

Auf einen Blick

| | |
|--|------------|
| Einführung | 23 |
| TEIL I: Mit der Programmierung in C beginnen | 29 |
| Kapitel 1: Ein schneller Start für die Ungeduldigen | 31 |
| Kapitel 2: Die Sache mit dem Programmieren..... | 43 |
| Kapitel 3: Die Anatomie von C | 51 |
| TEIL II: Das kleine Einmaleins der C-Programmierung | 65 |
| Kapitel 4: Versuch und Irrtum | 67 |
| Kapitel 5: Werte und Konstanten | 79 |
| Kapitel 6: Ein Abstellplatz | 89 |
| Kapitel 7: Eingabe und Ausgabe | 99 |
| Kapitel 8: Entscheidungsfindung..... | 111 |
| Kapitel 9: Schleifen, Schleifen, Schleifen | 125 |
| Kapitel 10: Mit Funktionen Spaß haben..... | 141 |
| TEIL III: Auf dem vorhandenen Wissen aufbauen | 155 |
| Kapitel 11: Das unvermeidliche Mathematikkapitel | 157 |
| Kapitel 12: Her mit den Arrays | 173 |
| Kapitel 13: Mit Text Spaß haben | 191 |
| Kapitel 14: Die multivariablen Strukturen | 207 |
| Kapitel 15: Das Leben in der Kommandozeile..... | 217 |
| Kapitel 16: Variabler Blödsinn | 225 |
| Kapitel 17: Binärer Wahnsinn..... | 237 |
| TEIL IV: Für Fortgeschrittene | 255 |
| Kapitel 18: Einführung in Pointer..... | 257 |
| Kapitel 19: Tief im Pointer-Land..... | 271 |
| Kapitel 20: Verkettete Listen..... | 289 |
| Kapitel 21: Es wird Zeit..... | 309 |
| TEIL V: Und der ganze Rest | 317 |
| Kapitel 22: Dauerhafte Speicherfunktionen | 319 |
| Kapitel 23: Verwaltung von Dateien | 335 |
| Kapitel 24: Projekte für Fortgeschrittene und Hartgesottene | 345 |
| Kapitel 25: Programmfehler raus!..... | 355 |
| TEIL VI: Der Top-Ten-Teil | 367 |
| Kapitel 26: Zehn gängige Schnitzer..... | 369 |
| Anhänge | 377 |
| Stichwortverzeichnis | 390 |

Inhaltsverzeichnis

| | |
|--|-----------|
| Einführung | 23 |
| Ist die Sprache C wichtig? | 23 |
| Der Ansatz dieses Buchs. | 24 |
| Wie dieses Buch vorgeht | 24 |
| Symbole in diesem Buch | 26 |
| Gedanken zum Schluss. | 26 |
| TEIL I: | |
| MIT DER PROGRAMMIERUNG IN C BEGINNEN | 29 |
| Kapitel 1 | |
| Ein schneller Start für die Ungeduldigen | 31 |
| Was Sie zum Programmieren benötigen | 31 |
| Tools zum Programmieren besorgen | 31 |
| Eine Entwicklungsumgebung (IDE) besorgen. | 32 |
| Die IDE Code::Blocks betrachten. | 32 |
| Code::Blocks installieren | 33 |
| Eine Reise durch den Arbeitsbereich von Code::Blocks | 34 |
| Ihr erstes Projekt | 36 |
| Ein neues Projekt erstellen | 36 |
| Den Quelltext überprüfen. | 38 |
| Das Projekt erstellen und laufen lassen | 40 |
| Speichern und schließen. | 41 |
| Kapitel 2 | |
| Die Sache mit dem Programmieren | 43 |
| Die Geschichte des Programmierens | 43 |
| Ein Überblick über die frühe Geschichte des Programmierens | 44 |
| Die Einführung der Programmiersprache C. | 44 |
| Der Prozess des Programmierens | 45 |
| Das Programmieren verstehen | 45 |
| Quellcode schreiben | 46 |
| In Objektcode verwandeln | 47 |
| Mit der Bibliothek von C verknüpfen. | 48 |
| Durchlaufen und testen | 49 |
| Kapitel 3 | |
| Die Anatomie von C | 51 |
| Teile der Programmiersprache C | 51 |
| Schlüsselwörter | 52 |
| Funktionen | 53 |

