

# Inhalt

Materialien zum Buch .....	16
Vorwort .....	17
<b>1 Der Einstieg in die Welt von C</b>	<b>21</b>
<hr/>	
<b>1.1 Die Sprache C</b> .....	<b>21</b>
<b>1.2 Die C-Standardbibliothek</b> .....	<b>23</b>
1.2.1 Header-Dateien und Programmbibliothek .....	24
<b>1.3 Die nötigen Werkzeuge für C</b> .....	<b>28</b>
<b>1.4 Übersetzen mit der Entwicklungsumgebung</b> .....	<b>32</b>
<b>1.5 Übersetzen mit GCC und Clang</b> .....	<b>35</b>
<b>1.6 Listings zum Buch</b> .....	<b>37</b>
<b>1.7 Kontrollfragen und Aufgaben im Buch</b> .....	<b>38</b>
<b>1.8 Aufgabe</b> .....	<b>38</b>
<b>2 Erste Schritte in C</b>	<b>41</b>
<hr/>	
<b>2.1 Das erste Programm genauer betrachtet</b> .....	<b>41</b>
<b>2.2 Die Funktion printf()</b> .....	<b>44</b>
<b>2.3 Zeichensätze in C</b> .....	<b>46</b>
2.3.1 Basis-Ausführungszeichensatz .....	46
2.3.2 Einige wichtige Escape-Sequenzen .....	47
<b>2.4 Symbole in C</b> .....	<b>49</b>
2.4.1 Bezeichner .....	49
2.4.2 Reservierte Schlüsselwörter .....	51
2.4.3 Literale .....	52
2.4.4 Einfache Begrenzer .....	54

## Inhalt

<b>2.5 Kommentare .....</b>	55
<b>2.6 Kontrollfragen und Aufgaben .....</b>	56
<b>3 Basisdatentypen in C</b>	59
<hr/>	
<b>3.1 Variablen .....</b>	59
<b>3.2 Deklaration und Definition .....</b>	60
<b>3.3 Initialisierung und Zuweisung von Werten .....</b>	62
<b>3.4 Datentypen für Ganzzahlen .....</b>	63
3.4.1 Vorzeichenlos und vorzeichenbehaftet .....	66
3.4.2 Suffixe für Ganzzahlen .....	69
<b>3.5 Datentypen für Zeichen .....</b>	70
3.5.1 Der Datentyp char .....	71
3.5.2 Der Datentyp wchar_t .....	72
3.5.3 Unicode-Unterstützung .....	73
<b>3.6 Datentypen für Fließkommazahlen .....</b>	74
3.6.1 Suffixe für Fließkommazahlen .....	76
3.6.2 Komplexe Gleitkommatypen .....	76
<b>3.7 Boolescher Datentyp .....</b>	78
<b>3.8 Speicherbedarf mit sizeof ermitteln .....</b>	80
<b>3.9 Die Wertebereiche der Datentypen ermitteln .....</b>	82
3.9.1 Limits von Integer-Typen .....	83
3.9.2 Limits von Fließkommazahlen .....	85
3.9.3 Integer-Typen mit fester Größe verwenden .....	86
3.9.4 Sicherheit beim Kompilieren mit static_assert .....	87
<b>3.10 Konstanten erstellen .....</b>	88
<b>3.11 Lebensdauer und Sichtbarkeit von Variablen .....</b>	89
<b>3.12 void – ein unvollständiger Typ .....</b>	91
<b>3.13 Kontrollfragen und Aufgaben .....</b>	91

---

<b>4 Rechnen mit C und Operatoren</b>	93
<b>4.1 Werte formatiert einlesen mit scanf()</b>	93
<b>4.2 Operatoren</b>	97
<b>4.3 Arithmetische Operatoren</b>	101
<b>4.4 Der Inkrement- und der Dekrement-Operator</b>	103
<b>4.5 Bit-Operatoren</b>	106
<b>4.6 Implizite Typumwandlung</b>	112
4.6.1 Arithmetische Umwandlung	113
4.6.2 Typpromotionen	115
<b>4.7 Explizite Typumwandlung</b>	116
<b>4.8 Mathematische Funktionen in C</b>	117
<b>4.9 Kontrollfragen und Aufgaben</b>	121
<b>5 Bedingte Anweisungen und Verzweigungen</b>	125
<b>5.1 Bedingte Anweisungen</b>	125
5.1.1 Bedingte Anweisungen mit if	126
5.1.2 Vergleichsoperatoren	128
5.1.3 Mini-Exkurs zum Anweisungsblock	131
<b>5.2 Die alternative Verzweigung</b>	131
<b>5.3 Der Bedingungsoperator ?:</b>	134
<b>5.4 Die mehrfache Verzweigung mit if und else if</b>	135
5.4.1 Verschachteln von Verzweigungen	138
<b>5.5 Mehrfache Verzweigung mit switch</b>	140
5.5.1 Austritt aus der Fallunterscheidung mit break	143
<b>5.6 Logische Verknüpfungen</b>	147
5.6.1 Der !-Operator	147

