

Auf einen Blick

1	Einige Grundbegriffe	21
2	Einführung in die Programmierung	35
3	Ausgewählte Sprachelemente von C	45
4	Arithmetik	83
5	Aussagenlogik	107
6	Elementare Datentypen und ihre Darstellung	129
7	Modularisierung	181
8	Zeiger und Adressen	223
9	Programmgrobstruktur	241
10	Die Standard C Library	253
11	Kombinatorik	273
12	Leistungsanalyse und Leistungsmessung	305
13	Sortieren	347
14	Datenstrukturen	393
15	Ausgewählte Datenstrukturen	437
16	Abstrakte Datentypen	493
17	Elemente der Graphentheorie	507
18	Zusammenfassung und Ergänzung	575
19	Einführung in C++	677
20	Objektorientierte Programmierung	717
21	Das Zusammenspiel von Objekten	775
22	Vererbung	805
23	Zusammenfassung und Überblick	879
24	Die C++-Standardbibliothek und Ergänzung	953
A	Aufgaben und Lösungen	1041

Inhalt

Vorwort	19
1 Einige Grundbegriffe	21
1.1 Algorithmus	24
1.2 Datenstruktur	28
1.3 Programm	30
1.4 Programmiersprachen	31
1.5 Aufgaben	33
2 Einführung in die Programmierung	35
2.1 Softwareentwicklung	35
2.2 Die Programmierumgebung	40
2.2.1 Der Editor	41
2.2.2 Der Compiler	42
2.2.3 Der Linker	43
2.2.4 Der Debugger	43
2.2.5 Der Profiler	43
3 Ausgewählte Sprachelemente von C	45
3.1 Programmrahmen	45
3.2 Zahlen	46
3.3 Variablen	46
3.4 Operatoren	48
3.4.1 Zuweisungsoperator	48
3.4.2 Arithmetische Operatoren	49
3.4.3 Typkonvertierungen	55
3.4.4 Vergleichsoperationen	55

3.5	Kontrollfluss	56
3.5.1	Bedingte Befehlsausführung	57
3.5.2	Wiederholte Befehlsausführung	59
3.5.3	Verschachtelung von Kontrollstrukturen	65
3.6	Elementare Ein- und Ausgabe	67
3.6.1	Bildschirmausgabe	67
3.6.2	Tastatureingabe	69
3.6.3	Kommentare und Layout	72
3.7	Beispiele	73
3.7.1	Das erste Programm	73
3.7.2	Das zweite Programm	75
3.7.3	Das dritte Programm	79
3.8	Aufgaben	81
4	Arithmetik	83
4.1	Folgen	85
4.2	Summen und Produkte	96
4.3	Aufgaben	100
5	Aussagenlogik	107
5.1	Aussagen	108
5.2	Aussagenlogische Operatoren	108
5.3	Boolesche Funktionen	116
5.4	Logische Operatoren in C	119
5.5	Beispiele	120
5.5.1	Kugelspiel	120
5.5.2	Schaltung	122
5.6	Aufgaben	126

6	Elementare Datentypen und ihre Darstellung	129
6.1	Zahlendarstellungen	130
6.1.1	Dualdarstellung	134
6.1.2	Oktalдарstellung	135
6.1.3	Hexadezimaldarstellung	136
6.2	Bits und Bytes	137
6.3	Skalare Datentypen in C	139
6.3.1	Ganze Zahlen	140
6.3.2	Gleitkommazahlen	144
6.4	Bitoperationen	146
6.5	Programmierbeispiele	150
6.5.1	Kartentrick	150
6.5.2	Zahlenraten	152
6.5.3	Addierwerk	154
6.6	Zeichen	156
6.7	Arrays	159
6.7.1	Eindimensionale Arrays	160
6.7.2	Mehrdimensionale Arrays	162
6.8	Zeichenketten	164
6.9	Programmierbeispiele	173
6.9.1	Buchstabenstatistik	173
6.9.2	Sudoku	175
6.10	Aufgaben	178
7	Modularisierung	181
7.1	Funktionen	181
7.2	Arrays als Funktionsparameter	186
7.3	Lokale und globale Variablen	190
7.4	Rekursion	192
7.5	Der Stack	198
7.6	Beispiele	200
7.6.1	Bruchrechnung	200
7.6.2	Das Damenproblem	202

