
Contents

Preface	xix
1 Introduction to Computers, the Internet and the World Wide Web	1
1.1 Introduction	2
1.2 What Is a Computer?	4
1.3 Computer Organization	4
1.4 Evolution of Operating Systems	5
1.5 Personal, Distributed and Client/Server Computing	6
1.6 Machine Languages, Assembly Languages and High-Level Languages	6
1.7 FORTRAN, COBOL, Pascal and Ada	8
1.8 History of C	8
1.9 C Standard Library	9
1.10 C++	10
1.11 Java	10
1.12 BASIC, Visual Basic, Visual C++, C# and .NET	11
1.13 Key Software Trend: Object Technology	12
1.14 Basics of a Typical C Program Development Environment	13
1.15 Hardware Trends	15
1.16 History of the Internet	16
1.17 History of the World Wide Web	17
1.18 General Notes About C and this Book	18
2 Introduction to C Programming	26
2.1 Introduction	27
2.2 A Simple C Program: Printing a Line of Text	27
2.3 Another Simple C Program: Adding Two Integers	31

2.4	Memory Concepts	36
2.5	Arithmetic in C	37
2.6	Decision Making: Equality and Relational Operators	40
3	Structured Program Development in C	56
3.1	Introduction	57
3.2	Algorithms	57
3.3	Pseudocode	58
3.4	Control Structures	58
3.5	The <code>if</code> Selection Statement	60
3.6	The <code>if...else</code> Selection Statement	62
3.7	The <code>while</code> Repetition Statement	65
3.8	Formulating Algorithms: Case Study 1 (Counter-Controlled Repetition)	67
3.9	Formulating Algorithms with Top-Down, Stepwise Refinement: Case Study 2 (Sentinel-Controlled Repetition)	69
3.10	Formulating Algorithms with Top-Down, Stepwise Refinement: Case Study 3 (Nested Control Structures)	75
3.11	Assignment Operators	79
3.12	Increment and Decrement Operators	80
4	C Program Control	100
4.1	Introduction	101
4.2	Repetition Essentials	101
4.3	Counter-Controlled Repetition	102
4.4	<code>for</code> Repetition Statement	104
4.5	<code>for</code> Statement: Notes and Observations	106
4.6	Examples Using the <code>for</code> Statement	107
4.7	<code>switch</code> Multiple-Selection Statement	111
4.8	<code>do...while</code> Repetition Statement	117
4.9	<code>break</code> and <code>continue</code> Statements	119
4.10	Logical Operators	120
4.11	Confusing Equality (<code>==</code>) and Assignment (<code>=</code>) Operators	123
4.12	Structured Programming Summary	125
5	C Functions	143
5.1	Introduction	144
5.2	Program Modules in C	144
5.3	Math Library Functions	145
5.4	Functions	147
5.5	Function Definitions	148
5.6	Function Prototypes	152
5.7	Headers	154
5.8	Calling Functions: Call by Value and Call by Reference	155
5.9	Random Number Generation	156
5.10	Example: A Game of Chance	161
5.11	Storage Classes	164

5.12	Scope Rules	167
5.13	Recursion	170
5.14	Example Using Recursion: Fibonacci Series	173
5.15	Recursion vs. Iteration	177
6	C Arrays	199
6.1	Introduction	200
6.2	Arrays	200
6.3	Defining Arrays	202
6.4	Array Examples	202
6.5	Passing Arrays to Functions	216
6.6	Sorting Arrays	220
6.7	Case Study: Computing Mean, Median and Mode Using Arrays	223
6.8	Searching Arrays	228
6.9	Multiple-Subscripted Arrays	233
7	C Pointers	258
7.1	Introduction	259
7.2	Pointer Variable Definitions and Initialization	259
7.3	Pointer Operators	260
7.4	Calling Functions by Reference	263
7.5	Using the <code>const</code> Qualifier with Pointers	267
7.6	Bubble Sort Using Call-by-Reference	273
7.7	<code>sizeof</code> Operator	277
7.8	Pointer Expressions and Pointer Arithmetic	279
7.9	Relationship between Pointers and Arrays	281
7.10	Arrays of Pointers	286
7.11	Case Study: Card Shuffling and Dealing Simulation	286
7.12	Pointers to Functions	292
8	C Characters and Strings	317
8.1	Introduction	318
8.2	Fundamentals of Strings and Characters	318
8.3	Character Handling Library	320
8.4	String Conversion Functions	325
8.5	Standard Input/Output Library Functions	330
8.6	String Manipulation Functions of the String Handling Library	335
8.7	Comparison Functions of the String Handling Library	337
8.8	Search Functions of the String Handling Library	339
8.9	Memory Functions of the String Handling Library	346
8.10	Other Functions of the String Handling Library	350
9	C Formatted Input/Output	364
9.1	Introduction	365
9.2	Streams	365
9.3	Formatting Output with <code>printf</code>	365