14 Inhaltsverzeichnis

| | |
|--|-----------|
| Operatoren | 54 |
| Variablen und Werte | 55 |
| Anweisungen und Strukturen | 55 |
| Kommentare | 56 |
| Ein typisches Programm in C betrachten | 58 |
| Programmstrukturen in C verstehen | 58 |
| Die Funktion main() bestimmen | 59 |
| Etwas an das Betriebssystem zurückgeben | 60 |
| Eine Funktion hinzufügen | 61 |
| TEIL II: | |
| DAS KLEINE EINMALEINS DER C-PROGRAMMIERUNG | 65 |
| Kapitel 4 | |
| Versuch und Irrtum | 67 |
| Zeug auf dem Bildschirm anzeigen | 67 |
| Eine witzige Botschaft anzeigen | 67 |
| Die Funktion puts() einführen | 68 |
| Weiteren Text hinzufügen | 69 |
| Eine Anweisung auskommentieren | 70 |
| Mit Absicht Mist bauen | 71 |
| Mehr Anzeigeblödsinn | 73 |
| Text mithilfe von printf() anzeigen | 73 |
| Die Funktion printf() einführen | 74 |
| Den Zeilenvorschub verstehen | 74 |
| Escape-Sequenzen einsetzen | 75 |
| Nochmal mit Absicht Mist bauen | 76 |
| Kapitel 5 | |
| Werte und Konstanten | 79 |
| Ein Treffpunkt für verschiedene Werte | 79 |
| Werte verstehen | 79 |
| Werte mithilfe von printf() anzeigen | 81 |
| Die zusätzlichen Nullen beachten | 82 |
| Der Computer macht die Mathematik | 83 |
| Einfache Arithmetik ins Spiel bringen | 83 |
| Die Sache mit den ganzen Zahlen und den Gleitkommazahlen überdenken | 84 |
| Immer das Gleiche | 86 |
| Den gleichen Wert immer wieder verwenden | 86 |
| Konstanten einführen | 87 |
| Konstanten verwenden | 87 |
| Kapitel 6 | |
| Ein Abstellplatz | 89 |
| Werte, die variieren | 89 |
| Ein schnelles Beispiel erstellen | 89 |

| | |
|--|----|
| Typen von Variablen einführen | 90 |
| Variablen verwenden | 91 |
| Variabler Wahnsinn | 94 |
| Charakteristischere Variablentypen verwenden | 94 |
| Mehrere Variablen erzeugen | 95 |
| Einen Wert bei der Erzeugung zuordnen | 97 |
| Variablen wiederverwenden | 97 |

Kapitel 7

Eingabe und Ausgabe 99

| | |
|--|-----|
| Ein- und Ausgabe von Zeichen | 99 |
| Eingabe- und Ausgabegeräte verstehen | 99 |
| Zeichen mit getchar() abrufen | 100 |
| Die Funktion putchar() verwenden | 102 |
| Mit Zeichen-Variablen arbeiten | 103 |
| Text-Ein- und Ausgabe, aber meistens Eingabe | 104 |
| Strings speichern | 104 |
| Die Funktion scanf() einführen | 105 |
| Einen String mit scanf() lesen | 106 |
| Werte mit scanf() lesen | 107 |
| fgets() zur Texteingabe verwenden | 108 |

Kapitel 8

Entscheidungsfindung 111

| | |
|---|-----|
| Was wäre, wenn? | 111 |
| Einen einfachen Vergleich anstellen | 111 |
| Das Schlüsselwort if einführen | 112 |
| Werte auf verschiedene Arten vergleichen | 113 |
| Den Unterschied zwischen = und == kennen | 115 |
| Vergessen, wo das Semikolon stehen muss | 116 |
| Mehrere Entscheidungen | 116 |
| Unübersichtlichere Entscheidungen fällen | 117 |
| Eine dritte Möglichkeit hinzufügen | 118 |
| Mehrere Vergleiche mithilfe der Logik durchführen | 118 |
| Einen logischen Vergleich erstellen | 119 |
| Einige logische Operatoren hinzufügen | 119 |
| Der alte Trick | |
| mit der switch-case-Anweisung | 120 |
| Multiple-Choice-Entscheidungen treffen | 120 |
| Die switch-case-Struktur verstehen | 122 |
| Keine Pausen einlegen | 123 |
| Das verrückte Werkzeug ? | 124 |

