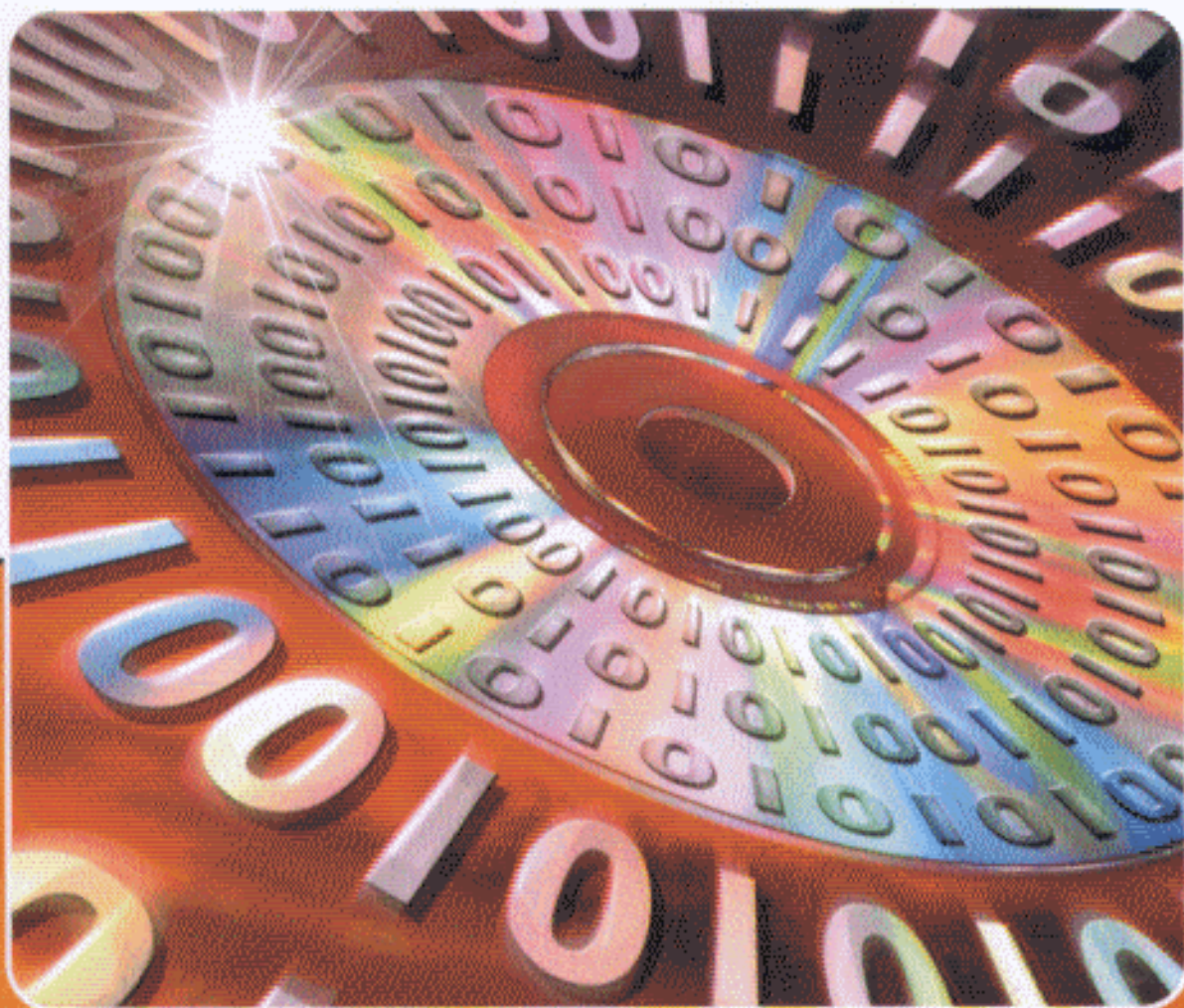


Software Development in C

A Practical Approach to Programming and Design



David Conger



Contents

CHAPTER 1	A BRIEF HISTORY OF C	2
Objectives	2	
Outline	2	
Preview	3	
1.1 Beginning with Binary	3	
1.2 Improving Things with Assembler	3	
1.3 Function Oriented Languages	4	
1.4 C	4	
1.5 A Word About C++	5	
Summary	5	
Technical Notes	6	
Review Questions	6	
Exercises	6	
Glossary	7	
CHAPTER 2	A FIRST LOOK AT C PROGRAMMING	8
Objectives	8	
Outline	8	
Preview	9	
2.1 What Is a Computer Program?	10	
2.2 Designing Programs	11	
2.3 Developing Programs	11	
2.4 Debugging Programs	13	
2.5 Hello, World—Our First C Program	14	
2.6 Getting Input with scanf()	17	
2.7 Using Pseudocode	19	
2.8 Troubleshooting Guide	19	
Summary	22	
Technical Notes	22	
Tips	22	
Review Questions	23	
Exercises	23	
Glossary	24	
CHAPTER 3	ATOMIC DATA TYPES: INTEGER AND FLOATING-POINT VARIABLES	26
Objectives	26	
Outline	26	
Preview	27	
3.1 What Are Atomic Data Types?	28	
3.2 Variables and Variable Names	28	
3.3 Signed Integers	29	
3.4 Unsigned Integers	34	
3.5 Floating-Point Numbers	34	
3.6 Overflow and Underflow	35	
3.7 Troubleshooting Guide	36	
Summary	40	
Technical Notes	40	
Tips	40	
Traps	40	
Review Questions	41	
Exercises	41	
Glossary	42	
CHAPTER 4	ATOMIC DATA TYPES IN C: CHARACTERS	44
Objectives	44	
Outline	44	
Preview	45	
4.1 Characters	45	
4.2 Signed and Unsigned Characters	48	
4.3 Troubleshooting Guide	48	
Summary	49	
Technical Note	49	
Trap	49	
Review Questions	49	
Exercises	50	
Glossary	51	
CHAPTER 5	INTRODUCTION TO C OPERATORS	52
Objectives	52	