9.4	Printing Integers	366
9.5	Printing Floating-Point Numbers	368
9.6	Printing Strings and Characters	369
9.7	Other Conversion Specifiers	370
9.8	Printing with Field Widths and Precision	372
9.9	Using Flags in the <code>printf</code> Format Control String	374
9.10	Printing Literals and Escape Sequences	377
9.11	Formatting Input with <code>scanf</code>	378
10 C Structures, Unions, Bit Manipulations and Enumerations		393
10.1	Introduction	394
10.2	Structure Definitions	394
10.3	Initializing Structures	397
10.4	Accessing Members of Structures	397
10.5	Using Structures with Functions	399
10.6	<code>typedef</code>	399
10.7	Example: High-Performance Card Shuffling and Dealing Simulation	400
10.8	Unions	403
10.9	Bitwise Operators	405
10.10	Bit Fields	414
10.11	Enumeration Constants	417
11 C File Processing		429
11.1	Introduction	430
11.2	Data Hierarchy	430
11.3	Files and Streams	432
11.4	Creating a Sequential-Access File	433
11.5	Reading Data from a Sequential-Access File	438
11.6	Random-Access Files	443
11.7	Creating a Random-Access File	444
11.8	Writing Data Randomly to a Random-Access File	446
11.9	Reading Data from a Random-Access File	449
11.10	Case Study: Transaction-Processing Program	450
12 C Data Structures		466
12.1	Introduction	467
12.2	Self-Referential Structures	468
12.3	Dynamic Memory Allocation	468
12.4	Linked Lists	470
12.5	Stacks	478
12.6	Queues	484
12.7	Trees	490

13	C Preprocessor	520
13.1	Introduction	521
13.2	<code>#include</code> Preprocessor Directive	521
13.3	<code>#define</code> Preprocessor Directive: Symbolic Constants	522
13.4	<code>#define</code> Preprocessor Directive: Macros	522
13.5	Conditional Compilation	524
13.6	<code>#error</code> and <code>#pragma</code> Preprocessor Directives	525
13.7	<code>#</code> and <code>##</code> Operators	525
13.8	Line Numbers	526
13.9	Predefined Symbolic Constants	526
13.10	Assertions	527
14	Other C Topics	532
14.1	Introduction	533
14.2	Redirecting Input/Output on UNIX and Windows Systems	533
14.3	Variable-Length Argument Lists	534
14.4	Using Command-Line Arguments	536
14.5	Notes on Compiling Multiple-Source-File Programs	538
14.6	Program Termination with <code>exit</code> and <code>atexit</code>	540
14.7	<code>volatile</code> Type Qualifier	541
14.8	Suffixes for Integer and Floating-Point Constants	541
14.9	More on Files	542
14.10	Signal Handling	544
14.11	Dynamic Memory Allocation: Functions <code>calloc</code> and <code>realloc</code>	546
14.12	Unconditional Branching with <code>goto</code>	547
15	C++ as a “Better C”	554
15.1	Introduction	555
15.2	C++	555
15.3	A Simple Program: Adding Two Integers	556
15.4	C++ Standard Library	558
15.5	Header Files	559
15.6	Inline Functions	561
15.7	References and Reference Parameters	563
15.8	Default Arguments and Empty Parameter Lists	567
15.9	Unary Scope Resolution Operator	569
15.10	Function Overloading	572
15.11	Function Templates	573
16	C++ Classes and Data Abstraction	581
16.1	Introduction	582
16.2	Implementing a Time Abstract Data Type with a Class	583
16.3	Class Scope and Accessing Class Members	589
16.4	Separating Interface from Implementation	591
16.5	Controlling Access to Members	595
16.6	Access Functions and Utility Functions	598

