

# Contents

---

## I INTRODUCTION

---

<b>1 Computers and Systems</b>	2
1.1 The Physical Computer	2
1.2 The Operating System	10
1.3 Languages	12
1.4 What You Should Remember	16
1.5 Exercises	17
<b>2 Programs and Programming</b>	20
2.1 What Is a Program?	20
2.2 Problem Specification and Analysis	23
2.3 The Development Environment	27
2.4 Program Construction	29
2.5 Program Execution and Testing	34
2.6 What You Should Remember	37
2.7 Exercises	38
<b>3 Fundamental Concepts</b>	42
3.1 Parts of a Program	42
3.2 The Simplest Program	44
3.3 Variables, Input, Output, and Sequencing	46
3.4 Simple Calculations	52
3.5 The Flow of Control	57

3.6	Asking Questions: Conditional Statements	60
3.7	Loops and Repetition	70
3.8	An Application	75
3.9	What You Should Remember	80
3.10	Exercises	84

## II COMPUTATION

---

<b>4</b>	<b>Objects, Types, and Expressions</b>	94
4.1	Variables, Constants, and Literals	94
4.2	Expressions and Parse Trees	101
4.3	Arithmetic, Assignment, and Combination Operators	107
4.4	Increment and Decrement Operators	108
4.5	Relational Operators	114
4.6	Logic Operators	115
4.7	An Example: A Voltage Ramp	120
4.8	Case Study: Using a Parse Tree to Debug	123
4.9	What You Should Remember	128
4.10	Exercises	131
<b>5</b>	<b>Using Functions and Libraries</b>	138
5.1	Libraries	139
5.2	Using Libraries	141
5.3	Function Types	147
5.4	User-Defined Functions	148
5.5	Math Library Application: Roots of a Quadratic Equation	155
5.6	What You Should Remember	159
5.7	Exercises	162
<b>6</b>	<b>More Repetition and Decision</b>	170
6.1	New Loops	171
6.2	Applications of Loops	179
6.3	The <code>switch</code> Statement	196
6.4	Counted Loop Application: Integration by Simpson's Rule	202
6.5	Sentinel Loop Application: Interpolation (Optional Topic)	209
6.6	What You Should Remember	216
6.7	Exercises	220

### III BASIC DATA TYPES

---

<b>7 Using Numeric Types</b>	228
7.1 Integer Types	228
7.2 Floating-Point Types in C	231
7.3 Reading and Writing Numbers	234
7.4 Integer Operations	240
7.5 Mixing Types in Computations (Advanced Topic)	246
7.6 Pseudo-Random Numbers	260
7.7 Application: A Guessing Game	264
7.8 What You Should Remember	269
7.9 Exercises	274
<b>8 The Trouble with Numbers</b>	282
8.1 Floating-Point Comparisons	282
8.2 Calculation Errors (Advanced Topic)	288
8.3 Optional Application: Finding the Real Roots of an Equation	294
8.4 What You Should Remember	302
8.5 Exercises	304
<b>9 Program Design</b>	312
9.1 Modular Programs	312
9.2 Communication Between Functions	316
9.3 Declaration-Call Correspondence	321
9.4 Data Modularity	331
9.5 Function Call Graphs	339
9.6 Program Design and Construction	340
9.7 What You Should Remember	353
9.8 Exercises	356
<b>10 An Introduction to Arrays</b>	366
10.1 Arrays	366
10.2 Using Arrays	374
10.3 Parallel Arrays	378
10.4 Array Arguments and Parameters	382
10.5 What You Should Remember	398
10.6 Exercises	400

<b>11</b>	<b>Character Data and Enumerations</b>	410
11.1	Representation of Characters	410
11.2	Input and Output with Characters	413
11.3	Operations on Characters (Advanced Topic)	419
11.4	Character Application: An Improved Processing Loop	423
11.5	Enumerated Types	425
11.6	What You Should Remember	432
11.7	Exercises	435
<b>12</b>	<b>An Introduction to Pointers</b>	442
12.1	A First Look at Pointers	442
12.2	Call by Address	448
12.3	Application: The Bisection Method (Advanced Topic)	459
12.4	What You Should Remember	471
12.5	Exercises	474
<b>IV</b>	<b>STRUCTURED DATA TYPES</b>	
<b>13</b>	<b>Strings</b>	486
13.1	String Representation	486
13.2	String I/O	493
13.3	The String Library	501
13.4	Arrays of Strings	508
13.5	String Processing Applications (Optional Topic)	517
13.6	Application: A Gas Pressure Table	530
13.7	What You Should Remember	535
13.8	Exercises	539
<b>14</b>	<b>Structures Types</b>	552
14.1	Declarations	552
14.2	Operations on Structures	555
14.3	Application: Points in a Rectangle	565
14.4	Application: The Monte Carlo Method (Optional Topic)	578
14.5	What You Should Remember	585
14.6	Exercises	588
<b>15</b>	<b>Streams and Files</b>	598
15.1	Streams and Buffers	599
15.2	Programmer-Defined Streams	604

