

Karlheinz Zeiner

# Programmieren lernen mitC

3., überarbeitete und aktualisierte Auflage



Carl Hanser Verlag München Wien

## Überblick

### Teil I

<b>1</b>	<b>Programmieren lernen - Grundlagen.....</b>	<b>1</b>
1.1	Zwei kleine C Programme.....	1
1.2	Grundbegriffe der Syntax, Kommentare.....	9
1.3	Datentypen und Ausdrücke.....	14
1.4	Algorithmen und Kontrollstrukturen.....	21
1.5	Vektoren und Zeichenketten.....	38
1.6	Adressen und Zeiger.....	45
1.7	Modularisierung, Funktionen.....	49
1.8	Ein-/Ausgabe.....	68

### Teil 2

<b>2</b>	<b>Datendarstellung in Rechnern, elementare Datentypen.....</b>	<b>79</b>
<b>3</b>	<b>Operatoren und Ausdrücke, mathematische Funktionen.....</b>	<b>109</b>
<b>4</b>	<b>Kontrollstrukturen, strukturierte Programmierung.....</b>	<b>127</b>
<b>5</b>	<b>Funktionen, Speicherklassen und Modulkonzept.....</b>	<b>159</b>
<b>6</b>	<b>Grundsätze für die Programmentwicklung.....</b>	<b>197</b>
<b>7</b>	<b>Der Preprozessor.....</b>	<b>211</b>
<b>8</b>	<b>Bit-Operatoren und Ausdrücke.....</b>	<b>221</b>
<b>9</b>	<b>Zeiger.....</b>	<b>229</b>
<b>10</b>	<b>Vektoren.....</b>	<b>243</b>
<b>11</b>	<b>Zeichenketten.....</b>	<b>269</b>
<b>12</b>	<b>Strukturen, dynamische Datenstrukturen, Unionen.....</b>	<b>283</b>
<b>13</b>	<b>Dateien (Files).....</b>	<b>301</b>

Das Buch gliedert sich in zwei Teile. Der erste Teil behandelt die Grundlagen des Programmierens. Die weiteren Kapitel vertiefen die Themen des ersten Teils oder beschäftigen sich mit zusätzlichen Themen.

**Falls** Sie noch in keiner Sprache programmiert haben, **dann** sollten Sie sich mit dem 1. Teil solange auseinandersetzen, bis sie die Übungsaufgaben lösen und die Kontrollfragen beantworten können, **sonst** können Sie das 1. Kapitel als Leitfaden durch die wichtigsten Themen betrachten, um je nach Bedarf zu den weiteren Kapiteln zu wechseln.

Die Reihenfolge der Kapitel 2 bis 13 ist zwar sinnvoll, aber nicht unbedingt zwingend. Neben dem 1. Teil beschäftigen sich die Kapitel 5 (Modulkonzept) und 9 (Zeiger) mit fundamentalen Konzepten in C.