16.7	Initializing Class Objects: Constructors	601
16.8	Using Default Arguments with Constructors	601
16.9	Using Destructors	605
16.10	When Constructors and Destructors Are Called	606
16.11	Using Data Members and Member Functions	609
16.12	A Subtle Trap: Returning a Reference to a <code>private</code> Data Member	614
16.13	Assignment by Default Memberwise Copy	616
16.14	Software Reusability	618
17	C++ Classes: Part II	627
17.1	Introduction	628
17.2	<code>const</code> (Constant) Objects and <code>const</code> Member Functions	628
17.3	Composition: Objects as Members of Classes	636
17.4	<code>friend</code> Functions and <code>friend</code> Classes	642
17.5	Using the <code>this</code> Pointer	645
17.6	Dynamic Memory Allocation with Operators <code>new</code> and <code>delete</code>	650
17.7	<code>static</code> Class Members	651
17.8	Data Abstraction and Information Hiding	657
17.8.1	Example: Array Abstract Data Type	658
17.8.2	Example: String Abstract Data Type	659
17.8.3	Example: Queue Abstract Data Type	659
17.9	Container Classes and Iterators	660
18	C++ Operator Overloading	667
18.1	Introduction	668
18.2	Fundamentals of Operator Overloading	669
18.3	Restrictions on Operator Overloading	670
18.4	Operator Functions as Class Members vs. as <code>friend</code> Functions	672
18.5	Overloading Stream-Insertion and Stream-Extraction Operators	673
18.6	Overloading Unary Operators	676
18.7	Overloading Binary Operators	676
18.8	Case Study: An Array Class	677
18.9	Converting between Types	689
18.10	Overloading <code>++</code> and <code>--</code>	690
19	C++ Inheritance	699
19.1	Introduction	700
19.2	Inheritance: Base Classes and Derived Classes	702
19.3	Protected Members	704
19.4	Casting Base-Class Pointers to Derived-Class Pointers	704
19.5	Using Member Functions	710
19.6	Overriding Base-Class Members in a Derived Class	710
19.7	Public, Protected and Private Inheritance	714
19.8	Direct Base Classes and Indirect Base Classes	715
19.9	Using Constructors and Destructors in Derived Classes	716
19.10	Implicit Derived-Class Object to Base-Class Object Conversion	720

19.11	Software Engineering with Inheritance	721
19.12	Composition vs. Inheritance	722
19.13	<i>Uses A</i> and <i>Knows A</i> Relationships	722
19.14	Case Study: Point, Circle, Cylinder	723
20	C++ Virtual Functions and Polymorphism	737
20.1	Introduction	738
20.2	Type Fields and <code>switch</code> Statements	738
20.3	<code>virtual</code> Functions	739
20.4	Abstract Base Classes and Concrete Classes	740
20.5	Polymorphism	740
20.6	New Classes and Dynamic Binding	742
20.7	Virtual Destructors	743
20.8	Case Study: Inheriting Interface and Implementation	743
20.9	Polymorphism, <code>virtual</code> Functions and Dynamic Binding “Under the Hood”	752
21	C++ Stream Input/Output	760
21.1	Introduction	762
21.2	Streams	762
	21.2.1 <code>iostream</code> Library Header Files	763
	21.2.2 Stream Input/Output Classes and Objects	763
21.3	Stream Output	765
	21.3.1 Stream-Insertion Operator	765
	21.3.2 Cascading Stream-Insertion/Extraction Operators	767
	21.3.3 Output of <code>char *</code> Variables	768
	21.3.4 Character Output with Member Function <code>put</code> ; Cascading <code>puts</code>	769
21.4	Stream Input	769
	21.4.1 Stream-Extraction Operator	769
	21.4.2 <code>get</code> and <code>getline</code> Member Functions	772
	21.4.3 <code>istream</code> Member Functions <code>peek</code> , <code>putback</code> and <code>ignore</code>	775
	21.4.4 Type-Safe I/O	775
21.5	Unformatted I/O with <code>read</code> , <code>gcount</code> and <code>write</code>	775
21.6	Stream Manipulators	776
	21.6.1 Integral Stream Base: <code>dec</code> , <code>oct</code> , <code>hex</code> and <code>setbase</code>	776
	21.6.2 Floating-Point Precision (<code>precision</code> , <code>setprecision</code>)	777
	21.6.3 Field Width (<code>setw</code> , <code>width</code>)	779
	21.6.4 User-Defined Manipulators	780
21.7	Stream Format States	781
	21.7.1 Format State Flags	781
	21.7.2 Trailing Zeros and Decimal Points (<code>ios::showpoint</code>)	783
	21.7.3 Justification (<code>ios::left</code> , <code>ios::right</code> , <code>ios::internal</code>)	783
	21.7.4 Padding (<code>fill</code> , <code>setfill</code>)	786
	21.7.5 Integral Stream Base (<code>ios::dec</code> , <code>ios::oct</code> , <code>ios::hex</code> , <code>ios::showbase</code>)	787
	21.7.6 Floating-Point Numbers; Scientific Notation (<code>ios::scientific</code> , <code>ios::fixed</code>)	788