15.3	Stream Output	607
15.4	Stream Input	612
15.5	Errors and Exceptions	617
15.6	File Application: Random Selection Without Replacement	628
15.7	Application: Measuring Torque (Optional Topic)	638
15.8	What You Should Remember	643
15.9	Exercises	647
<b>16</b>	<b>Simple Array Algorithms</b>	<b>656</b>
16.1	Searching an Array Data Structure	656
16.2	Application: Screening out Faulty Data (Optional Topic)	663
16.3	Sorting by Selection	677
16.4	What You Should Remember	687
16.5	Exercises	691
<b>17</b>	<b>Two-Dimensional Arrays</b>	<b>698</b>
17.1	Nested Loops: Printing a Table	698
17.2	Introduction to Two-Dimensional Arrays	701
17.3	Application: Transformation of 2D Point Coordinates	711
17.4	Application: Image Processing	719
17.5	What You Should Remember	728
17.6	Exercises	730
<b>18</b>	<b>Calculating with Bits</b>	<b>742</b>
18.1	Unsigned Numbers and Hexadecimal Notation	743
18.2	Bitwise Operators	748
18.3	Application: Simple Encryption and Decryption	755
18.4	Bitfield Types	759
18.5	What You Should Remember	771
18.6	Exercises	774

## **V ADVANCED TECHNIQUES**

---

<b>19</b>	<b>Dynamic Arrays</b>	<b>786</b>
19.1	Dynamic Memory Allocation	786
19.2	Using Dynamic Arrays: A Simulation	799
19.3	Dynamic Matrix: An Array of Pointers (Advanced Topic)	809
19.4	What You Should Remember	823
19.5	Exercises	827

**Contents**

<b>20 Working with Pointers</b>	834
20.1 Pointers—Old and New Ideas	834
20.2 Application: A Menu of Pointers to Functions (Advanced Topic)	842
20.3 Using Pointers with Arrays	846
20.4 Insertion Sort	852
20.5 What You Should Remember	859
20.6 Exercises	864
<b>21 Recursion</b>	874
21.1 Storage Classes	874
21.2 The Run-Time Stack (Advanced Topic)	878
21.3 Iteration and Recursion	881
21.4 A Simple Example of Recursion	883
21.5 A More Complex Example: Binary Search	886
21.6 Quicksort	896
21.7 What You Should Remember	908
21.8 Exercises	911
<b>22 Making Programs General</b>	918
22.1 Command-Line Arguments	918
22.2 Functions as Parameters	927
22.3 What You Should Remember	934
22.4 Exercises	937
<b>23 Modular Organization</b>	944
23.1 Constructing a Modular Program	944
23.2 Modular Application: Finding the Roots of an Equation	954
23.3 What You Should Remember	977
23.4 Exercises	979

**APPENDIXES**


---

<b>A The ASCII Code</b>	984
<b>B The Precedence of Operators in C</b>	986
<b>C Keywords</b>	988

## **D Advanced Aspects of C Operators**

D.1 Assignment Combination Operators	990
D.2 More on Lazy Evaluation and Skipping	992
D.3 The Conditional Operator	996
D.4 The Comma Operator	998
D.5 Summary	998

## **E Number Representation and Conversion**

E.1 Number Systems and Number Representation	1000
E.2 Signed and Unsigned Integers	1001
E.3 Representation of Real Numbers	1002
E.4 Base Conversion	1004
E.5 Self-Test Exercises	1006

## **F The tools Library**

F.1 Using the <code>tools</code> Library in a Program	1008
F.2 Portability Command	1013
F.3 Functions Declared in <code>tools.h</code> and Defined in <code>tools.c</code>	1014
F.4 Process and Stream Management	1015
F.5 Time and Date Functions	1020
F.6 Numeric Functions	1022
F.7 Strings and Menus	1023

## **G The Standard C Environment**

G.1 Built-in Facilities	1026
G.2 Standard Files of Constants	1027
G.3 The Standard Libraries and <code>main()</code>	1028
G.4 Libraries Not Explored	1039

## **H Glossary**

1042

## **I Answers to Self-Test Exercises**

1058

## **Index**

1096