5.6.2	Der &&-Operator – Logisches UND .....	150
5.6.3	Der   -Operator – Logisches ODER .....	151
<b>5.7</b>	<b>Kontrollfragen und Aufgaben .....</b>	<b>154</b>
<hr/>		
<b>6</b>	<b>Schleifen – Programmteile wiederholen</b>	<b>157</b>
<hr/>		
<b>6.1</b>	<b>Die Zählschleife – for .....</b>	<b>157</b>
<b>6.2</b>	<b>Die kopfgesteuerte while-Schleife .....</b>	<b>162</b>
<b>6.3</b>	<b>Die fußgesteuerte do-while-Schleife .....</b>	<b>165</b>
<b>6.4</b>	<b>Kontrolliertes Verlassen von Schleifen .....</b>	<b>168</b>
6.4.1	break .....	168
6.4.2	continue .....	169
<b>6.5</b>	<b>Kontrollfragen und Aufgaben .....</b>	<b>171</b>
<hr/>		
<b>7</b>	<b>Funktionen erstellen</b>	<b>173</b>
<hr/>		
<b>7.1</b>	<b>Funktionen definieren .....</b>	<b>174</b>
<b>7.2</b>	<b>Funktionen aufrufen .....</b>	<b>175</b>
<b>7.3</b>	<b>Funktionsdeklaration (Vorausdeklaration) .....</b>	<b>176</b>
<b>7.4</b>	<b>Funktionsparameter .....</b>	<b>179</b>
<b>7.5</b>	<b>Rückgabewert von Funktionen .....</b>	<b>181</b>
7.5.1	Die return-Anweisung .....	183
<b>7.6</b>	<b>Exkurs: Funktionen bei der Ausführung .....</b>	<b>185</b>
<b>7.7</b>	<b>Inline-Funktionen .....</b>	<b>186</b>
<b>7.8</b>	<b>Rekursionen .....</b>	<b>188</b>
<b>7.9</b>	<b>Die main()-Funktion .....</b>	<b>190</b>
7.9.1	Der Rückgabewert von main() an das Betriebssystem .....	190

<b>7.10 Programme mit exit() beenden .....</b>	192
<b>7.11 Globale, lokale und statische Variablen .....</b>	194
7.11.1 Lokale Variablen .....	195
7.11.2 Globale Variablen .....	196
7.11.3 Die Speicherklasse static .....	199
7.11.4 Die Speicherklasse extern .....	202
<b>7.12 Kontrollfragen und Aufgaben .....</b>	202

---

## **8 Präprozessor-Direktiven** 205

<b>8.1 Dateien einfügen mit #include .....</b>	206
<b>8.2 Konstanten und Makros mit #define und #undef .....</b>	208
8.2.1 Symbolische Konstanten mit #define definieren ....	208
8.2.2 Makros mit #define definieren .....	211
8.2.3 Symbolische Konstanten und Makros aufheben (#undef) .....	214
<b>8.3 Bedingte Kompilierung .....</b>	215
8.3.1 Mehrfaches Inkludieren vermeiden .....	218
<b>8.4 Programmdiagnose mit assert() .....</b>	222
<b>8.5 Generische Auswahl .....</b>	223
<b>8.6 Eigene Header-Dateien erstellen .....</b>	225
<b>8.7 Kontrollfragen und Aufgaben .....</b>	227

---

## **9 Arrays und Zeichenketten (Strings)** 231

<b>9.1 Was genau sind Arrays? .....</b>	231
9.1.1 Arrays definieren .....	232
9.1.2 Arrays mit Werten versehen und auf sie zugreifen .....	233

## Inhalt

9.1.3	Initialisierung mit einer Initialisierungsliste .....	238
9.1.4	Bestimmte Elemente direkt initialisieren .....	239
9.1.5	Arrays mit Schreibschutz .....	240
9.1.6	Arrays mit fester und variabler Länge (VLA) .....	240
9.1.7	Arrays mit scanf einlesen .....	242
9.1.8	Arrays an Funktionen übergeben .....	243
<b>9.2</b>	<b>Mehrdimensionale Arrays .....</b>	<b>245</b>
9.2.1	Zweidimensionalen Arrays Werte zuweisen und auf sie zugreifen .....	246
9.2.2	Zweidimensionale Arrays an eine Funktion übergeben .....	249
9.2.3	Mehr als zwei Dimensionen verwenden .....	251
<b>9.3</b>	<b>Strings (Zeichenketten) .....</b>	<b>252</b>
9.3.1	Strings initialisieren .....	252
9.3.2	Strings einlesen .....	254
9.3.3	Unicode-Unterstützung .....	256
9.3.4	String-Funktionen der Standardbibliothek <string.h> .....	258
9.3.5	Sicherere Funktionen zum Schutz vor Speicherüberschreitungen .....	261
9.3.6	Umwandlungsfunktionen zwischen Zahlen und Strings .....	262
<b>9.4</b>	<b>Kontrollfragen und Aufgaben .....</b>	<b>262</b>
<b>10</b>	<b>Zeiger (Pointer)</b>	<b>265</b>
<b>10.1</b>	<b>Zeiger vereinbaren .....</b>	<b>265</b>
<b>10.2</b>	<b>Zeiger verwenden .....</b>	<b>267</b>
10.2.1	Explizite Typumwandlung für den byteweisen Zugriff .....	268

