

INHALTSVERZEICHNIS

Vorwort	5
Vorwort zur deutschen Ausgabe	6
Einleitung	7
Inhaltsverzeichnis	9
Kapitel 1	
Einführung in die Programmiersprache C	17
Was bietet C?	17
Welche Voraussetzungen braucht der Leser?	19
Die grundlegende Charakteristik von C-Programmen	19
Funktionen	19
Die main()-Funktion	21
Geschweifte Klammern	21
Die C-Standardprogrammsammlung	23
Die Verwendung von Strichpunkten in C	25
Argumente für Funktionen	25
Ein Überblick	27
Wie man C-Programme schreiben sollte	28
Groß- oder Kleinbuchstaben?	28
Das Setzen von Klammern und Einrückungen	28
Klammern und Funktionen	29
Klammern innerhalb von Funktionen	29
Variable	31
Variablennamen	33
Schlüsselworte	33
printf() und Variable	34
Zusammenfassung	37
Kapitel 2	
Operatoren, Variablen und Schleifen	41
Operatoren	41
Arithmetische und relationale Operatoren	41
Die if-else Anweisung	42
if-else und geschweifte Klammern	44
Logische Operatoren	45

Inhaltsverzeichnis

Kapitel 2 (Fortsetzung)

Inkrementieren und Dekrementieren von Variablen	47
Programmschleifen	49
Schleifen mit while	49
#define	50
putchar(c)	51
Steueranweisungen für die printf()-Funktion	52
do-while-Schleifen	53
Schleifen mit for	53
Wie man Schleifen unterbricht und verläßt	58
Die continue-Anweisung	59

Anhang zu Kapitel 2

Einiges über Zahlensysteme	63
Das binäre Zahlensystem	63
Das oktale Zahlensystem	65
Das sedezimale Zahlensystem	67

Kapitel 3

Die Erstellung eigener Funktionen	73
Die Form von C-Funktionen	73
Die Typangabe	75
Der Funktionsname	75
Die Liste der Argumente	77
Die Vereinbarung des Argumenttyps	78
Die eigentliche Funktion	79
Die return-Anweisung und C	80
Die Rückgabe von Ergebnissen	81
Der Geltungsbereich von Variablen	82
Compiler und Linker	83
Extern definierte Variablen	84
Die automatische Zuweisung des Typs	89
Lokale Variable und Werte	91
Die 'statische' Variable	91
Registervariable	92
Funktionen und die zugehörigen Variablen	93
Matrizen - eine Ausnahme von der Zuordnungsregel	96
Wie man ein C-Programm entwirft	97

Kapitel 4

Wie man in C Zeiger anwendet	103
Die Definition von Zeigern	105
Wie man einen Zeiger initialisiert	106
Die l- und r-Werte	107
Zeiger und ihre l- und r-Werte	108
Die Bedeutung von Zeigern	111
Ein Anwendungsbeispiel für Zeiger	112
Noch ein Beispielprogramm	117
Das Inkrementieren und Dekrementieren von Zeigern	122

Kapitel 5

Ein- und Ausgabeoperationen in C	127
Die Verwendung der Standard-Ein-/Ausgabefunktion	127
Ein einfaches Programm	129
Die #include-Anweisung	130
Die #define-Anweisung	130
Die Null als Terminator	132
Die gleichzeitige Verwendung mehrerer Funktionen	134
Einige Feinheiten von Zeichenketten-Matrizen	135
Wie man numerische Daten in ein Programm einbringt	137
Zusatzanweisungen für printf()	140
scanf() - die universelle Eingabefunktion	143
Einige einfache Eingabefunktionen	146
Eine Eingabefunktion mit etwas mehr Pfiff	149
Die switch-Anweisung	153

Kapitel 6

Andere Datentypen	157
Die grundlegenden Datentypen	157
Erweiterungen der grundlegenden Datentypen	158
short	159
unsigned	159
long	159
Das Mischen von Datentypen	160
char und int	160
Umwandlungen und Zuordnungen	161
Binäre Operatoren und gemischte Datentypen	162
Fließkommaarithmetik und doppelte Genauigkeit	162

Inhaltsverzeichnis

Kapitel 6 (Fortsetzung)

Die Überführung von Funktionsargumenten	163
Die explizite Umwandlung des Datentyps	164
Die Anwendung der neuen Datentypen	164
Die Umwandlung von ASCII-Zeichen in Fließkommazahlen	164
Die Berechnung der Quadratwurzel einer Zahl	168
Die Verwendung vorzeichenloser Datenelemente	170
Die Verwendung des Datentyps long	172
Die Rechts-Links-Regel	173
Abkürzungen für Datentypen	175
Mehrdimensionale Matrizen	176

Kapitel 7

Strukturen und Datensätze	181
Wie man eine Struktur initialisiert	183
Ein Anwendungsbeispiel für Strukturen	186
Strukturen und Funktionen	187
Der Geltungsbereich von Strukturen	190
Das Ändern eines Strukturelements innerhalb einer Funktion	191
Die Übergabe einer ganzen Struktur an eine Funktion	192
Strukturen und Matrizen	195
Datensätze	203
Die Hierarchie der Operatoren	207

Kapitel 8

Das Arbeiten mit Diskettendateien	213
Maschinenorientierter oder hochsprachlicher Zugriff auf Diskettendateien?	213
Das Eröffnen einer Diskettendatei	214
Das Lesen einer Datei	220
argc und argv - Argumente für die Befehlszeile	221
Einfache Grafik	224
Der maschinenorientierte Zugriff auf Diskettendateien	229
open()	229
read()	231
write()	232
close()	233
Ein Beispielprogramm für maschinennahen Dateizugriff	234
Alternativen	236

Anhang zu Kapitel 8

Detailinformation zum Diskettendateienzugriff	239
Die FILE-Struktur	239
Wie man mit Dateien kommuniziert	242
stdin, stdout, stderr	243

Kapitel 9

Häufige Fehler und wie man sie findet	247
Häufige Fehler	247
Wenn der Strichpunkt falsch steht oder fehlt	247
Fehlende Klammern	248
Wertzweisungen kontra Vergleich	251
Kommentare	253
Argumente für Funktionen sind Kopien	254
Fehlende Argumentenvereinbarung in Funktionen	256
Fehlende Funktionsdefinitionen in main()	257
Funktionen geben Ganzzahlwerte an main() zurück	258
Nicht initialisierte Zeiger enthalten unsinnige Werte	260
Der 'gemeine' Fehler	260
Die Fehlersuche	262
Die verschiedenen Fehlerarten	262
Syntaxfehler	262
Programmfehler	263
Latente Fehler	263
Wie man Fehler feststellt und sie einkreist	264
Einige abschließende Betrachtungen	269

Anhang A

Der ASCII-Zeichensatz	273
-----------------------	-----

Anhang B

Syntax-Übersicht	277
Stichwortverzeichnis	297