Contents

Preface **1**
About This Book . . . 1
The Success of C . . . 2
Standards . . . 4
Hosted and Free-Standing Environments . . . 5
Typographical conventions . . . 6
Order of topics . . . 6
Example programs . . . 7
Deference to Higher Authority . . . 7
Address for the Standard . . . 7

1 An Introduction to C **9**
1.1 The form of a C program . . . 9
1.2 Functions . . . 10
1.3 A description of Example 1.1 . . . 12
1.3.1 What was in it . . . 12
1.3.2 Layout and comment . . . 12
1.3.3 Preprocessor statements . . . 13
1.3.3.1 Define statements . . . 14
1.3.3.2 Summary . . . 14
1.3.4 Function declaration and definition . . . 15
1.3.4.1 Declaration . . . 15
1.3.4.2 Definition . . . 15
1.3.4.3 Summary . . . 16
1.3.5 Strings . . . 17
1.3.6 The main function . . . 18
1.3.7 Declarations . . . 18
1.3.8 Assignment statement . . . 19
1.3.9 The while statement . . . 19
1.3.10 The return statement . . . 20
1.3.10.1 Summary . . . 21

1.3.11 Progress so far . . . 21
1.4 Some more programs . . . 21
1.4.1 A program to find prime numbers . . . 22
1.4.2 The division operators . . . 24
1.4.3 An example performing input . . . 24
1.4.4 Simple arrays . . . 25
1.4.5 Summary . . . 27
1.5 Terminology . . . 27
1.6 Summary . . . 28
1.7 Exercises . . . 28

2 Variables and Arithmetic 31
2.1 Some fundamentals . . . 31
2.2 The alphabet of C . . . 31
2.2.1 Basic Alphabet . . . 32
2.2.2 Trigraphs . . . 33
2.2.3 Multibyte Characters . . . 34
2.2.4 Summary . . . 35
2.3 The Textual Structure of Programs . . . 36
2.3.1 Program Layout . . . 36
2.3.2 Comment . . . 37
2.3.3 Translation phases . . . 38
2.4 Keywords and identifiers . . . 38
2.4.1 Keywords . . . 38
2.4.2 Identifiers . . . 39
2.5 Declaration of variables . . . 40
2.6 Real types . . . 41
2.6.1 Summary of real arithmatic . . . 44
2.6.2 Printing real numbers . . . 44
2.7 Integral types . . . 45
2.7.1 Plain integers . . . 45
2.7.2 Character variables . . . 46
2.7.3 More complicated types . . . 49
2.7.4 Summary of integral types . . . 51
2.7.5 Printing the integral types . . . 51
2.8 Expressions and arithmetic . . . 52
2.8.1 Conversions . . . 53
2.8.1.1 Integral promotions . . . 54
2.8.1.2 Signed and unsigned integers . . . 54
2.8.1.3 Floating and integral . . . 55
2.8.1.4 The usual arithmetic conversions . . . 55

2.8.1.5 Wide characters . . . 57
2.8.1.6 Casts . . . 60
2.8.2 Operators . . . 62
2.8.2.1 The multiplicative operators . . . 62
2.8.2.2 Additive operators . . . 62
2.8.2.3 The bitwise operators . . . 63
2.8.2.4 The assignment operators . . . 65
2.8.2.5 Increment and decrement operators . . . 66
2.8.3 Precedence and grouping . . . 69
2.8.4 Parentheses . . . 72
2.8.5 Side Effects . . . 73
2.9 Constants . . . 73
2.9.1 Integer constants . . . 73
2.9.2 Real constants . . . 77
2.10 Summary . . . 78
2.11 Exercises . . . 79