7.6.3	Permutationen	210
7.6.4	Labyrinth	213
7.7	Aufgaben	218
8	Zeiger und Adressen	223
8.1	Zeigerarithmetik	230
8.2	Zeiger und Arrays	232
8.3	Funktionszeiger	235
8.4	Aufgaben	239
9	Programmstruktur	241
9.1	Der Präprozessor	241
9.1.1	Includes	242
9.1.2	Symbolische Konstanten	244
9.1.3	Makros	245
9.1.4	Bedingte Kompilierung	247
9.2	Ein kleines Projekt	249
10	Die Standard C Library	253
10.1	Mathematische Funktionen	254
10.2	Zeichenklassifizierung und -konvertierung	256
10.3	Stringoperationen	257
10.4	Ein- und Ausgabe	260
10.5	Variable Anzahl von Argumenten	263
10.6	Freispeicherverwaltung	265
10.7	Aufgaben	271

11	Kombinatorik	273
11.1	Kombinatorische Grundaufgaben	274
11.2	Permutationen mit Wiederholungen	274
11.3	Permutationen ohne Wiederholungen	275
11.3.1	Kombinationen ohne Wiederholungen	277
11.3.2	Kombinationen mit Wiederholungen	278
11.3.3	Zusammenfassung	280
11.4	Kombinatorische Algorithmen	283
11.4.1	Permutationen mit Wiederholungen	284
11.4.2	Kombinationen mit Wiederholungen	286
11.4.3	Kombinationen ohne Wiederholungen	288
11.4.4	Permutationen ohne Wiederholungen	290
11.5	Beispiele	293
11.5.1	Juwelenraub	293
11.5.2	Geldautomat	298
12	Leistungsanalyse und Leistungsmessung	305
12.1	Leistungsanalyse	308
12.2	Leistungsmessung	320
12.2.1	Überdeckungsanalyse	322
12.2.2	Performance-Analyse	323
12.3	Laufzeitklassen	324
13	Sortieren	347
13.1	Sortierverfahren	347
13.1.1	Bubblesort	349
13.1.2	Selectionsort	351
13.1.3	Insertionsort	353
13.1.4	Shellsort	356
13.1.5	Quicksort	359
13.1.6	Heapsort	370

13.2 Leistungsanalyse der Sortierverfahren	376
13.2.1 Bubblesort	376
13.2.2 Selectionsort	377
13.2.3 Insertionsort	378
13.2.4 Shellsort	379
13.2.5 Quicksort	380
13.2.6 Heapsort	381
13.3 Leistungsmessung der Sortierverfahren	383
13.4 Grenzen der Optimierung von Sortierverfahren	388

14 Datenstrukturen

14.1 Strukturdeklarationen	395
14.1.1 Variablendefinitionen	398
14.2 Zugriff auf Strukturen	400
14.2.1 Direktzugriff	401
14.2.2 Indirektzugriff	403
14.3 Datenstrukturen und Funktionen	405
14.4 Ein vollständiges Beispiel (Teil 1)	409
14.5 Dynamische Datenstrukturen	415
14.6 Ein vollständiges Beispiel (Teil 2)	421
14.7 Die Freispeicherverwaltung	432
14.8 Aufgaben	435

15 Ausgewählte Datenstrukturen

15.1 Listen	439
15.2 Bäume	448
15.2.1 Traversierung von Bäumen	451
15.2.2 Aufsteigend sortierte Bäume	461
15.3 Treaps	470
15.3.1 Heaps	471
15.3.2 Der Container als Treap	473

15.4 Hash-Tabellen	482
15.4.1 Speicherkomplexität	489
15.4.2 Laufzeitkomplexität	490

16 Abstrakte Datentypen

16.1 Der Stack als abstrakter Datentyp	495
16.2 Die Queue als abstrakter Datentyp	500

17 Elemente der Graphentheorie

17.1 Graphentheoretische Grundbegriffe	510
17.2 Die Adjazenzmatrix	511
17.3 Beispielgraph (Autobahnnetz)	512
17.4 Traversierung von Graphen	514
17.5 Wege in Graphen	516
17.6 Der Algorithmus von Warshall	518
17.7 Kantentabellen	522
17.8 Zusammenhang und Zusammenhangskomponenten	523
17.9 Gewichtete Graphen	530
17.10 Kürzeste Wege	532
17.11 Der Algorithmus von Floyd	533
17.12 Der Algorithmus von Dijkstra	539
17.13 Erzeugung von Kantentabellen	546
17.14 Der Algorithmus von Ford	548
17.15 Minimale Spannbäume	551
17.16 Der Algorithmus von Kruskal	552
17.17 Hamiltonsche Wege	557
17.18 Das Travelling-Salesman-Problem	562