Kapitel 9

Schleifen, Schleifen, Schleifen 125

| | |
|------------------------------|-----|
| Ein kleines Déjà-Vu | 125 |
| Der Kick der Schleifen | 126 |

16 Inhaltsverzeichnis

| | |
|---|-----|
| Etwas x-mal machen | 126 |
| Die Schleife for einführen | 127 |
| Mit der Anweisung for zählen | 129 |
| Buchstabenschleifen durchlaufen | 129 |
| Schleifen verschachteln | 130 |
| Die Freuden der while-Schleife | 131 |
| Eine while-Schleife strukturieren | 132 |
| Die do-while-Schleife verwenden | 133 |
| Und noch mehr zum Thema Schleifen | 135 |
| Endlosschleifen | 135 |
| Endlosschleifen absichtlich verwenden | 136 |
| Aus einer Endlosschleife entkommen | 137 |
| Eine Schleife vermässeln | 138 |

Kapitel 10

Mit Funktionen Spaß haben 141

| | |
|--|-----|
| Anatomie einer Funktion | 141 |
| Eine Funktion erstellen | 141 |
| Prototyping (oder nicht) | 144 |
| Funktionen und Variablen | 146 |
| Variablen in Funktionen verwenden | 146 |
| Einen Wert an eine Funktion übergeben | 147 |
| Mehrere Werte zu einer Funktion schicken | 149 |
| Funktionen erzeugen, die Werte zurückgeben | 149 |
| Vorzeitig zurückgeben | 152 |

TEIL III:

AUF DEM VORHANDENEN WISSEN AUFBAUEN 155

Kapitel 11

Das unvermeidliche Mathematikkapitel 157

| | |
|--|-----|
| Mathematische Operatoren außerhalb der Unendlichkeit | 157 |
| Inkrementieren und Dekrementieren | 158 |
| Die Operatoren ++ und -- voranstellen | 160 |
| Den Rest (Modulo) entdecken | 161 |
| Mit Zuweisungsoperatoren Zeit sparen | 162 |
| Der Wahnsinn: Mathematische Funktionen | 163 |
| Gebräuchliche mathematische Funktionen erkunden | 163 |
| Die Trigonometrie erleiden | 165 |
| Es ist absolut zufällig | 167 |
| Zufallszahlen ausspucken | 168 |
| Die Zahlen noch zufälliger machen | 169 |
| Die heilige Rangordnung | 171 |
| In die richtige Reihenfolge bringen | 171 |
| Mithilfe von Klammern die Rangordnung ändern | 172 |

Kapitel 12**Her mit den Arrays. 173**

| | |
|---|-----|
| Schauen Sie sich das Array an | 173 |
| Arrays vermeiden | 173 |
| Arrays verstehen | 174 |
| Initialisierung eines Arrays | 176 |
| Mit char-Arrays (Strings) spielen | 177 |
| Mit leeren char-Arrays arbeiten | 178 |
| Arrays sortieren | 180 |
| Mehrdimensionale Arrays | 182 |
| Ein zweidimensionales Array | 183 |
| Mit dreidimensionalen Arrays ausflippen | 185 |
| Ein initialisiertes mehrdimensionales Array deklarieren | 186 |
| Arrays und Funktionen | 187 |
| Ein Array an eine Funktion übergeben | 188 |
| Eine Funktion gibt ein Array zurück. | 189 |

Kapitel 13**Mit Text Spaß haben 191**

| | |
|--|-----|
| Funktionen zum Verändern von Zeichen | 191 |
| CTYPEs einführen | 191 |
| Zeichen testen | 193 |
| Zeichen ändern | 194 |
| Massenweise Stringfunktionen | 195 |
| Stringfunktionen betrachten | 195 |
| Text vergleichen | 196 |
| Strings erstellen | 198 |
| Mit der Formatierung printf() Spaß haben | 198 |
| Gleitkommazahlen formatieren | 199 |
| Die Ausgabebreite festlegen | 200 |
| Die Ausgabe ausrichten | 202 |
| Sanft den Fluss hinab | 203 |
| Das Flussprinzip veranschaulichen | 203 |
| Mit dem Flussprinzip arbeiten | 204 |

