

Table of Contents

Preface.....	ix
<hr/>	
Part I. The Environment	
1. Set Yourself Up for Easy Compilation.....	3
Use a Package Manager	4
Compiling C with Windows	6
POSIX for Windows	6
Compiling C with POSIX	8
Compiling C Without POSIX	9
Which Way to the Library?	10
A Few of My Favorite Flags	12
Paths	13
Runtime Linking	16
Using Makefiles	17
Setting Variables	17
The Rules	20
Using Libraries from Source	24
Using Libraries from Source (Even if Your Sysadmin Doesn't Want You To)	25
Compiling C Programs via Here Document	27
Include Header Files from the Command Line	27
The Unified Header	28
Here Documents	29
Compiling from stdin	30
2. Debug, Test, Document.....	33
Using a Debugger	33

A Debugging Detective Story	36
GDB Variables	45
Print Your Structures	46
Using Valgrind to Check for Errors	50
Unit Testing	52
Using a Program as a Library	55
Coverage	57
Error Checking	58
What is the User's Involvement in the Error?	58
The Context in Which the User is Working	59
How Should the Error Indication Be Returned?	61
Interweaving Documentation	62
Doxygen	62
Literate Code with CWEB	64
3. Packaging Your Project.....	67
The Shell	68
Replacing Shell Commands with Their Outputs	69
Use the Shell's for Loops to Operate on a Set of Files	70
Test for Files	72
fc	75
Makefiles vs. Shell Scripts	77
Packaging Your Code with Autotools	79
An Autotools Demo	81
Describing the Makefile with Makefile.am	84
The configure Script	89
4. Version Control.....	95
Changes via diff	96
Git's Objects	97
The Stash	102
Trees and Their Branches	102
Merging	104
The Rebase	105
Remote Repositories	106
5. Playing Nice with Others.....	109
Dynamic Loading	109
The Limits of Dynamic Loading	112
The Process	112
Writing to Be Read by Nonnatives	113
The Wrapper Function	113

Smuggling Data Structures Across the Border	114
Linking	116
Python Host	116
Compiling and Linking	118
The Conditional Subdirectory for Automake	118
Distutils Backed with Autotools	120

Part II. The Language

6. Your Pal the Pointer.....	125
Automatic, Static, and Manual Memory	125
Persistent State Variables	130
Pointers Without malloc	131
Structures Get Copied, Arrays Get Aliased	133
malloc and Memory-Twiddling	136
The Fault Is in Our Stars	137
All the Pointer Arithmetic You Need to Know	138
Typedef as a teaching tool	141
7. Inessential C Syntax that Textbooks Spend a Lot of Time Covering.....	143
Don't Bother Explicitly Returning from main	143
Let Declarations Flow	144
Set Array Size at Runtime	146
Cast Less	147
Enums and Strings	149
Labels, gotos, switches, and breaks	150
goto Considered	151
switch	154
Deprecate Float	155
Comparing Unsigned Integers	158
Safely Parse Strings to Numbers	158
8. Important C Syntax that Textbooks Often Do Not Cover.....	163
Cultivate Robust and Flourishing Macros	163
The Preprocessor	168
Test Macros	172
Header Guards	174
Linkage with static and extern	176
Externally Linked Variables in Header Files	177
The const Keyword	179
Noun-Adjective Form	180

Tension	181
Depth	181
The char const ** Issue	182
9. Easier Text Handling.....	187
Making String Handling Less Painful with asprintf	187
Security	190
Constant Strings	191
Extending Strings with asprintf	193
A Pæan to strtok	194
Unicode	199
The Encoding for C Code	201
Unicode Libraries	202
The Sample Code	203
10. Better Structures.....	207
Compound Literals	208
Initialization via Compound Literals	209
Variadic Macros	210
Safely Terminated Lists	211
Multiple Lists	212
Foreach	214
Vectorize a Function	214
Designated Initializers	216
Initialize Arrays and Structs with Zeros	218
Typedefs Save the Day	219
A Style Note	221
Return Multiple Items from a Function	222
Reporting Errors	223
Flexible Function Inputs	225
Declare Your Function as printf-Style	226
Optional and Named Arguments	228
Polishing a Dull Function	231
The Void Pointer and the Structures It Points To	236
Functions with Generic Inputs	237
Generic Structures	241
11. Object-Oriented Programming in C.....	247
Extending Structures and Dictionaries	249
Implementing a Dictionary	251
C, with fewer seams	255
Functions in Your Structs	260

Vtables	264
Scope	269
Private Struct Elements	271
Overload	272
_Generic	274
Count References	277
Example: A Substring Object	277
Example: An Agent-Based Model of Group Formation	282
Conclusion	288
12. Parallel Threads.....	291
The Environment	292
The Ingredients	293
OpenMP	294
Compiling OpenMP, pthreads, and C atoms	296
Interference	297
Map-reduce	298
Multiple Tasks	299
Thread Local	300
Localizing Nonstatic Variables	302
Shared Resources	302
Atoms	307
Pthreads	310
C atoms	314
Atomic structs	317
13. Libraries.....	323
GLib	323
POSIX	324
Parsing Regular Expressions	324
Using mmap for Gigantic Data Sets	329
The GNU Scientific Library	332
SQLite	334
The Queries	336
libxml and cURL	337
Epilogue.....	343
A. C 101.....	345