## Inhaltsverzeichnis

Programmieren lernen - Grundlagen.....	1
<b>1.1 Zwei kleine C Programme.....</b>	<b>1</b>
<b>1.2 Grundbegriffe der Syntax, Kommentare.....</b>	<b>9</b>
<b>1.2.1 Zeichensatz und Wörter.....</b>	<b>9</b>
1.2.2 Syntaxnotation.....	10
1.2.3 Schlüsselwörter.....	11
1.2.4 Kommentare.....	12
1.2.5 Fehlermeldungen des Compilers.....	13
<b>1.3 Datentypen und Ausdrücke.....</b>	<b>14</b>
1.3.1 Datentypen.....	14
1.3.2 Ausdrücke und Operatoren.....	16
<b>1.4 Algorithmen und Kontrollstrukturen.....</b>	<b>21</b>
1.4.1 Algorithmen.....	21
1.4.2 - Sequenz.....	21
1.4.3 Wiederholung.....	22
1.4.4 Auswahl.....	31
1.4.5 Zusammenfassung: Von der Aufgabe zum Programm.....	35
<b>1.5 Vektoren und Zeichenketten.....</b>	<b>38</b>
1.5.1 Vektoren.....	38
1.5.2 Zeichenketten.....	42
<b>1.6 Adressen und Zeiger.....</b>	<b>45</b>
<b>1.7 Modularisierung, Funktionen.....</b>	<b>49</b>
1.7.1 Allgemeines, Begriffe.....	49
1.7.2 Datenaustausch zwischen Funktionen.....	54
1.7.3 Funktionsprototypen.....	65
<b>1.8 Ein-/Ausgabe.....</b>	<b>68</b>
1.8.1 Allgemeines.....	68
1.8.2 Ein-/Ausgabe und Dateien.....	69
1.8.3 Funktionen der Standardbibliothek für die Ein-/Ausgabe.....	71
 Datendarstellung in Rechnern, elementare Datentypen.....	 79
<b>2.1 Allgemeines zu Zahlensystemen und Codes.....</b>	<b>79</b>
<b>2.1.1 Zahlensysteme.....</b>	<b>79</b>
2.1.2 Codes.....	82
<b>2.2 Ganze Zahlen (<i>integer</i>).....</b>	<b>83</b>
2.2.1 Interne Darstellung (Codierung) ganzer Zahlen.....	83
2.2.2 Datentypen für ganze Zahlen in C.....	88
2.2.3 (f)printf und (f)scanf Umwandlungen für ganze Zahlen.....	90

2.3	<b>Zeichen</b> .....	<b>91</b>
2.3.1	Codierung von Zeichen (alphanumerische Daten).....	91
2.3.2	Zeichen ( <i>characters</i> ) und der Datentyp <code>char</code> .....	92
2.3.3	Ein- und Ausgabe von Zeichen, <code>(f)printf</code> und <code>(f)scanf</code> -Umwandlungen.....	93
2.4	<b>Reelle Zahlen</b> .....	<b>96</b>
2.4.1	Festkommazahlen.....	96
2.4.2	Gleitkommadarstellung.....	97
2.4.3	C-Gleitkommatypen ( <i>floating types</i> ).....	99
2.4.4	<code>(f)printf</code> und <code>(f)scanf</code> Umwandlungen für Gleitkommazahlen.....	100
2.5	<b>Der <code>sizeof</code>-Operator</b> .....	<b>102</b>
2.6	<b>Typumwandlungen (<code>caste</code>)</b> .....	<b>103</b>
2.7	<b>Vereinbarungen</b> .....	<b>103</b>
2.8	<b><code>typedef</code> und <code>enum</code></b> .....	<b>106</b>
2.8.1	<code>typedef</code> .....	106
2.8.2	Aufzählungen, <code>enum</code> .....	106
3	<b>Operatoren und Ausdrücke, mathematische Funktionen</b> .....	<b>109</b>
3.1	<b>Arithmetische Operatoren</b> .....	<b>109</b>
3.2	<b>Die Zuweisungsoperatoren, L-Werte</b> .....	<b>110</b>
3.3	<b>Unäre Ausdrücke</b> .....	<b>114</b>
3.4	<b>Vergleichsoperatoren</b> .....	<b>115</b>
3.5	<b>Logische Operatoren und Ausdrücke</b> .....	<b>118</b>
3.6	<b>Der Komma-Operator</b> .....	<b>119</b>
3.7	<b>Implizite Typumwandlung</b> .....	<b>120</b>
3.8	<b>Rangordnung der Operatoren und Reihenfolge der Auswertung</b> .....	<b>121</b>
3.9	<b>Mathematische Funktionen</b> .....	<b>124</b>
4	<b>Kontrollstrukturen, strukturierte Programmierung</b> .....	<b>127</b>
4.1	<b>Steuer- oder Kontrollfluß, Flußdiagramme</b> .....	<b>127</b>
4.2	<b>Strukturierte Programmierung</b> .....	<b>128</b>
4.3	<b>Arten von Strukturblöcken</b> .....	<b>130</b>
4.3.1	Elementarblock.....	130
4.3.2	Sequenz.....	131
4.3.3	Auswahl.....	132
4.3.4	Wiederholung, Iteration.....	140
4.4	<b>C-Spezifisches</b> .....	<b>146</b>
4.4.1	Die Leeraanweisung ( <i>empty-statement</i> ).....	146
4.4.2	Die <code>goto</code> -Anweisung.....	146
4.4.3	Die <code>continue</code> -Anweisung.....	147
4.5	<b>Programmbeispiele</b> .....	<b>148</b>
4.5.1	Kalender.....	148
4.5.2	Rechnen mitzählen in der Zweierkomplementdarstellung.....	152