3 Control of Flow and Logical Expressions 81
3.1 The Task ahead . . . 81
3.1.1 Logical expressions and Relational Operators . . . 81
3.2 Control of flow . . . 83
3.2.1 The if statement . . . 83
3.2.2 The while and do statements . . . 85
3.2.2.1 Handy hints . . . 86
3.2.3 The for statement . . . 87
3.2.4 A brief pause . . . 89
3.2.5 The switch statement . . . 89
3.2.5.1 The major restriction . . . 91
3.2.5.2 Integral Constant Expression . . . 91
3.2.6 The break statement . . . 92
3.2.7 The continue statement . . . 92
3.2.8 goto and labels . . . 93
3.2.9 Summary . . . 94
3.3 More logical expressions . . . 95
3.4 Strange operators . . . 96
3.4.1 The ?: operator . . . 97
3.4.2 The comma operator . . . 98
3.5 Summary . . . 99
3.6 Exercises . . . 100

4 Functions **101**
4.1 Changes . . . 101
4.1.1 Footnotes . . . 101
4.2 The type of functions . . . 102
4.2.1 Declaring functions . . . 102
4.2.2 The return statement . . . 104
4.2.3 Arguments to functions . . . 105
4.2.4 Function prototypes . . . 107
4.2.5 Argument Conversions . . . 110
4.2.6 Function definitions . . . 111
4.2.6.1 Summary . . . 112
4.2.7 Compound statements and declarations . . . 113
4.2.8 Footnotes . . . 114
4.3 Recursion and argument passing . . . 115
4.3.1 Call by value . . . 115
4.3.2 Call by reference . . . 116
4.3.3 Recursion . . . 117
4.3.4 Footnotes . . . 120
4.4 Linkage . . . 121
4.4.1 Effect of scope . . . 124
4.4.2 Internal static . . . 125
4.4.3 Footnotes . . . 126
4.5 Summary . . . 127
4.5.1 Footnotes . . . 128
4.6 Exercises . . . 128
4.6.1 Footnotes . . . 129

5 Arrays and Pointers **131**
5.1 Opening shots . . . 131
5.1.1 So why is this important? . . . 131
5.1.2 Effect of the Standard . . . 132
5.2 Arrays . . . 132
5.2.1 Multidimensional arrays . . . 133
5.3 Pointers . . . 135
5.3.1 Declaring pointers . . . 135
5.3.2 Arrays and pointers . . . 141
5.3.2.1 Summary . . . 143
5.3.3 Qualified types . . . 143
5.3.4 Pointer arithmetic . . . 145
5.3.5 void, null and dubious pointers . . . 146
5.4 Character handling . . . 148

5.4.1 Strings . . . 150
5.4.2 Pointers and increment operators . . . 152
5.4.3 Untyped pointers . . . 154
5.5 Sizeof and storage allocation . . . 155
5.5.1 What sizeof can't do . . . 165
5.5.2 The type of sizeof . . . 166
5.6 Pointers to functions . . . 166
5.7 Expressions involving pointers . . . 168
5.7.1 Conversions . . . 168
5.7.2 Arithmetic . . . 169
5.7.3 Relational expressions . . . 170
5.7.4 Assignment . . . 170
5.7.5 Conditional operator . . . 171
5.8 Summary . . . 171
5.9 Exercises . . . 172

6 Structured Data Types 173
6.1 History . . . 173
6.2 Structures . . . 174
6.2.1 Pointers and structures . . . 178
6.2.2 Linked lists and other structures . . . 181
6.2.3 Trees . . . 187
6.3 Unions . . . 192
6.4 Bitfields . . . 195
6.5 Enums . . . 196
6.6 Qualifiers and derived types . . . 197
6.7 Initialization . . . 198
6.7.1 Constant expressions . . . 198
6.7.2 More initialization . . . 199
6.8 Summary . . . 203
6.9 Exercises . . . 204

7 The Preprocessor 205
7.1 Effect of the Standard . . . 205
7.2 How the preprocessor works . . . 206
7.3 Directives . . . 208
7.3.1 The null directive . . . 208
7.3.2 # define . . . 209
7.3.2.1 Macro substitution . . . 210
7.3.2.2 Stringizing . . . 211
7.3.2.3 Token pasting . . . 212

