# Contents

Chapter 1
**SPECIFYING SYNTAX 1**

1.1 GRAMMARS AND BNF 2
Context-Free Grammars 4
Context-Sensitive Grammars 8
Exercises 8

1.2 THE PROGRAMMING LANGUAGE WREN 10
Ambiguity 12
Context Constraints in Wren 13
Semantic Errors in Wren 15
Exercises 16

1.3 VARIANTS OF BNF 18
Exercises 20

1.4 ABSTRACT SYNTAX 21
Abstract Syntax Trees 21
Abstract Syntax of a Programming Language 23
Exercises 29

1.5 FURTHER READING 30

Chapter 2
**INTRODUCTION TO LABORATORY ACTIVITIES 31**

2.1 SCANNING 33
Exercises 39

2.2 LOGIC GRAMMARS 40
Motivating Logic Grammars 41
Improving the Parser 44
Prolog Grammar Rules 46
Parameters in Grammars 47
Executing Goals in a Logic Grammar 49
Exercises 49

2.3 PARSING WREN 50
Handling Left Recursion 52
Left Factoring 55
Exercises 56

2.4 FURTHER READING 57

Chapter 3
**ATTRIBUTE GRAMMARS 59**

3.1 CONCEPTS AND EXAMPLES 59
Examples of Attribute Grammars 60
Formal Definitions 66

Semantics via Attribute Grammars 67
Exercises 71

3.2 AN ATTRIBUTE GRAMMAR FOR WREN 74
The Symbol Table 74
Commands 80
Expressions 82
Exercises 90

3.3 LABORATORY: CONTEXT CHECKING WREN 92
Declarations 96
Commands 99
Expressions 101
Exercises 102

3.4 FURTHER READING 103

Chapter 4
**TWO-LEVEL GRAMMARS 105**

4.1 CONCEPTS AND EXAMPLES 105
Fortran String Literals 111
Derivation Trees 113
Exercises 115

4.2 A TWO-LEVEL GRAMMAR FOR WREN 116
Declarations 117
Commands and Expressions 124
Exercises 132

4.3 TWO-LEVEL GRAMMARS AND PROLOG 132
Implementing Two-Level Grammars in Prolog 133
Two-Level Grammars and Logic Programming 136
Exercises 138

4.4 FURTHER READING 138

Chapter 5
**THE LAMBDA CALCULUS 139**

5.1 CONCEPTS AND EXAMPLES 140
Syntax of the Lambda Calculus 140
Curried Functions 143
Semantics of Lambda Expressions 145
Exercises 146

5.2 LAMBDA REDUCTION 147
Reduction Strategies 151
Correlation with Parameter Passing 155
Constants in the Pure Lambda Calculus 156
Functional Programming Languages 158
Exercises 158

5.3 LABORATORY: A LAMBDA CALCULUS EVALUATOR 160
Scanner and Parser 160
The Lambda Calculus Evaluator 162
Exercises 165

5.4 FURTHER READING 166

Chapter 6
**SELF-DEFINITION OF PROGRAMMING LANGUAGES 167**

6.1 SELF-DEFINITION OF LISP 167
Metacircular Interpreter 169
Running the Interpreter 174
Exercises 178

6.2 SELF-DEFINITION OF PROLOG 179
Displaying Failure 181
Exercises 185

6.3 FURTHER READING 185

Chapter 7
**TRANSLATIONAL SEMANTICS 187**

7.1 CONCEPTS AND EXAMPLES 187
A Program Translation 189
Exercises 191

7.2 ATTRIBUTE GRAMMAR CODE GENERATION 191
Expressions 193
Commands 201
Exercises 213

7.3 LABORATORY: IMPLEMENTING CODE GENERATION 215
Commands 217
Expressions 219
Exercises 221

7.4 FURTHER READING 222

Chapter 8
**TRADITIONAL OPERATIONAL SEMANTICS 223**

8.1 CONCEPTS AND EXAMPLES 224
VDL 226
Exercises 227

8.2 SECD: AN ABSTRACT MACHINE 228
Example 231
Parameter Passing 232
Static Scoping 233
Exercises 234

8.3 LABORATORY: IMPLEMENTING THE SECD MACHINE 235
Exercises 237

8.4 STRUCTURAL OPERATIONAL SEMANTICS: INTRODUCTION 238
Specifying Syntax 239
Inference Systems and Structural Induction 242
Exercises 244

8.5 STRUCTURAL OPERATIONAL SEMANTICS: EXPRESSIONS 245
Semantics of Expressions in Wren 245

Example 248
Outcomes 250
Exercises 252

8.6 STRUCTURAL OPERATIONAL SEMANTICS: COMMANDS 253
A Sample Computation 256
Semantic Equivalence 260
Natural Semantics 261
Exercises 262

8.7 LABORATORY: IMPLEMENTING STRUCTURAL OPERATIONAL SEMANTICS 264
Commands 265
Expressions 267
Top-Level Driver 268
Exercises 269

