1 | Die Übersetzung mehrerer Quelldateien zu einem Programm .. | 234 |
| 11.2 | Organisation des Quelltextes | 236 |
| 11.3 | Eigene Bibliotheken | 238 |
| 11.4 | Automatisierte Übersetzung mit <code>make</code> | 240 |
| 11.5 | Einbindung von FORTRAN-Programmen | 243 |

| | | |
|-----------|---|------------|
| 11.6 | Kontrollfragen zu Kapitel 11 | 245 |
| 11.7 | Übungsaufgaben zu Kapitel 11 | 248 |
| 12 | Pseudozufallszahlen | 249 |
| 12.1 | Ein wenig „Mathematik des Zufalls“ | 250 |
| 12.2 | Pseudozufallszahlen | 256 |
| 12.3 | Erzeugung von Zufallszahlen gemäß einer Verteilung | 259 |
| 12.4 | Einfache Monte-Carlo-Methoden | 261 |
| 12.5 | Übungsaufgaben zu Kapitel 12 | 265 |
| 13 | Programmierprojekte | 269 |
| 13.1 | Projekt 1: Simulation von Warteschlangen | 269 |
| 13.1.1 | Der grundlegende Algorithmus | 269 |
| 13.1.2 | Der Zufallszahlengenerator | 272 |
| 13.1.3 | Das Hauptprogramm | 273 |
| 13.1.4 | Zwei Beispielrechnungen | 274 |
| 13.2 | Projekt 2: Planetenbahnen | 276 |
| 13.2.1 | Das mathematische Modell: Newtons Gravitationsgesetz | 276 |
| 13.2.2 | Grundlegende Datenstrukturen | 277 |
| 13.2.3 | Implementierung des Euler-Verfahrens | 280 |
| 13.2.4 | Erste Simulation | 282 |
| 13.2.5 | Zweite Simulation | 284 |
| 13.2.6 | Die Planetenbahnen als Animation | 286 |
| 13.3 | Übungsaufgaben zu Kapitel 13 | 288 |
| | Installation von cygwin | 289 |
| | Die Kommandozeile von LINUX | 291 |
| | Kurze Einführung in gnuplot | 297 |
| | Reservierte Wörter und Operatoren in C | 301 |
| | Lösungen zu den Kontrollfragen | 305 |
| | Literaturverzeichnis | 307 |
| | Sachverzeichnis | 309 |



<http://www.springer.com/978-3-540-45383-3>

Programmieren in C

Eine mathematikorientierte Einführung

Kirsch, R.; Schmitt, U.

2007, XII, 312 S., Softcover

ISBN: 978-3-540-45383-3