Kapitel 14**Die multivariablen Strukturen 207**

| | |
|--|-----|
| Hallo Struktur | 207 |
| Die Vielfachvariable einführen | 207 |
| struct verstehen | 208 |
| Eine Struktur füllen | 210 |
| Arrays aus Strukturen erzeugen | 211 |
| Sonderbare Strukturkonzepte | 213 |
| Strukturen in Strukturen packen | 213 |
| Eine Struktur an eine Funktion weitergeben | 215 |

Kapitel 15

| | |
|---|------------|
| Das Leben in der Kommandozeile | 217 |
| Ein Bildschirmfenster hervorzaubern | 217 |
| Ein Bildschirmfenster öffnen | 217 |
| Im Textmodus Code starten | 218 |
| Die Argumente der Funktion main() | 219 |
| Die Kommandozeile lesen | 220 |
| Die Argumente von main() verstehen | 221 |
| Zeit zum Aussteigen | 223 |
| Das Programm verlassen | 223 |
| Ein anderes Programm starten | 224 |

Kapitel 16

| | |
|---|------------|
| Variabler Blödsinn | 225 |
| Variablenkontrolle | 225 |
| Typumwandlung bis zur Ungläubigkeit | 225 |
| Neue Dinge mit typedef erzeugen | 227 |
| Statische Variablen erzeugen | 229 |
| Variablen – Variablen überall | 231 |
| Globale Variablen verwenden | 232 |
| Eine globale Strukturvariable | 233 |

Kapitel 17

| | |
|---|------------|
| Binärer Wahnsinn | 237 |
| Binäre Grundlagen | 237 |
| Die Zweiheit verstehen | 237 |
| Binäre Werte anzeigen | 239 |
| Die Manipulation von Bits | 241 |
| Den bitweise arbeitenden Operator verwenden | 241 |
| & bitweise verwenden | 243 |
| Ausschließlich mit dem exklusiven ODER (XOR) arbeiten | 244 |
| Die Operatoren ~ und ! verstehen | 246 |
| Binäre Werte verschieben | 246 |
| Die Funktion binbin() erklären | 250 |
| Viel Freude mit Hex | 250 |

TEIL IV:

| | |
|-----------------------------------|------------|
| FÜR FORTGESCHRITTENE | 255 |
|-----------------------------------|------------|

Kapitel 18

| | |
|---|------------|
| Einführung in Pointer | 257 |
| Das größte Problem bei Pointern | 257 |
| Die Größe des Variablenspeichers abschätzen | 258 |
| Die Speicherung von Variablen verstehen | 258 |

| | |
|---|-----|
| Die Größe einer Variablen auslesen | 259 |
| Die Adresse einer Variablen überprüfen | 262 |
| Die Informationen zur Speicherung von Variablen zusammenfassen | 265 |
| Das scheußlich komplexe Thema der Pointer | 265 |
| Pointer einführen | 265 |
| Mit Pointern arbeiten | 268 |

Kapitel 19

Tief im Pointer-Land 271

| | |
|--|-----|
| Pointer und Arrays | 271 |
| Die Adresse eines Arrays erhalten | 271 |
| Pointer-Mathematik in einem Array | 273 |
| Die Array-Notation durch Pointer ersetzen | 277 |
| Strings sind Pointer-Dinge | 278 |
| Zum Anzeigen eines Strings Pointer verwenden | 279 |
| Einen String durch die Verwendung eines Pointers deklarieren | 280 |
| Ein Array aus Pointern erstellen | 281 |
| Strings sortieren | 284 |
| Pointer in Funktionen | 285 |
| Einen Pointer an eine Funktion weitergeben | 285 |
| Von einer Funktion einen Pointer zurückbekommen | 286 |

