

Inhalt

1	Betriebssysteme, Programmiersprachen	11
1.1	Betriebssysteme	12
1.2	Programmiersprachen	13
1.3	Empfehlungen für "Einsteiger", "Umsteiger", "Fortgeschrittene"	15
2	Hilfsmittel für die C-Programmierung	16
2.1	Compiler, Linker, Standard-Libraries	16
2.2	Editoren	17
2.3	Manuals, Lehrbücher	18
2.4	Benötigt man einen "Debugger"?	19
3	Grundlagen der Programmiersprache C	20
3.1	Wie lernt man eine Programmiersprache?	20
3.2	Vom Problem zum Programm	20
3.3	Das kleinste C-Programm "minimain.c"	21
3.4	C-Historie: Das "Hello, World"-Programm "hllworld.c"	22
3.5	Arithmetik und "for-Schleife": Programm "hptokw01.c"	25
3.6	Einige Grenzwerte der Implementation: Programm "limits.c"	30
3.7	Bedingte Anweisung und "Casting": Programm "reihe01.c"	31
3.8	Zeitmessung mit clock (): Programm "reihe02.c"	36
3.9	Standardfunktionen und "while-Schleife": Programm "valtab01.c"	38
3.10	Definition und Aufruf einer Funktion: Programm "valtab02.c"	42
3.11	Erster Kontakt mit Pointern: Programm "valtab03.c"	45
3.12	Formatgesteuerte Eingabe mit scanf: Programm "valtab04.c"	49
3.13	Stabilisierung der Eingabe: Programm "valtab05.c"	53
3.14	String-Konstanten als Funktionsargumente: Programm "valtab06.c"	56
3.15	Arrays und Strings: Programme "string1.c" und "syscall.c"	58
3.16	Zwischenbilanz	63
4	Arbeiten mit Libraries	72
4.1	Erzeugen einer Library	73
4.2	Einbinden einer persönlichen Library: Programm "valtab07.c"	76
4.3	Libraries mit Funktionen, die voneinander abhängig sind	78
4.4	Einbinden von Funktionen aus fremden Libraries	79
4.4.1	Ein mathematischer Parser für die "valtab"-Programme	81
4.4.2	Einbau von Parser-Library-Funktionen: Programm "valtab08.c"	83

5	Fortgeschrittene Programmiertechniken	88
5.1	Noch einmal: Strings und Pointer	88
5.2	Pointer-Arithmetik	97
5.3	Mehrdimensionale Arrays, Pointer-Arrays	101
5.4	Kommandozeilen-Argumente	105
6	File-Operationen und Speicherplatzverwaltung	107
6.1	Öffnen und Schließen eines Files, Zeichen lesen mit fgetc	107
6.2	Lesen und Schreiben mit fgetc und fputc, temporäre Files	110
6.3	Lesen mit fgets, formatiertes Lesen mit fscanf	114
6.4	Speicherplatz dynamisch allokalieren	118
6.5	Speicherplatzverwaltung	123
6.5.1	Speicherklassen auto und static innerhalb von Funktionen	123
6.5.2	Externe Variablen	125
6.5.3	Sichtbarkeit von Variablen	126
6.5.4	Was man sonst noch zu diesem Thema wissen sollte	127
7	Strukturen, verkettete Listen	129
7.1	Definition von Strukturen, Zugriff auf die Komponenten	129
7.2	Strukturen in Strukturen, Pointer auf Strukturen	133
7.3	Rekursive Strukturen, verkettete Listen	135
7.4	Sortieren mit verketteten Listen: Programm femfile4.c	140
7.5	"Unions" und "Enumerations"	147
7.5.1	"Unions"	147
7.5.2	Aufzählungstyp enum	148
8	Rekursionen, Baumstrukturen, Dateisysteme	150
8.1	Baumstrukturen	151
8.2	Die Dateisysteme unter UNIX und DOS	153
8.3	Eine UNIX-Besonderheit: Links auf Files und Directories	154
8.4	File-Informationen über die Files eines Directories	155
8.4.1	UNIX-Version (GNU-C)	156
8.4.2	DOS-Version ("16-Bit-Welt")	163
8.4.3	Version für das "DOS-Fenster der 32-Bit-Welt"	166
8.4.4	Pflege der privaten Library	167
8.5	Erster rekursiver Funktionsaufruf, Scannen eines Directory-Trees	167
8.6	Selektives Listen der Files eines Directory-Trees: Programm "lst.c"	169
8.7	Sortieren mit einem binären Baum: Programm "lstsort.c"	172
8.8	Zwischenbilanz	178
9	Grundlagen der Windows-Programmierung	182
9.1	Das Windows-Konzept	183
9.2	Botschaften (Nachrichten, "Messages")	184
9.3	Das kleinste Windows-Programm "miniwin.c"	185
9.4	Windows-Skelett-Programm "winskiel.c"	194

9.5	Text- und Graphik-Ausgabe, der "Device context"	196
9.5.1	Die Botschaft WM_PAINT, Programm "Hello, Winworld"	196
9.5.2	"Gefüllte Flächen", die GDI-Objekte "Pen" und "Brush"	200
9.5.3	Zeichnen mit MoveToEx und LineTo, Programm "rosette1.c"	205
9.6	Maus-Botschaften, Programm "mouse1.c"	208
9.7	Textausgabe mit TextOut, die Funktion GetSystemMetrics	211
9.8	Fehlersuche in Windows-Programmen	214
10	Ressourcen	218
10.1	Menü und Message-Box, Programm "menu1.c"	218
10.2	Stringtable und Dialog-Box, Programm "dialog1.c"	222
10.2.1	Modale und nicht-modale Dialoge	222
10.2.2	Definition einer Dialog-Box, Ressourcen-Datei "dialog1.rc"	223
10.2.3	Quelltext des Programms "dialog1.c"	225
10.3	Dialog-Funktion, Dialog-Box, Ressourcen-Editor	232
10.3.1	Ressourcen-Datei "hpkwin01.rc"	233
10.3.2	Programm "hpkwin01.c"	234
10.3.3	Erzeugen eines Dialog-Prototyps mit einem Ressourcen-Editor	239
10.4	Icon und Cursor	241
10.4.1	Erzeugen von Icons und Cursorformen	241
10.4.2	Ressourcen-Datei mit Icon, Cursor, Stringtable und Menü	243
10.4.3	Programm "cursor1.c"	244
11	C vertiefen oder C++ lernen?	252
11.1	Empfehlungen für die Vertiefung der C-Kenntnisse	252
11.2	Empfehlungen für das Lernen von C++	254
Anhang A	Ein Blick in die Speicherzellen	255
A1	Stellenwertsysteme	255
A2	Darstellung von Integer-Werten im Rechner	257
A3	Darstellung von float- und double-Werten im Rechner	
Anhang B	"Stack" und "Heap"	264
B1	Der Stack	264
B2	Der Heap	267
Literatur		268
Sachverzeichnis		269