<b>10.3 Zugriff auf den Inhalt von Zeigern .....</b>	269
10.3.1 Der NULL-Zeiger .....	272
10.3.2 Deklaration, Adressierung und Dereferenzierung von Zeigern .....	273
<b>10.4 Zeiger als Funktionsparameter .....</b>	274
<b>10.5 Zeiger als Rückgabewert .....</b>	275
<b>10.6 Zeigerarithmetik .....</b>	278
<b>10.7 Zugriff auf Array-Elemente über Zeiger .....</b>	279
<b>10.8 Arrays und Zeiger als Funktionsparameter .....</b>	283
10.8.1 Der const-Array-Parameter .....	283
<b>10.9 char-Arrays und Zeiger .....</b>	285
<b>10.10 Arrays von Zeigern .....</b>	287
<b>10.11 void-Zeiger .....</b>	290
<b>10.12 Typqualifizierer bei Zeigern .....</b>	291
10.12.1 Konstanter Zeiger .....	291
10.12.2 Zeiger für konstante Daten .....	292
10.12.3 Konstante Zeiger und Zeiger für konstante Daten .....	293
10.12.4 Konstante Parameter für Funktionen .....	293
10.12.5 restrict-Zeiger .....	294
<b>10.13 Zeiger auf Funktionen .....</b>	296
<b>10.14 Kontrollfragen und Aufgaben .....</b>	300

---

## **11 Dynamische Speicherverwaltung**

<b>11.1 Neuen Speicher zur Laufzeit reservieren .....</b>	304
<b>11.2 Speicherblöcke vergrößern oder verkleinern .....</b>	310

<b>11.3 Speicherblöcke wieder freigeben</b> .....	314
11.3.1 Memory Leaks (Speicherlecks) .....	315
<b>11.4 Kontrollfragen und Aufgaben</b> .....	317
<b>12 Komplexe Datentypen</b>	319
<hr/>	
<b>12.1 Strukturen</b> .....	320
12.1.1 Strukturtypen deklarieren .....	320
12.1.2 Definition einer Strukturvariablen .....	322
12.1.3 Erlaubte Operationen auf Strukturvariablen .....	323
12.1.4 Deklaration und Definition zusammenfassen .....	323
12.1.5 Synonyme für Strukturtypen erstellen .....	324
12.1.6 Selektion von Komponenten einer Strukturvariablen .....	325
12.1.7 Strukturen initialisieren .....	329
12.1.8 Nur bestimmte Komponenten einer Strukturvariablen initialisieren .....	330
12.1.9 Zuweisung bei Strukturvariablen .....	331
12.1.10 Größe und Speicherausrichtung einer Struktur .....	331
12.1.11 Strukturen vergleichen .....	332
12.1.12 Strukturen, Funktionen und Strukturzeiger .....	332
12.1.13 Arrays von Strukturvariablen .....	336
12.1.14 Strukturvariablen als Komponenten in Strukturen .....	339
12.1.15 Zeiger als Komponenten in Strukturen .....	345
<b>12.2 Unions</b> .....	348
<b>12.3 Der Aufzählungstyp enum</b> .....	351
<b>12.4 Eigene Typen mit <code>typedef</code> deklarieren</b> .....	353
<b>12.5 Kontrollfragen und Aufgaben</b> .....	355