<b>Funktionen, Speicherklassen und Modulkonzept.....</b>	<b>159</b>
<b>5.1 Definition von Funktionen.....</b>	<b>159</b>
<b>5.2 Funktionsprototypen.....</b>	<b>162</b>
<b>5.3 Aufruf einer Funktion und Übergabemechanismus.....</b>	<b>162</b>
<b>5.4 Die Speicherklasse auto.....</b>	<b>165</b>
<b>5.5 Die Speicherklasse extern.....</b>	<b>167</b>
<b>5.6 Programme mit mehreren Quelltext- bzw. Objektdateien.....</b>	<b>168</b>
<b>5.7 Die Speicherklassen register, static, Besonderheiten.....</b>	<b>171</b>
5.7.1 Die Speicherklasse register.....	171
5.7.2 Die Speicherklasse static.....	172
5.7.3 Externe Variablen mit static-Attribut.....	173
5.7.4 Das static-Attribut bei Funktionen.....	173
<b>5.8 Datenkapselung.....</b>	<b>174</b>
<b>5.9 Verwaltung größerer Programme, Definitionsdateien.....</b>	<b>179</b>
<b>5.10 Make-Utility.....</b>	<b>187</b>
<b>5.11 Rekursion.....</b>	<b>189</b>
<b>Grundsätze für die Programmentwicklung.....</b>	<b>197</b>
<b>6.1 Phasenmodell.....</b>	<b>197</b>
6.1.1 Planungsphase.....	197
6.1.2 Definitionsphase.....	198
6.1.3 Entwurfsphase.....	198
6.1.4 Codieren - Implementieren.....	203
6.1.5 Testen.....	204
6.1.6 Wartung und Pflege.....	205
<b>6.2 Prinzipien der Softwareentwicklung.....</b>	<b>205</b>
<b>6.3 Qualität von Softwareprodukten.....</b>	<b>207</b>
<b>6.4 Dokumentation.....</b>	<b>208</b>
<b>Der Preprozessor.....</b>	<b>211</b>
<b>7.1 Die #include-Anweisung.....</b>	<b>211</b>
<b>7.2 Die #define-Anweisung (1).....</b>	<b>212</b>
<b>7.3 Die #define-Anweisung (2), Makros.....</b>	<b>213</b>
<b>7.4 Bedingte Übersetzung (conditional compilation).....</b>	<b>217</b>
<b>7.5 #line (Zeilennummern).....</b>	<b>219</b>
<b>7*6 Textersatz in Zeichenketten, der Operator #.....</b>	<b>219</b>
<b>7.7 #error und #pragma.....</b>	<b>219</b>