7.3.2.4 Rescanning . . . 212
7.3.2.5 Notes . . . 213
7.3.3 # undef . . . 215
7.3.4 # include . . . 215
7.3.5 Predefined names . . . 217
7.3.6 #line . . . 218
7.3.7 Conditional compilation . . . 218
7.3.8 #pragma . . . 220
7.3.9 #error . . . 221
7.4 Summary . . . 221
7.5 Exercises . . . 222

8 Specialized Areas of C 223
8.1 Government Health Warning . . . 223
8.2 Declarations, Definitions and Accessibility . . . 223
8.2.1 Storage class specifiers . . . 224
8.2.1.1 Duration . . . 224
8.2.2 Scope . . . 227
8.2.3 Linkage . . . 228
8.2.4 Linkage and definitions . . . 229
8.2.5 Realistic use of linkage and definitions . . . 231
8.3 Typedef . . . 233
8.4 Const and volatile . . . 236
8.4.1 Const . . . 237
8.4.2 Volatile . . . 239
8.4.2.1 Indivisible Operations . . . 243
8.5 Sequence points . . . 244
8.6 Summary . . . 246

9 Libraries 247
9.1 Introduction . . . 247
9.1.1 Headers and standard types . . . 247
9.1.2 Character set and cultural dependencies . . . 248
9.1.3 The ¡stddef.h¿ Header . . . 249
9.1.4 The ¡errno.h¿ Header . . . 250
9.2 Diagnostics . . . 251
9.3 Character handling . . . 252
9.4 Localization . . . 254
9.4.1 The setlocale function . . . 256
9.4.2 The localeconv function . . . 257
9.5 Limits . . . 258

9.5.1 Limits.h . . . 258
9.5.2 Float.h . . . 258
9.6 Mathematical functions . . . 261
9.7 Non-local jumps . . . 263
9.8 Signal handling . . . 265
9.9 Variable numbers of arguments . . . 268
9.10 Input and output . . . 270
9.10.1 Introduction . . . 270
9.10.2 The I/O model . . . 271
9.10.2.1 Text streams . . . 271
9.10.2.2 Binary streams . . . 272
9.10.2.3 Other streams . . . 272
9.10.3 The stdio.h header file . . . 273
9.10.4 Opening, closing and buffering of streams . . . 274
9.10.4.1 Opening . . . 274
9.10.4.2 Closing . . . 274
9.10.4.3 Buffering . . . 274
9.10.5 Direct file manipulation . . . 275
9.10.6 Opening named files . . . 276
9.10.7 Freopen . . . 278
9.10.8 Closing files . . . 278
9.10.9 Setbuf, setvbuf . . . 278
9.10.10 Fflush . . . 279
9.11 Formatted I/O . . . 279
9.11.1 Output: the printf family . . . 280
9.11.2 Input: the scanf family . . . 282
9.12 Character I/O . . . 284
9.12.1 Character input . . . 285
9.12.2 Character output . . . 286
9.12.3 String output . . . 286
9.12.4 String input . . . 286
9.13 Unformatted I/O . . . 286
9.14 Random access functions . . . 287
9.14.1 Error handling . . . 289
9.15 General Utilities . . . 290
9.15.1 String conversion functions . . . 290
9.15.2 Random number generation . . . 292
9.15.3 Memory allocation . . . 292
9.15.4 Communication with the environment . . . 293
9.15.5 Searching and sorting . . . 294
9.15.6 Integer arithmetic functions . . . 295

9.15.7 Functions using multibyte characters 296
9.16 String handling 297
9.16.1 Copying 297
9.16.2 String and byte comparison 298
9.16.3 Character and string searching functions 299
9.16.4 Miscellaneous functions 300
9.17 Date and time 300
9.18 Summary 303

10 Complete Programs in C **305**
10.1 Putting it all together 305
10.2 Arguments to main 305
10.3 Interpreting program arguments 308
10.4 A pattern matching program 311
10.5 A more ambitious example 316
10.6 Afterword 331

Answers to Exercises **333**
Chapter 1 333
Chapter 2 337
Chapter 3 341
Chapter 4 342
Chapter 5 346
Chapter 6 348
Chapter 7 349