

# Contents

|                                                 |           |
|-------------------------------------------------|-----------|
| <b>Contents</b>                                 | <b>6</b>  |
| <b>Foreword</b>                                 | <b>11</b> |
| <b>Introduction</b>                             | <b>13</b> |
| How to Use This Book . . . . .                  | 14        |
| Bonus Material and Learning Resources . . . . . | 16        |
| <b>Getting the CPython Source Code</b>          | <b>21</b> |
| What's in the Source Code? . . . . .            | 22        |
| <b>Setting Up Your Development Environment</b>  | <b>25</b> |
| IDE or Editor? . . . . .                        | 25        |
| Setting Up Visual Studio . . . . .              | 27        |
| Setting Up Visual Studio Code . . . . .         | 29        |
| Setting Up JetBrains CLion . . . . .            | 34        |
| Setting Up Vim . . . . .                        | 38        |
| Conclusion . . . . .                            | 42        |

|                                                               |           |
|---------------------------------------------------------------|-----------|
| <b>Compiling CPython</b>                                      | <b>45</b> |
| Compiling CPython on macOS . . . . .                          | 46        |
| Compiling CPython on Linux . . . . .                          | 49        |
| Installing a Custom Version . . . . .                         | 50        |
| A Quick Primer on Make . . . . .                              | 51        |
| CPython's Make Targets . . . . .                              | 52        |
| Compiling CPython on Windows . . . . .                        | 55        |
| Profile-Guided Optimization . . . . .                         | 61        |
| Conclusion . . . . .                                          | 63        |
| <b>The Python Language and Grammar</b>                        | <b>65</b> |
| Why CPython Is Written in C and Not Python . . . . .          | 66        |
| The Python Language Specification . . . . .                   | 68        |
| The Parser Generator . . . . .                                | 73        |
| Regenerating Grammar . . . . .                                | 74        |
| Conclusion . . . . .                                          | 79        |
| <b>Configuration and Input</b>                                | <b>81</b> |
| Configuration State . . . . .                                 | 84        |
| Build Configuration . . . . .                                 | 88        |
| Building a Module From Input . . . . .                        | 88        |
| Conclusion . . . . .                                          | 94        |
| <b>Lexing and Parsing With Syntax Trees</b>                   | <b>95</b> |
| Concrete Syntax Tree Generation . . . . .                     | 96        |
| The CPython Parser-Tokenizer . . . . .                        | 100       |
| Abstract Syntax Trees . . . . .                               | 105       |
| Important Terms to Remember . . . . .                         | 115       |
| Example: Adding an Almost-Equal Comparison Operator . . . . . | 115       |
| Conclusion . . . . .                                          | 121       |

|                                                           |                |
|-----------------------------------------------------------|----------------|
| <b>The Compiler</b>                                       | <b>123</b>     |
| Related Source Files . . . . .                            | 124            |
| Important Terms . . . . .                                 | 125            |
| Instantiating a Compiler . . . . .                        | 126            |
| Future Flags and Compiler Flags . . . . .                 | 127            |
| Symbol Tables . . . . .                                   | 129            |
| Core Compilation Process . . . . .                        | 136            |
| Assembly . . . . .                                        | 142            |
| Creating a Code Object . . . . .                          | 147            |
| Using Instaviz to Show a Code Object . . . . .            | 148            |
| Example: Implementing the Almost-Equal Operator . . . . . | 150            |
| Conclusion . . . . .                                      | 156            |
| <br><b>The Evaluation Loop</b>                            | <br><b>157</b> |
| Related Source Files . . . . .                            | 158            |
| Important Terms . . . . .                                 | 158            |
| Constructing Thread State . . . . .                       | 159            |
| Constructing Frame Objects . . . . .                      | 160            |
| Frame Execution . . . . .                                 | 168            |
| The Value Stack . . . . .                                 | 171            |
| Example: Adding an Item to a List . . . . .               | 177            |
| Conclusion . . . . .                                      | 182            |
| <br><b>Memory Management</b>                              | <br><b>183</b> |
| Memory Allocation in C . . . . .                          | 184            |
| Design of the Python Memory Management System . . . . .   | 187            |
| The CPython Memory Allocator . . . . .                    | 189            |
| The Object and PyMem Memory Allocation Domains . . . . .  | 199            |
| The Raw Memory Allocation Domain . . . . .                | 203            |
| Custom Domain Allocators . . . . .                        | 203            |
| Custom Memory Allocation Sanitizers . . . . .             | 204            |
| The PyArena Memory Arena . . . . .                        | 207            |
| Reference Counting . . . . .                              | 208            |
| Garbage Collection . . . . .                              | 215            |
| Conclusion . . . . .                                      | 226            |