<b>8</b>	<b>Bit-Operatoren und Ausdrücke</b> .....	<b>221</b>
8.1	Bit- und Schiebe-Operatoren.....	221
8.2	Bitmuster und Bit-Masken.....	225
<b>9</b>	<b>Zeiger</b> .....	<b>229</b>
9.1	Einleitende Bemerkungen.....	229
9.2	Vereinbarung von Zeigervariablen und Zeigertypen.....	231
9.3	Inhalts- und Adreßoperator, Wertzuweisungen an Zeiger.....	231
9.4	Zeigerarithmetik.....	234
9.4.1	Summe und Differenz von Zeiger und int.....	234
9.4.2	Vergleichsoperationen mit Zeigern.....	236
9.4.3	Zeigersubtraktion.....	236
9.5	Speicherplatz anfordern und freigeben.....	239
<b>10</b>	<b>Vektoren</b> .....	<b>243</b>
10.1	Eindimensionale Vektoren.....	243
10.1.1	Vereinbarung und Zusammenhang mit Zeigerarithmetik.....	243
10.1.2	Vektoren als formale und aktuelle Parameter von Funktionen.....	244
10.1.3	Dynamische Erzeugung eines Vektors.....	246
10.2	Mehrdimensionale Vektoren.....	247
10.2.1	Definition und Indizierung.....	247
10.2.2	Anordnung der Elemente im Speicher.....	248
10.2.3	Mehrdimensionale Vektoren als Argument von Funktionen.....	251
10.3	Initialisierung von Vektoren.....	252
10.4	Algorithmen und Programmbeispiele.....	252
10.4.1	Quicksort.....	252
10.4.2	Lineare Regression.....	257
10.4.3	Gauß'scher Algorithmus.....	261
<b>11</b>	<b>Zeichenketten</b> .....	<b>269</b>
11.1	Allgemeines, Zeichenketten in C.....	269
11.1.1	Vektoren als Zeichenketten.....	270
11.1.2	Zeiger auf den Typ char.....	271
11.2	Ein-/Ausgabe von Zeichenketten.....	272
11.3	Funktionen für Zeichenketten in der C-Bibliothek.....	274
11.4	Vektoren mit Zeigern auf Zeichenketten.....	274
11.5	Argumente der Funktion main.....	280

12	Strukturen, dynamische Datenstrukturen, Unionen.....	283
12.1	Strukturen.....	283
12.1.1	Typ-Deklarationen und Definition von Variablen.....	283
12.1.2	Zugriff auf Strukturen und die Felder einer Struktur.....	285
12.1.3	Strukturen und Funktionen.....	286
12.1.4	Initialisierung von Strukturen.....	290
12.2	Verkettete Listen, Bäume.....	290
12.3	Unionen.....	298
13	Dateien ( <i>Files</i> ).....	301
13.1	Dateioperationen.....	301
13.2	Fehlererkennung und Behandlung.....	302
13.3	Binärdateien.....	303
13.4	Random Access.....	309
Anhang A:	Die ANSI-C Standard-Bibliothek.....	311
A.1	Diagnose <assert.h>.....	311
A.2	Test und Behandlung von Zeichen <ctype.h>.....	311
A.3	Fehlerbehandlung <errno.h>.....	312
A.4	Grenzwerte der Gleitkommadarstellung <float.h>.....	312
A.5	Wertebereich für Ganzzahltypen <limits.h>.....	313
A.6	Lokale Besonderheiten <locale.h>.....	313
A.7	Mathematische Funktionen <math.h>.....	313
A.8	Nichtlokale Sprünge <setjmp.h>.....	313
A.9	Signale <signal.h>.....	313
A.10	Variable Argumentlisten <stdarg.h>.....	314
A.11	Allgemein gültige Definitionen <stddef.h>.....	314
A.12	Ein-/Ausgabe <stdio.h>.....	315
A.13	Allgemeine Hilfsfunktionen <stdlib.h>.....	322
A.14	Speicherblockfunktionen und Funktionen für Zeichenketten <string.h>.....	326
A.15	Uhrzeit und Zeit <time.h>.....	328
Anhang B:	Syntaxzusammenfassung.....	329
Literatur.....		336
Stichwortverzeichnis.....		337
ASCII-Code Tabelle.....		345
Tabelle der Operatoren, Rangordnung der Operatoren.....		346