
Table of Contents

Preface.....	ix
--------------	----

Part I. The Environment

1. Set Yourself Up for Easy Compilation.....	1
Use a Package Manager	2
Compiling C with Windows	4
POSIX for Windows	4
Compiling C with POSIX	6
Compiling C Without POSIX	7
Which Way to the Library?	8
A Few of My Favorite Flags	10
Paths	11
Runtime Linking	14
Using Makefiles	15
Setting Variables	15
The Rules	18
Using Libraries from Source	22
Using Libraries from Source (Even if Your Sysadmin Doesn't Want You To)	23
Compiling C Programs via Here Document	25
Include Header Files from the Command Line	25
The Unified Header	26
Here Documents	27
Compiling from stdin	28
2. Debug, Test, Document.....	31
Using a Debugger	31

A Debugging Detective Story	34
GDB Variables	43
Print Your Structures	44
Using Valgrind to Check for Errors	48
Unit Testing	50
Using a Program as a Library	53
Coverage	54
Error Checking	55
What is the User's Involvement in the Error?	56
The Context in Which the User is Working	57
How Should the Error Indication Be Returned?	59
Interweaving Documentation	59
Doxygen	60
Literate Code with CWEB	61
3. Packaging Your Project.	65
The Shell	66
Replacing Shell Commands with Their Outputs	67
Use the Shell's for Loops to Operate on a Set of Files	68
Test for Files	70
fc	73
Makefiles vs. Shell Scripts	75
Packaging Your Code with Autotools	77
An Autotools Demo	79
Describing the Makefile with Makefile.am	82
The configure Script	87
4. Version Control.	93
Changes via diff	94
Git's Objects	95
The Stash	99
Trees and Their Branches	100
Merging	101
The Rebase	103
Remote Repositories	104
5. Playing Nice with Others.	107
Dynamic Loading	107
The Limits of Dynamic Loading	110
The Process	110
Writing to Be Read by Nonnatives	111
The Wrapper Function	111

Smuggling Data Structures Across the Border	112
Linking	114
Python Host	114
Compiling and Linking	116
The Conditional Subdirectory for Automake	116
Distutils Backed with Autotools	118

Part II. The Language

6. Your Pal the Pointer.....	123
Automatic, Static, and Manual Memory	123
Persistent State Variables	128
Pointers Without malloc	129
Structures Get Copied, Arrays Get Aliased	131
malloc and Memory-Twiddling	134
The Fault Is in Our Stars	135
All the Pointer Arithmetic You Need to Know	136
Typedef as a teaching tool	139
7. Inessential C Syntax that Textbooks Spend a Lot of Time Covering.....	141
Don't Bother Explicitly Returning from main	141
Let Declarations Flow	142
Set Array Size at Runtime	144
Cast Less	145
Enums and Strings	147
Labels, gotos, switches, and breaks	148
goto Considered	149
switch	152
Deprecate Float	153
Comparing Unsigned Integers	156
Safely Parse Strings to Numbers	156
8. Important C Syntax that Textbooks Often Do Not Cover.....	161
Cultivate Robust and Flourishing Macros	161
The Preprocessor	166
Test Macros	170
Header Guards	172
Linkage with static and extern	174
Externally Linked Variables in Header Files	175
The const Keyword	177
Noun-Adjective Form	178

Tension	179
Depth	179
The char const ** Issue	180
9. Easier Text Handling.....	185
Making String Handling Less Painful with asprintf	185
Security	188
Constant Strings	189
Extending Strings with asprintf	191
A Pæan to strtok	192
Unicode	197
The Encoding for C Code	199
Unicode Libraries	200
The Sample Code	201
10. Better Structures.....	205
Compound Literals	206
Initialization via Compound Literals	207
Variadic Macros	208
Safely Terminated Lists	209
Multiple Lists	210
Foreach	212
Vectorize a Function	212
Designated Initializers	214
Initialize Arrays and Structs with Zeros	216
Typedefs Save the Day	217
A Style Note	218
Return Multiple Items from a Function	220
Reporting Errors	221
Flexible Function Inputs	223
Declare Your Function as printf-Style	224
Optional and Named Arguments	226
Polishing a Dull Function	228
The Void Pointer and the Structures It Points To	234
Functions with Generic Inputs	234
Generic Structures	239
11. Object-Oriented Programming in C.....	245
Extending Structures and Dictionaries	247
Implementing a Dictionary	249
C, with fewer seams	253
Functions in Your Structs	258

Vtables	262
Scope	267
Private Struct Elements	268
Overload	270
_Generic	271
Count References	274
Example: A Substring Object	275
Example: An Agent-Based Model of Group Formation	279
Conclusion	286
12. Parallel Threads.....	289
The Environment	290
The Ingredients	291
OpenMP	292
Compiling OpenMP, pthreads, and C atoms	294
Interference	295
Map-reduce	296
Multiple Tasks	297
Thread Local	298
Localizing Nonstatic Variables	300
Shared Resources	300
Atoms	305
Pthreads	308
C atoms	312
Atomic structs	315
13. Libraries.....	321
GLib	321
POSIX	322
Parsing Regular Expressions	322
Using mmap for Gigantic Data Sets	327
The GNU Scientific Library	330
SQLite	332
The Queries	334
libxml and cURL	335
Epilogue.....	341
A. C 101.....	343

Glossary.....	363
References.....	367
Index.....	371