Outline	52	
Preview	53	
5.1 Addition, Subtraction, Multiplication, and Division	53	
5.2 The Process of Program Development	57	
5.3 Precedence	60	
5.4 Troubleshooting Guide	62	
Summary	62	
Tips	62	
Review Questions	63	
Exercises	63	
Glossary	64	

CHAPTER 6	OTHER NUMERIC OPERATORS	66
Objectives	66	6.6 Troubleshooting Guide
Outline	66	Summary
Preview	67	Tips
6.1 Increment and Decrement	67	Traps
6.2 Modulus	69	Review Questions
6.3 sizeof	70	Exercises
6.4 +=, -=, *=, and /=	71	Glossary
6.5 Type Casting	71	

CHAPTER 7	LOGICAL OPERATORS	80
Objectives	80	7.7 Troubleshooting Guide
Outline	80	Summary
Preview	81	Technical Notes
7.1 Logical Operators	81	Tips
7.2 What Is Truth?	82	Traps
7.3 Arithmetic Values and Truth	82	Review Questions
7.4 Logical Operators	82	Exercises
7.5 Logical AND	89	Glossary
7.6 Logical OR	90	

CHAPTER 8	FLOW CONTROL: BRANCHING	100
Objectives	100	Technical Notes
Outline	100	Tips
Preview	101	Traps
8.1 What Are Flow Control Statements?	101	Review Questions
8.2 Branching	101	Exercises
8.3 Troubleshooting Guide	112	Glossary
Summary	115	

CHAPTER 9	FLOW CONTROL: LOOPING	120
Objectives	120	Technical Notes
Outline	120	Tips
Preview	121	Traps
9.1 Looping: You Can Say That Again	121	Review Questions
9.2 Troubleshooting Guide	130	Exercises
Summary	134	Glossary

CHAPTER 10	SINGLE-DIMENSIONAL ARRAYS	140
Objectives	140	Summary
Outline	140	Technical Notes
Preview	141	Tips
10.1 Declaring Arrays	141	Traps
10.2 Accessing Array Elements	142	Review Questions
10.3 Initializing Arrays	149	Exercises
10.4 Magic Numbers	152	Glossary
10.5 Troubleshooting Guide	157	