|                                                    |                |
|----------------------------------------------------|----------------|
| <b>Parallelism and Concurrency</b>                 | <b>227</b>     |
| Models of Parallelism and Concurrency . . . . .    | 229            |
| The Structure of a Process . . . . .               | 229            |
| Multiprocess Parallelism . . . . .                 | 232            |
| Multithreading . . . . .                           | 257            |
| Asynchronous Programming . . . . .                 | 271            |
| Generators . . . . .                               | 271            |
| Coroutines . . . . .                               | 278            |
| Asynchronous Generators . . . . .                  | 283            |
| Subinterpreters . . . . .                          | 285            |
| Conclusion . . . . .                               | 289            |
| <br><b>Objects and Types</b>                       | <br><b>291</b> |
| Examples in This Chapter . . . . .                 | 292            |
| Built-in Types . . . . .                           | 293            |
| Object and Variable Object Types . . . . .         | 294            |
| The type Type . . . . .                            | 295            |
| The bool and long Types . . . . .                  | 299            |
| The Unicode String Type . . . . .                  | 304            |
| The Dictionary Type . . . . .                      | 314            |
| Conclusion . . . . .                               | 321            |
| <br><b>The Standard Library</b>                    | <br><b>323</b> |
| Python Modules . . . . .                           | 323            |
| Python and C Modules . . . . .                     | 325            |
| <br><b>The Test Suite</b>                          | <br><b>329</b> |
| Running the Test Suite on Windows . . . . .        | 329            |
| Running the Test Suite on Linux or macOS . . . . . | 330            |
| Test Flags . . . . .                               | 331            |
| Running Specific Tests . . . . .                   | 331            |
| Testing Modules . . . . .                          | 333            |
| Test Utilities . . . . .                           | 334            |
| Conclusion . . . . .                               | 335            |

|                                                               |                |
|---------------------------------------------------------------|----------------|
| <b>Debugging</b>                                              | <b>337</b>     |
| Using the Crash Handler . . . . .                             | 338            |
| Compiling Debug Support . . . . .                             | 338            |
| Using LLDB for macOS . . . . .                                | 339            |
| Using GDB . . . . .                                           | 343            |
| Using Visual Studio Debugger . . . . .                        | 346            |
| Using CLion Debugger . . . . .                                | 348            |
| Conclusion . . . . .                                          | 353            |
| <br><b>Benchmarking, Profiling, and Tracing</b>               | <br><b>355</b> |
| Using <code>timeit</code> for Microbenchmarks . . . . .       | 356            |
| Using the Python Benchmark Suite for Runtime Benchmarks       | 358            |
| Profiling Python Code With <code>cProfile</code> . . . . .    | 364            |
| Profiling C Code With <code>DTrace</code> . . . . .           | 367            |
| Conclusion . . . . .                                          | 372            |
| <br><b>Next Steps</b>                                         | <br><b>373</b> |
| Writing C Extensions for CPython . . . . .                    | 373            |
| Improving Your Python Applications . . . . .                  | 374            |
| Contributing to the CPython Project . . . . .                 | 375            |
| Keep Learning . . . . .                                       | 378            |
| <br><b>Appendix: Introduction to C for Python Programmers</b> | <br><b>381</b> |
| The C Preprocessor . . . . .                                  | 381            |
| Basic C Syntax . . . . .                                      | 385            |
| Conclusion . . . . .                                          | 392            |