

Table of Contents

Preface	ix
Chapter 1 Overview of Compilers and Language Translation.....	1
1.1 The Role of Programming Languages.....	1
1.2 Translators and Compilers	1
1.3 Tombstone Diagrams	4
1.4 Bootstrapping a Compiler.....	7
1.5 Interpreters.....	10
1.6 The Compiler Project.....	11
1.7 Essential Terms and Concepts.....	13
1.8 Exercises	14
Chapter 2 Structure of a Compiler.....	15
2.1 Scanner.....	16
2.2 Parser	16
2.3 Constraint Analyzer	17
2.4 Code Generator	17
2.5 Optimizer.....	18
2.6 Final Code Generator.....	19
2.7 Tables and Maps.....	19
2.8 Error Handler	20
2.9 Passes	21
2.10 Compiler Design Goals	21
2.11 Essential Terms and Concepts.....	22
2.12 Exercises	23
Chapter 3 Context-Free Grammars.....	25
3.1 Specifying a Programming Language.....	25
3.2 Context-Free Grammars	26
3.3 Alternate Rule Notations	33
3.4 Grammar Transformations	36
3.5 Derivations and Parse Trees.....	37
3.6 Abstract Syntax Trees.....	41
3.7 A Context-Free Grammar for Context-Free Grammars.....	42
3.8 Essential Terms and Concepts.....	43
3.9 Exercises	44

Chapter 4 The Programming Language CPRL	47
4.1 General Lexical Considerations.....	47
4.2 Declarations, Statements, and Expressions	48
4.3 Types	49
4.4 Statements.....	52
4.5 Programs	54
4.6 Subprograms	54
4.7 Essential Terms and Concepts.....	56
4.8 Exercises	57
Chapter 5 Lexical Analysis (a.k.a. Scanning)	59
5.1 Class Position	59
5.2 Class Source	59
5.3 Class Symbol	61
5.4 Class Token	62
5.5 Class ErrorHandler.....	63
5.6 Class TokenBuffer	64
5.7 Class Scanner	64
5.8 Handling Lexical Errors	71
5.9 Testing Class Scanner.....	73
5.10 Essential Terms and Concepts.....	75
5.11 Exercises	75
Chapter 6 Syntax Analysis (a.k.a. Parsing).....	77
6.1 Example: Implementing Method <code>parseLoopStmt()</code>	78
6.2 Recursive Descent Parsing	81
6.3 First and Follow Sets.....	87
6.4 LL(k) Grammars and Recursive Descent Parsing.....	94
6.5 Variables versus Variable Expressions	96
6.6 Handling Grammar Limitations	96
6.7 Scope and Visibility	99
6.8 Class <code>IdTable</code>	102
6.9 Parsing Variables and Variable Expressions	107
6.10 Class Parser	109
6.11 Essential Terms and Concepts.....	110
6.12 Exercises	111

Chapter 7 Error Handling/Recovery	113
7.1 Types of Compilation Errors	113
7.2 Handling Errors	115
7.3 Error Recovery	116
7.4 Additional Error Recovery Strategies	122
7.5 Essential Terms and Concepts	123
7.6 Exercises	123
Chapter 8 Abstract Syntax Trees	125
8.1 Overview of Abstract Syntax Trees	125
8.2 Structure of Abstract Syntax Trees	127
8.3 Extending Scopes with References to Declarations	129
8.4 Types and Declarations	135
8.5 Structural References versus Nonstructural References	137
8.6 Determining Types of Variables and Expressions	139
8.7 Maintaining Context During Parsing	142
8.8 Essential Terms and Concepts	144
8.9 Exercises	144
Chapter 9 Constraint Analysis	145
9.1 Overview of Constraint Analysis	145
9.2 Constraint Rules for CPRL/ \emptyset	149
9.3 Examples of Constraint Analysis	151
9.4 Essential Terms and Concepts	154
9.5 Exercises	154
Chapter 10 The CPRL Virtual Machine	157
10.1 Overview of the CVM	157
10.2 CVM Uses Relative Addressing	159
10.3 Loading a Program into Memory	159
10.4 Using the Stack to Hold Temporary Values	162
10.5 Essential Terms and Concepts	168
10.6 Exercises	169
Chapter 11 Code Generation	171
11.1 Overview of Code Generation	171
11.2 Labels and Branching	172
11.3 Load and Store Instructions	176
11.4 Computing Relative Addresses	177
11.5 Expressions	179

11.6 Statements	182
11.7 Disassembler	186
11.8 Essential Terms and Concepts.....	188
11.9 Exercises	189
Chapter 12 Code Optimization.....	191
12.1 Overview of Code Optimization	191
12.2 Common Optimizations	194
12.3 Optimization in CPRL	199
12.4 Essential Terms and Concepts.....	201
12.5 Exercises	201
Chapter 13 Subprograms	203
13.1 Review of Subprograms, Scope, and Parameters	203
13.2 Run-time Organization for Subprograms	207
13.3 Activation Record	209
13.4 Parameters	216
13.5 Subprogram Calls and Returns	218
13.6 Calling Conventions for CPRL on CVM	219
13.7 Computing Relative Addresses	220
13.8 Example of Program Execution	224
13.9 Essential Terms and Concepts.....	236
13.10 Exercises	236
Chapter 14 Arrays.....	239
14.1 Using CPRL Arrays	239
14.2 Implementing CPRL Arrays	241
14.3 Essential Terms and Concepts.....	244
14.4 Exercises	245
Chapter 15 Strings.....	247
15.1 Using CPRL Strings	247
15.2 Implementing CPRL Strings	249
15.3 Essential Terms and Concepts.....	254
15.4 Exercises	255
Chapter 16 Records.....	257
16.1 Using CPRL Records	257
16.2 Implementing CPRL Records	260
16.3 Essential Terms and Concepts.....	265
16.4 Exercises	265

Appendix A The Compiler Project	267
Appendix B Additional Project Exercises	277
Appendix C Definition of the Programming Language CPRL.....	281
C.1 Lexical Considerations.....	281
C.2 Types	283
C.3 Constants and Variables.....	285
C.4 Operators and Expressions	286
C.5 Statements	287
C.6 Programs.....	289
C.7 Subprograms.....	290
Appendix D The CPRL Grammar	293
Appendix E Definition of the CPRL Virtual Machine	297
E.1 Specification.	297
E.2 Implementation	298
E.3 CVM Instruction Set Architecture	301
Appendix F Searching for Reserved Words	311
F.1 Benchmarking the Search Algorithms	312
F.2 Sequential Search 1.....	313
F.3 Sequential Search 2.....	314
F.4 Sequential Search 3.....	315
F.5 Binary Search	316
F.6 Search by Length.....	317
F.7 Search by First Character	319
F.8 Gperf Hash Search	320
F.9 Search Using When Expression.....	323
F.10 Search Using HashMap.....	324
Annotated Compiler References and Websites.....	325
Index.....	329