---

<b>13 Dynamische Datenstrukturen</b>	359
<b>13.1 Einfach verkettete Listen</b> .....	360
13.1.1 Ein neues Element in die Liste einfügen .....	366
13.1.2 Ein Element suchen und ausgeben .....	369
13.1.3 Ein Element aus der Liste entfernen .....	370
13.1.4 Das erste Element in der Liste löschen .....	371
13.1.5 Ein beliebiges Element in der Liste löschen .....	372
<b>13.2 Doppelt verkettete Listen</b> .....	373
<b>13.3 Kontrollfragen und Aufgaben</b> .....	375
<b>14 Eingabe- und Ausgabefunktionen</b>	377
<b>14.1 Verschiedene Streams und Standard-Streams</b> .....	377
14.1.1 Streams im Textmodus .....	378
14.1.2 Streams im binären Modus .....	379
14.1.3 Standard-Streams .....	379
<b>14.2 Dateien</b> .....	380
<b>14.3 Dateien öffnen</b> .....	382
14.3.1 Exklusiver Dateizugriff .....	385
14.3.2 Weitere Dateifunktionen .....	386
<b>14.4 Dateien schließen</b> .....	388
14.4.1 Das Limit für maximal geöffnete Dateien .....	388
<b>14.5 Auf Fehler oder das Dateiende prüfen</b> .....	389
14.5.1 Der End-of-File Indicator .....	389
14.5.2 Error Indicator .....	390
14.5.3 Informationen zum Fehler ausgeben .....	390
14.5.4 Fehler- und EOF-Flag zurücksetzen – clearerr() .....	391
<b>14.6 Weitere Funktionen für die Ein- und Ausgabe</b> .....	391
14.6.1 Einzelne Zeichen aus einem Stream lesen .....	392

14.6.2	Zeichen in den Stream zurückstellen .....	392
14.6.3	Einzelne Zeichen in einen Stream schreiben .....	392
14.6.4	Zeilenweise aus einem Stream lesen .....	395
14.6.5	Zeilenweise in einen Stream schreiben .....	395
14.6.6	Lesen und Schreiben in ganzen Blöcken .....	400
<b>14.7</b>	<b>Funktionen zur formatierten Ein-/Ausgabe .....</b>	<b>405</b>
14.7.1	Funktionen zur formatierten Ausgabe .....	405
14.7.2	Umwandlungsvorgaben für die printf-Familie .....	407
14.7.3	Weite und Feldbreite .....	408
14.7.4	Flags .....	409
14.7.5	Genauigkeitsangaben für Zahlen bei printf() .....	410
14.7.6	Funktionen für die formatierte Eingabe .....	414
14.7.7	Umwandlungsvorgaben für die scanf()-Familie .....	415
14.7.8	Suchmengenkonvertierung .....	416
<b>14.8</b>	<b>Wahlfreier Dateizugriff .....</b>	<b>418</b>
14.8.1	Die aktuelle Dateiposition ermitteln .....	418
14.8.2	Die aktuelle Dateiposition ändern .....	418
<b>14.9</b>	<b>Sicherere Funktionen mit C11 .....</b>	<b>422</b>
<b>14.10</b>	<b>Datei löschen oder umbenennen .....</b>	<b>423</b>
<b>14.11</b>	<b>Pufferung .....</b>	<b>423</b>
<b>14.12</b>	<b>Kontrollfragen und Aufgaben .....</b>	<b>425</b>
<b>15</b>	<b>Zeitroutinen (time.h)</b>	<b>429</b>
<b>15.1</b>	<b>Die Funktion clock() .....</b>	<b>429</b>
<b>15.2</b>	<b>Erweiterte Zeitfunktionen .....</b>	<b>431</b>
15.2.1	Eine plattformunabhängige delay()-Funktion .....	433
15.2.2	Der Datentyp time_t .....	434
15.2.3	Der Datentyp struct tm .....	436

<b>15.3 Kontrollfragen und Aufgaben .....</b>	441
<b>15.4 Schlusswort .....</b>	443
<b>Anhang</b>	445
<hr/>	
<b>A    Übersichtstabellen wichtiger Sprachelemente .....</b>	445
A.1    Operatorpriorität (Operator Precedence) .....	445
A.2    Reservierte Schlüsselwörter in C .....	447
A.3    Header-Dateien der Standardbibliothek .....	448
A.4    Kommandozeilenargumente .....	451
A.5    Weiterführende Ressourcen .....	454
<b>B    Lösungen der Übungsaufgaben .....</b>	455
B.1    Antworten und Lösungen zu Kapitel 2 .....	455
B.2    Antworten und Lösungen zu Kapitel 3 .....	456
B.3    Antworten und Lösungen zu Kapitel 4 .....	457
B.4    Antworten und Lösungen zu Kapitel 5 .....	459
B.5    Antworten und Lösungen zu Kapitel 6 .....	461
B.6    Antworten und Lösungen zu Kapitel 7 .....	464
B.7    Antworten und Lösungen zu Kapitel 8 .....	467
B.8    Antworten und Lösungen zu Kapitel 9 .....	470
B.9    Antworten und Lösungen zu Kapitel 10 .....	474
B.10    Antworten und Lösungen zu Kapitel 11 .....	480
B.11    Antworten und Lösungen zu Kapitel 12 .....	484
B.12    Antworten und Lösungen zu Kapitel 13 .....	487
B.13    Antworten und Lösungen zu Kapitel 14 .....	492
B.14    Antworten und Lösungen zu Kapitel 15 .....	495
<b>Index .....</b>	499