CHAPTER 11	MULTIDIMENSIONAL ARRAYS	166
Objectives	166	11.4 Initializing Multidimensional Arrays
Outline	166	11.5 Using Multidimensional Arrays in Programs
Preview	167	11.6 Troubleshooting Guide
11.1 Extending Single-Dimensional Arrays	167	Summary
11.2 Declaring Multidimensional Arrays	168	Technical Notes
11.3 Accessing Array Elements	168	Tips

Trap	177	Exercises	178
Review Questions	178	Glossary	180

CHAPTER 12 USER I/O, STRINGS, AND STRING FUNCTIONS 182

Objectives	182	Technical Notes	196
Outline	182	Tip	196
Preview	183	Traps	196
12.1 Input and Output of Characters and Strings	183	Review Questions	197
12.2 String Handling Functions	190	Exercises	197
12.3 Troubleshooting Guide	194	Glossary	198
Summary	196		

CHAPTER 13 STRUCTURED DESIGN WITH FUNCTIONS 200

Objectives	200	Summary	224
Outline	200	Technical Notes	224
Preview	201	Tips	224
13.1 Structured Design	202	Traps	225
13.2 Creating a Function	208	Review Questions	225
13.3 Structured Design with Functions	219	Exercises	225
13.4 Troubleshooting Guide	221	Glossary	228

CHAPTER 14 PROGRAMMER-DEFINED DATA TYPES 230

Objectives	230	14.6 Troubleshooting Guide	245
Outline	230	Summary	247
Preview	231	Technical Notes	247
14.1 Modeling Data	231	Tips	247
14.2 Structures	232	Traps	247
14.3 Unions	240	Review Questions	248
14.4 Enumerated Types	241	Exercises	248
14.5 typedef	244	Glossary	251

CHAPTER 15 DESIGNING DATA TYPES 252

Objectives	252	Summary	261
Outline	252	Technical Notes	261
Preview	253	Tips	261
15.1 Type Attributes	253	Review Questions	261
15.2 Operations on Programmer-Defined Types	254	Exercises	262
15.3 Identifying Essential Types in a Program	257	Glossary	263
15.4 Troubleshooting Guide	259		

CHAPTER 16 PREPROCESSOR DIRECTIVES 264

Objectives	264	16.8 #pragma	277
Outline	264	16.9 Troubleshooting Guide	277
Preview	265	Summary	279
16.1 What Is the Preprocessor?	266	Technical Notes	280
16.2 #define	267	Tips	280
16.3 #ifdef - #else - #endif	272	Trap	280
16.4 #ifndef - #else - #endif	274	Review Questions	280
16.5 #if defined() and #if !defined()	275	Exercises	280
16.6 #elif	275	Glossary	282
16.7 #include	276		

CHAPTER 17 ORGANIZING PROGRAMS 284

Objectives	284	Preview	285
Outline	284	17.1 Organizing Files	285

17.2 Troubleshooting Guide	288
Summary	290
Technical Note	290
Tips	290

Trap	290
Review Questions	291
Exercises	291

CHAPTER 18 POINTERS

294

Objectives	294
Outline	294
Preview	295
18.1 What Is a Pointer?	296
18.2 Using Pointers	296
18.3 Pointers and Arrays	301
18.4 Pointers and Functions	309
18.5 Pointers and Strings	320
18.6 Using Pointers in the Text Editor	324

18.7 Troubleshooting Guide	337
Summary	340
Technical Notes	340
Tips	341
Traps	341
Review Questions	341
Exercises	342
Glossary	344

CHAPTER 19 DYNAMIC MEMORY ALLOCATION

346

Objectives	346
Outline	346
Preview	347
19.1 Allocation and the Heap	348
19.2 Allocation and Deallocation	348
19.3 Dynamic Allocation in the Text Editor	359
19.4 Troubleshooting Guide	364

Summary	368
Technical Notes	369
Tips	369
Traps	369
Review Questions	369
Exercises	370
Glossary	373