18 Zusammenfassung und Ergänzung 575**19 Einführung in C++** 677

19.1 Schlüsselwörter	677
19.2 Kommentare	678
19.3 Datentypen, Datenstrukturen und Variablen	679
19.3.1 Automatische Typisierung von Aufzählungstypen	679
19.3.2 Automatische Typisierung von Strukturen	680
19.3.3 Vorwärtsverweise auf Strukturen	680
19.3.4 Der Datentyp bool	681
19.3.5 Verwendung von Konstanten	682
19.3.6 Definition von Variablen	683
19.3.7 Verwendung von Referenzen	684
19.3.8 Referenzen als Rückgabewerte	688
19.3.9 Referenzen außerhalb von Schnittstellen	689
19.4 Funktionen	690
19.4.1 Funktionsdeklarationen und Prototypen	691
19.4.2 Vorgegebene Werte in der Funktionsschnittstelle (Default-Werte)	692
19.4.3 Inline-Funktionen	694
19.4.4 Überladen von Funktionen	696
19.4.5 Parametersignatur von Funktionen	698
19.4.6 Zuordnung der Parametersignaturen und der passenden Funktion	699
19.4.7 Verwendung von C-Funktionen in C++-Programmen	700
19.5 Operatoren	701
19.5.1 Der Globalzugriff	702
19.5.2 Alle Operatoren in C++	703
19.5.3 Überladen von Operatoren	707
19.6 Auflösung von Namenskonflikten	711
19.6.1 Der Standardnamensraum std	715

20 Objektorientierte Programmierung 717

20.1 Ziele der Objektorientierung	717
20.2 Objektorientiertes Design	719
20.3 Klassen in C++	725
20.4 Aufbau von Klassen	725
20.4.1 Zugriffsschutz von Klassen	726
20.4.2 Datenmember	727
20.4.3 Funktionsmember	729
20.4.4 Verwendung des Zugriffsschutzes	731
20.4.5 Konstruktoren	735
20.4.6 Destruktoren	739
20.5 Instanziierung von Klassen	740
20.5.1 Automatische Variablen in C	740
20.5.2 Automatische Instanziierung in C++	741
20.5.3 Statische Variablen in C	741
20.5.4 Statische Instanziierung in C++	742
20.5.5 Dynamische Variablen in C	743
20.5.6 Dynamische Instanziierung in C++	743
20.5.7 Instanziierung von Arrays in C++	744
20.6 Operatoren auf Klassen	745
20.6.1 Friends	746
20.6.2 Operator als Methode der Klasse	747
20.7 Ein- und Ausgabe in C++	748
20.7.1 Überladen des <<-Operators	749
20.7.2 Tastatureingabe	750
20.7.3 Dateioperationen	752
20.8 Der this-Pointer	755
20.9 Beispiele	756
20.9.1 Menge	756
20.10 Aufgaben	771

21 Das Zusammenspiel von Objekten 775

21.1 Modellierung von Beziehungen	775
21.2 Komposition eigener Objekte	776

21.2.1	Komposition in C++	779
21.2.2	Implementierung der print-Methode für timestamp	780
21.2.3	Der Konstruktor von timestamp	781
21.2.4	Parametrierter Konstruktor einer komponierten Klasse	783
21.2.5	Konstruktionsoptionen der Klasse timestamp	785
21.3	Eine Klasse text	786
21.3.1	Der Copy-Konstruktor	788
21.3.2	Implementierung eines Copy-Konstruktors	790
21.3.3	Zuweisung von Objekten	791
21.3.4	Implementierung des Zuweisungsoperators	793
21.3.5	Erweiterung der Klasse text	794
21.3.6	Vorgehen für eigene Objekte	796
21.4	Übungen/Beispiel	797
21.4.1	Bingo	797
21.5	Aufgabe	803
 22 Vererbung		805
22.1	Darstellung der Vererbung	805
22.1.1	Mehrere abgeleitete Klassen	806
22.1.2	Wiederholte Vererbung	807
22.1.3	Mehrfachvererbung	807
22.2	Vererbung in C++	808
22.2.1	Ableiten einer Klasse	809
22.2.2	Gezieltes Aufrufen des Konstruktors der Basisklasse	810
22.2.3	Der geschützte Zugriffsbereich einer Klasse	812
22.2.4	Erweiterung abgeleiteter Klassen	813
22.2.5	Überschreiben von Funktionen der Basisklasse	814
22.2.6	Unterschiedliche Instanziierungen und deren Verwendung	817
22.2.7	Virtuelle Memberfunktionen	820
22.2.8	Verwendung des Schlüsselwortes virtual	821
22.2.9	Mehrfachvererbung	822
22.2.10	Zugriff auf die Methoden der Basisklassen	824
22.2.11	Statische Member	826
22.2.12	Rein virtuelle Funktionen	829
22.3	Beispiele	831
22.3.1	Würfelspiel	831
22.3.2	Partnervermittlung	855