21.7.7	Uppercase/Lowercase Control (<code>ios::uppercase</code>)	789
21.7.8	Setting and Resetting the Format Flags (<code>flags</code> , <code>setiosflags</code> , <code>resetiosflags</code>)	790
21.8	Stream Error States	791
21.9	Tying an Output Stream to an Input Stream	793
22	C++ Templates	806
22.1	Introduction	807
22.2	Class Templates	807
22.3	Class Templates and Nontype Parameters	813
22.4	Templates and Inheritance	814
22.5	Templates and <code>friends</code>	814
22.6	Templates and <code>static</code> Members	815
23	C++ Exception Handling	819
23.1	Introduction	820
23.2	When Exception Handling Should Be Used	822
23.3	Other Error-Handling Techniques	823
23.4	Basics of C++ Exception Handling: <code>try</code> , <code>throw</code> , <code>catch</code>	824
23.5	A Simple Exception-Handling Example: Divide by Zero	824
23.6	Throwing an Exception	827
23.7	Catching an Exception	828
23.8	Rethrowing an Exception	831
23.9	Exception Specifications	833
23.10	Processing Unexpected Exceptions	833
23.11	Stack Unwinding	834
23.12	Constructors, Destructors and Exception Handling	835
23.13	Exceptions and Inheritance	836
23.14	Processing <code>new</code> Failures	836
23.15	Class <code>auto_ptr</code> and Dynamic Memory Allocation	840
23.16	Standard Library Exception Hierarchy	842
24	Introduction to Java Applications and Applets	852
24.1	Introduction	853
24.2	Basics of a Typical Java Environment	854
24.3	General Notes about Java and This Book	857
24.4	A Simple Program: Printing a Line of Text	858
24.5	Another Java Application: Adding Integers	866
24.6	Sample Applets from the Java 2 Software Development Kit	871
24.6.1	The <code>TicTacToe</code> Applet	872
24.6.2	The <code>DrawTest</code> Applet	875
24.6.3	The <code>Java2D</code> Applet	876
24.7	A Simple Java Applet: Drawing a String	877
24.8	Two More Simple Applets: Drawing Strings and Lines	884
24.9	Another Java Applet: Adding Integers	886

