Contents

Preface v

I Introduction 1

1 Compilers 3

1.1 How Does a Compiler Work? 5

1.2 Phases of Compilation 6

1.3 A Simplified Model 13

2 The Peculiarities of C 15

3 Rules of the Game 19

3.1 The Source Language 19

3.2 The Object Language 21

3.3 The Runtime Library 22

II The Tour 23

4 Definitions and Data Structures 25

4.1 Definitions 25

4.2 Global Data 30

4.3 Function Prototypes 35

5 Utility Functions 39

6 Error Handling 43

7 Lexical Analysis 47

7.1 A Brief Theory of Scanning 56

8 Symbol Table Management 67

9 Syntax and Semantic Analysis **77**

9.1 A Brief Theory of Parsing 78

9.1.1 Mapping Grammars to Parsers 82

9.2 Expression Parsing 87

9.3 Constant Expression Parsing 115

9.4 Statement Parsing 119

9.5 Declaration Parsing 133

10 Preprocessing **151**

11 Code Generation **159**

11.1 A Brief Theory of Code Generation 159

11.2 The Code Generator 162

11.3 Framework 165

11.4 Load Operations 167

11.5 Binary Operators 168

11.6 Unary Operators 174

11.7 Jumps and Function Calls 175

11.8 Data Definitions 178

11.9 Increment Operators 179

11.10Switch Table Generation 183

11.11Store Operations 184

11.12Rvalue Computation 186

12 Target Description **189**

12.1 The 386 Target 192

12.2 Framework 192

12.3 Load Operations 193

12.4 Miscellanea 195

12.5 Binary Operations 196

12.6 Unary Operations 199

12.7 Increment Operations 200

12.8 Jumps and Branches 203

12.9 Store Operations 205

12.10Functions and Function Calls 206

12.11Data Definitions 207

13 The Compiler Controller **209**

III Runtime Environment 219

14 The Runtime Startup Module 221
14.1 The System Calls 225
14.2 The System Call Header 231

15 The Runtime Library 233
15.1 Library Initialization 233
15.2 Standard I/O 234
15.2.1 The stdio.h Header 235
15.2.2 Required Stdio Functions 239
15.3 Utility Library 263
15.3.1 The stdlib.h Header 263
15.3.2 Required Stdlib Functions 264
15.4 String Library 271
15.4.1 The string.h Header 271
15.4.2 Required String Functions 272
15.5 Character Types 275
15.5.1 The ctype.h Header 275
15.5.2 The Ctype Functions 276
15.6 The errno.h Header 277

IV Beyond SubC 279

16 Code Synthesis 281
16.1 Instruction Queuing 282
16.1.1 CISC versus RISC 291
16.1.2 Comparisons and Conditional Jumps 292
16.2 Register Allocation 294
16.2.1 Cyclic Register Allocation 298

17 Optimization 303
17.1 Peephole Optimization 303
17.2 Expression Rewriting 306
17.2.1 Constant Expression Folding 312
17.2.2 Strength Reduction 314
17.3 Common Subexpression Elimination 317
17.4 Emitting Code from an AST 322

V Conclusion 325

18 Bootstrapping a Compiler 327

18.1 Design 327

18.2 Implementation 328

18.3 Testing 330

18.4 Have Some Fun 332

VI Appendix 335

A Where Do We Go from Here? 337

A.1 Piece of Cake 337

A.2 This May Hurt a Bit 338

A.3 Bring 'Em On! 340

B (Sub)C Primer 343

B.1 Data Objects and Declarations 343

B.1.1 Void Pointers 346

B.2 Expressions 346

B.2.1 Pointer Arithmetics 349

B.3 Statements 350

B.4 Functions 352

B.5 Prototypes and External Declarations 353

B.6 Preprocessor 354

B.7 Library Functions 356

C 386 Assembly Primer 357

C.1 Registers 357

C.2 Assembler Syntax 359

C.2.1 Assembler Directives 359

C.2.2 Sample Program 360

C.3 Addressing Modes 361

C.3.1 Register 361

C.3.2 Immediate 361

C.3.3 Memory 361

C.3.4 Indirect 362

C.4 Instruction Summary 363

C.4.1 Move Instructions 363

C.4.2 Arithmetic Instructions 364

C.4.3 Branch/Call Instructions 365