CHAPTER 20 ENCAPSULATING DATA

374

Objectives	374
Outline	374
Preview	375
20.1 Scope and Persistence	376
20.2 Storage Classes	376
20.3 const	381
20.4 Troubleshooting Guide	383

Summary	386
Technical Notes	386
Tips	387
Traps	387
Review Questions	387
Exercises	387
Glossary	390

CHAPTER 21 FILE INPUT AND OUTPUT

392

Objectives	392
Outline	392
Preview	393
21.1 The Standard Files	393
21.2 File Input and Output	394
21.3 Printing Text	413
21.4 Adding File I/O to the Text Editor	414
21.5 Parsing Command-Line Arguments	418

21.6 Troubleshooting Guide	419
Summary	423
Technical Notes	424
Tips	424
Traps	425
Review Questions	425
Exercises	425
Glossary	426

CHAPTER 22 FIDDLING WITH BITS

428

Objectives	428
Outline	428
Preview	429
22.1 Bytes and Bits	429
22.2 Bitwise Operators	430
22.3 Bitfield Structures	439
22.4 Troubleshooting Guide	441

Summary	443
Technical Notes	443
Tips	443
Traps	443
Review Questions	444
Exercises	444
Glossary	446

CHAPTER 23 DESIGNING THE TEXT EDITOR

448

Objectives	448
------------------	-----

Outline	448
---------------	-----

Preview	449	23.8 Troubleshooting Guide	463
23.1 The Purpose of the Program	449	Summary	465
23.2 The Program Specification	450	Technical Notes	465
23.3 Top-Down Design	451	Tips	465
23.4 Designing the Program's Types	452	Traps	465
23.5 Filling Out Function Parameter Lists	458	Review Questions	466
23.6 Pseudocoding the Text Editor	460	Exercises	466
23.7 A Project Plan for the Text Editor	462	Glossary	466
CHAPTER 24	DEVELOPING THE TEXT EDITOR: TEDIT.C		468
Objectives	468	Summary	473
Outline	468	Technical Note	473
Preview	469	Review Questions	473
24.1 A Quick Look at the Finished Program	469	Exercises	473
24.2 Writing TEdit.c	470	Glossary	475
24.3 Troubleshooting Guide	471		
CHAPTER 25	DEVELOPING THE TEXT EDITOR: TBUFFER.H, TBUFFER.C, TSTRING.H, AND TSTRING.C		476
Objectives	476	Technical Notes	485
Outline	476	Tip	485
Preview	477	Trap	485
25.1 Writing TBuffer.h and TBuffer.c	477	Review Questions	485
25.2 Writing TString.h and TString.c	480	Exercises	486
25.3 Troubleshooting Guide	482	Glossary	487
Summary	485		
CHAPTER 26	DEVELOPING THE TEXT EDITOR: INSTLIST.H, INSTLIST.C, MISCTYPE.H, AND PLATFORM.H		488
Objectives	488	26.2 Writing Platform.h and MiscType.h	490
Outline	488	26.3 Project Post-Mortem	490
Preview	489	Review Questions	491
26.1 Writing InstList.h and InstList.c	489	Exercises	491
CHAPTER 27	MOVING TO C++, C#, AND JAVA		494
Objectives	494	27.3 Java	497
Outline	494	27.4 C#	497
Preview	495	Summary	498
27.1 C as the Basis for Other Languages	495	Technical Notes	498
27.2 C++	495	Glossary	498
APPENDIX A	INSTALLING AND USING VISUAL C++		500
Visual C++ Special Edition	500	Compiling with VC++	501
Installing VC++	500	Other Compilers	501
Using VC++	500		
APPENDIX B	THE ASCII CHARACTER SET		502
APPENDIX C	OPERATOR PRECEDENCE IN C		506
APPENDIX D	BINARY, DECIMAL, AND HEXADECIMAL NUMBERS		508
Number Bases	508	Converting Decimal to Hexadecimal	511
Converting Between Bases	510	Converting Binary to Hexadecimal	512
Converting Any Base to Decimal	510	Converting Hexadecimal to Binary	513
Converting Decimal to Binary	511	Exercises	513
INDEX			514