25	Beyond C and C++: Operators, Methods and Arrays in Java	905
25.1	Introduction	906
25.2	Primitive Data Types and Keywords	906
25.3	Logical Operators	908
25.4	Method Definitions	913
25.5	Java API Packages	918
25.6	Random Number Generation	922
25.7	Example: A Game of Chance	926
25.8	Methods of Class <code>JApplet</code>	934
25.9	Declaring and Allocating Arrays	935
25.10	Examples Using Arrays	936
25.11	References and Reference Parameters	946
25.12	Multiple-Subscripted Arrays	946
26	Java Object-Based Programming	962
26.1	Introduction	963
26.2	Implementing a Time Abstract Data Type with a Class	964
26.3	Class Scope	972
26.4	Creating Packages	972
26.5	Initializing Class Objects: Constructors	976
26.6	Using <i>Set</i> and <i>Get</i> Methods	977
26.7	Using the <code>this</code> Reference	984
26.8	Finalizers	986
26.9	Static Class Members	986
27	Java Object-Oriented Programming	999
27.1	Introduction	1000
27.2	Superclasses and Subclasses	1002
27.3	<code>protected</code> Members	1004
27.4	Relationship between Superclass Objects and Subclass Objects	1004
27.5	Implicit Subclass-Object-to-Superclass-Object Conversion	1012
27.6	Software Engineering with Inheritance	1012
27.7	Composition vs. Inheritance	1013
27.8	Introduction to Polymorphism	1014
27.9	Type Fields and <code>switch</code> Statements	1014
27.10	Dynamic Method Binding	1014
27.11	<code>final</code> Methods and Classes	1015
27.12	Abstract Superclasses and Concrete Classes	1015
27.13	Polymorphism Example	1016
27.14	New Classes and Dynamic Binding	1018
27.15	Case Study: Inheriting Interface and Implementation	1019
27.16	Case Study: Creating and Using Interfaces	1025
27.17	Inner Class Definitions	1030
27.18	Notes on Inner Class Definitions	1041
27.19	Type-Wrapper Classes for Primitive Types	1041

28	Java Graphics and Java2D	1051
28.1	Introduction	1052
28.2	Graphics Contexts and Graphics Objects	1054
28.3	Color Control	1055
28.4	Font Control	1062
28.5	Drawing Lines, Rectangles and Ovals	1068
28.6	Drawing Arcs	1071
28.7	Drawing Polygons and Polylines	1074
28.8	The Java2D API	1077
28.9	Java2D Shapes	1077
29	Java Graphical User Interface Components	1092
29.1	Introduction	1093
29.2	Swing Overview	1095
29.3	JLabel	1097
29.4	Event Handling Model	1100
29.5	JTextField and JPasswordField	1102
29.5.1	How Event Handling Works	1107
29.6	JTextArea	1108
29.7	JButton	1111
29.8	JCheckBox	1114
29.9	JComboBox	1117
29.10	Mouse Event Handling	1120
29.11	Layout Managers	1124
29.11.1	FlowLayout	1125
29.11.2	BorderLayout	1128
29.11.3	GridLayout	1131
29.12	Panels	1133
29.13	Creating a Self-Contained Subclass of JPanel	1135
29.14	Windows	1141
29.15	Using Menus with Frames	1142
30	Java Multimedia: Images, Animation and Audio	1163
30.1	Introduction	1164
30.2	Loading, Displaying and Scaling Images	1165
30.3	Loading and Playing Audio Clips	1168
30.4	Animating a Series of Images	1171
30.5	Animation Issues	1175
30.6	Customizing Applets via the HTML param Tag	1177
30.7	Image Maps	1182
30.8	Internet and World Wide Web Resources	1184
A	Internet and Web Resources	1190
A.1	C/C++ Resources	1190
A.2	C++ Tutorials	1191
A.3	C/C++ FAQs	1191

A.4	comp.lang.c++	1191
A.5	C/C++ Compilers	1192
A.6	Java Resources	1193
A.7	Java Products	1194
A.8	Java FAQs	1194
A.9	Java Tutorials	1194
A.10	Java Magazines	1195
A.11	Java Applets	1195
A.12	Multimedia	1196
A.13	Java Newsgroups	1196
B	C99 Internet and Web Resources	1197
B.1	C99 Resources	1197
C	Operator Precedence Charts	1199
D	ASCII Character Set	1204
E	Number Systems	1205
E.1	Introduction	1206
E.2	Abbreviating Binary Numbers as Octal Numbers and Hexadecimal Numbers	1209
E.3	Converting Octal Numbers and Hexadecimal Numbers to Binary Numbers	1210
E.4	Converting from Binary, Octal or Hexadecimal to Decimal	1210
E.5	Converting from Decimal to Binary, Octal or Hexadecimal	1211
E.6	Negative Binary Numbers: Two's Complement Notation	1213
F	C Standard Library Resources	1218
F.1	C Standard Library Resources	1218
	Index	1220