Kapitel 20

Verkettete Listen 289

| | |
|--|-----|
| Her mit dem Speicher! | 289 |
| Die Funktion malloc() einführen | 290 |
| Strings speichern | 291 |
| Speicherplatz frei geben | 293 |
| Erfassen Sie diese Verkettung | 295 |
| Platz für eine Struktur zuteilen | 295 |
| Eine verkettete Liste erzeugen | 297 |
| Eine verkettete Liste bearbeiten | 302 |
| Eine verkettete Liste speichern | 307 |

Kapitel 21

Es wird Zeit 309

| | |
|------------------------------------|-----|
| Wie spät ist es? | 309 |
| Den Kalender verstehen | 309 |
| Mit der Zeit in C arbeiten | 310 |
| Zeit zu programmieren | 311 |
| Die Uhr kontrollieren | 311 |
| Einen Zeitstempel betrachten | 313 |
| Den Zeitstring aufschneiden | 314 |
| Schlummern | 315 |

**TEIL V:
UND DER GANZE REST 317**

**Kapitel 22
Dauerhafte Speicherfunktionen..... 319**

| | |
|--|-----|
| Sequenzieller Dateizugriff | 319 |
| Den Zugriff auf Dateien in C verstehen | 319 |
| Text in eine Datei schreiben | 320 |
| Text aus einer Datei auslesen | 321 |
| Text an eine Datei anhängen | 323 |
| Binäre Daten schreiben..... | 324 |
| Mit Dateien mit binären Daten arbeiten..... | 326 |
| Direktzugriffsdateien..... | 328 |
| Eine Struktur in eine Datei schreiben | 328 |
| Lesen und zurückspulen..... | 331 |
| Eine bestimmte Aufzeichnung finden..... | 332 |
| Eine verkettete Liste in einer Datei speichern | 334 |

**Kapitel 23
Verwaltung von Dateien..... 335**

| | |
|--|-----|
| Verzeichnis-Wahnsinn | 335 |
| Ein Verzeichnis aufrufen | 335 |
| Weitere Informationen über Dateien sammeln | 337 |
| Dateien von Verzeichnissen trennen | 339 |
| Den Verzeichnisbaum erkunden | 340 |
| Rund um Dateien..... | 341 |
| Eine Datei umbenennen | 341 |
| Eine Datei kopieren | 342 |
| Eine Datei löschen | 343 |

**Kapitel 24
Projekte für Fortgeschrittene und Hartgesottene..... 345**

| | |
|---|-----|
| Ein aus zahlreichen Modulen bestehendes Monster..... | 345 |
| Zwei Quelltextdateien verbinden..... | 345 |
| Variablen, die von mehreren Modulen gemeinsam genutzt werden..... | 348 |
| Eine benutzerdefinierte Header-Datei erstellen | 349 |
| Mit anderen Bibliotheken verbinden | 352 |

**Kapitel 25
Programmfehler raus!..... 355**

| | |
|-----------------------------------|-----|
| Debugger von Code::Blocks | 355 |
| Die Einstellung Debugging | 355 |
| Der Debugger bei der Arbeit | 357 |
| Einen Haltepunkt markieren | 359 |
| Variablen ansehen | 360 |

| | |
|---|-----|
| Probleme durch die Verwendung von printf() und puts() lösen | 362 |
| Probleme dokumentieren. | 362 |
| Kommentare für Ihr zukünftiges Ich speichern. | 362 |
| Verbesserte Fehlermeldungen | 363 |

TEIL VI:

DER TOP-TEN-TEIL 367

Kapitel 26

Zehn gängige Schnitzer 369

| | |
|---|-----|
| Bedingungsfehler | 369 |
| == vs = | 370 |
| Gefährliche Semikolons in Schleifen. | 370 |
| Kommata in for-Schleifen. | 371 |
| Fehlende Pause in einer switch-Struktur. | 371 |
| Fehlende runde und geschweifte Klammern | 372 |
| Beherzigen Sie diese Warnung | 373 |
| Endlosschleifen | 374 |
| Patzter bei scanf() | 374 |
| Fließende Eingabebeschränkungen | 375 |

Anhang A: ASCII-Codes 377

Anhang B: Schlüsselwörter 382

Anhang C: Operatoren 383

Anhang D: Typen von Variablen 384

Anhang E: Escape-Sequenzen 385

Anhang F: Umwandlungszeichen 386

Anhang G: Rangordnung 388

Stichwortverzeichnis 390