23 Zusammenfassung und Überblick		879
23.1	Klassen und Instanzen	879
23.2	Member	881
23.2.1	Datenmember	881
23.2.2	Funktionsmember	882
23.2.3	Konstante Member	885
23.2.4	Statische Member	887
23.2.5	Operatoren	889
23.2.6	Zugriff auf Member	891
23.2.7	Zugriff von außen	891
23.2.8	Zugriff von innen	894
23.2.9	Der this-Pointer	898
23.2.10	Zugriff durch friends	899
23.3	Vererbung	900
23.3.1	Einfachvererbung	900
23.3.2	Mehrfachvererbung	905
23.3.3	Virtuelle Funktionen	911
23.3.4	Virtuelle Destruktoren	914
23.3.5	Rein virtuelle Funktionen	915
23.4	Zugriffsschutz und Vererbung	916
23.4.1	Geschützte Member	917
23.4.2	Zugriff auf die Basisklasse	917
23.4.3	Modifikation von Zugriffsrechten	921
23.5	Der Lebenszyklus von Objekten	922
23.5.1	Konstruktion von Objekten	925
23.5.2	Destruktion von Objekten	928
23.5.3	Kopieren von Objekten	929
23.5.4	Instanziierung von Objekten	934
23.5.5	Explizite und implizite Verwendung von Konstruktoren	937
23.5.6	Initialisierung eingelagerter Objekte	939
23.5.7	Initialisierung von Basisklassen	941
23.5.8	Instanziierungsregeln	943
23.6	Typüberprüfung und Typumwandlung	946
23.6.1	Dynamische Typüberprüfungen	946
23.7	Typumwandlung in C++	948

24 Die C++-Standardbibliothek und Ergänzung	953
24.1 Generische Klassen (Templates)	954
24.2 Ausnahmebehandlung (Exceptions)	962
24.3 Die C++-Standardbibliothek	973
24.4 Iteratoren	973
24.5 Strings (string)	976
24.5.1 Ein- und Ausgabe	977
24.5.2 Zugriff	978
24.5.3 Manipulation	981
24.5.4 Vergleich	986
24.5.5 Suchen	987
24.5.6 Speichermanagement	988
24.6 Dynamische Arrays (vector)	990
24.6.1 Die Beispielklasse klasse	990
24.6.2 Einbinden dynamischer Arrays	991
24.6.3 Konstruktion	991
24.6.4 Zugriff	992
24.6.5 Iteratoren	993
24.6.6 Manipulation	994
24.6.7 Speichermanagement	998
24.7 Listen (list)	998
24.7.1 Konstruktion	998
24.7.2 Zugriff	999
24.7.3 Iteratoren	1000
24.7.4 Manipulation	1002
24.7.5 Speichermanagement	1014
24.8 Stacks (stack)	1014
24.9 Warteschlangen (queue)	1017
24.10 Prioritätswarteschlangen (priority_queue)	1019
24.11 Geordnete Paare (pair)	1024
24.12 Mengen (set und multiset)	1025
24.12.1 Konstruktion	1026
24.12.2 Zugriff	1027
24.12.3 Manipulation	1029
24.13 Relationen (map und multimap)	1030
24.13.1 Konstruktion	1030

24.13.2 Zugriff	1031
24.13.3 Manipulation	1032
24.14 Algorithmen der Standardbibliothek	1032
24.14.1 Vererbung und virtuelle Funktionen in Containern	1037
A Aufgaben und Lösungen	1041
Kapitel 1	1042
Kapitel 3	1055
Kapitel 4	1069
Kapitel 5	1090
Kapitel 6	1103
Kapitel 7	1120
Kapitel 8	1144
Kapitel 10	1155
Kapitel 14	1162
Kapitel 20	1186
Kapitel 21	1203
Index	1209