8.8 FURTHER READING 269

Chapter 9
**DENOTATIONAL SEMANTICS 271**

9.1 CONCEPTS AND EXAMPLES 271
The Syntactic World 272
The Semantic World 273
Compositionality 276
Exercises 277

9.2 A CALCULATOR LANGUAGE 277
Calculator Semantics 280
Semantic Functions 282
A Sample Calculation 283
Exercises 284

9.3 THE DENOTATIONAL SEMANTICS OF WREN 285
Semantic Domains 286
Language Constructs in Wren 288
Auxiliary Functions 290
Semantic Equations 290
Error Handling 293
Semantic Equivalence 294
Input and Output 294
Elaborating a Denotational Definition 296
Exercises 302

9.4 LABORATORY: IMPLEMENTING DENOTATIONAL SEMANTICS 304
Exercises 309

9.5 DENOTATIONAL SEMANTICS WITH ENVIRONMENTS 310
Environments 311
Stores 312
Semantic Functions 313
Semantic Equations 316
Procedures 318
Exercises 321

9.6 CHECKING CONTEXT-SENSITIVE SYNTAX 323
Exercises 327
9.7 CONTINUATION SEMANTICS 328
Continuations 331
The Programming Language Gull 333
Auxiliary Functions 335
Semantic Equations 336
The Error Continuation 336
Exercises 338
9.8 FURTHER READING 339

Chapter 10
**DOMAIN THEORY AND FIXED-POINT SEMANTICS 341**

10.1 CONCEPTS AND EXAMPLES 341
Recursive Definitions of Functions 342
Recursive Definitions of Sets (Types) 343
Modeling Nontermination 344
Exercises 345
10.2 DOMAIN THEORY 345
Elementary Domains 348
Product Domains 349
Sum Domains (Disjoint Unions) 351
Function Domains 355
Continuity of Functions on Domains 361
Exercises 363
10.3 FIXED-POINT SEMANTICS 365
First Step 366
Second Step 368
Continuous Functionals 374
Fixed points for Nonrecursive Functions 379
Revisiting Denotational Semantics 380
Fixed-Point Induction 382
Exercises 384
10.4 LABORATORY: RECURSION IN THE LAMBDA CALCULUS 388
Conditional Expressions 390
Paradoxical Combinator 390
Fixed-Point Identity 392
Exercises 393
10.5 FURTHER READING 394

Chapter 11
**AXIOMATIC SEMANTICS 395**

11.1 CONCEPTS AND EXAMPLES 395
Axiomatic Semantics of Programming Languages 396
11.2 AXIOMATIC SEMANTICS FOR WREN 398
Assignment Command 398
Input and Output 400

Rules of Inference 401
While Command and Loop Invariants 405
More on Loop Invariants 408
Nested While Loops 410
Exercises 415

11.3 AXIOMATIC SEMANTICS FOR PELICAN 418
Blocks 420
Nonrecursive Procedures 422
Recursive Procedures 425
Exercises 429

11.4 PROVING TERMINATION 432
Steps in Showing Termination 433
Termination of Recursive Procedures 435
Exercises 436

11.5 INTRODUCTION TO PROGRAM DERIVATION 437
Table of Cubes 437
Binary Search 440
Exercises 441

11.6 FURTHER READING 442

Chapter 12
**ALGEBRAIC SEMANTICS 443**

12.1 CONCEPTS AND EXAMPLES 444
A Module for Truth Values 446
Module Syntax 447
A Module for Natural Numbers 448
A Module for Characters 452
A Parameterized Module and Some Instantiations 453
A Module for Finite Mappings 456
Exercises 459

12.2 MATHEMATICAL FOUNDATIONS 460
Ground Terms 461
Σ-Algebras 461
A Congruence from the Equations 463
The Quotient Algebra 465
Homomorphisms 466
Consistency and Completeness 467
Exercises 469

12.3 USING ALGEBRAIC SPECIfiCATIONS 471
Data Abstraction 471
A Module for Unbounded Queues 472
Implementing Queues as Unbounded Arrays 474
Verification of Queue Axioms 477
ADTs As Algebras 477
Abstract Syntax and Algebraic Specifications 481
Exercise 485

12.4 ALGEBRAIC SEMANTICS FOR WREN 487
Types and Values in Wren 488
Abstract Syntax for Wren 489
A Type Checker for Wren 490
An Interpreter for Wren 494
A Wren System 498
Exercises 499
12.5 LABORATORY: IMPLEMENTING ALGEBRAIC SEMANTICS 499
Module Booleans 500
Module Naturals 501
Declarations 503
Commands 503
Expressions 505
Exercises 505
12.6 FURTHER READING 506

Chapter 13
**ACTION SEMANTICS 507**

13.1 CONCEPTS AND EXAMPLES 508
Data and Sorts 511
Yielders 514
Actions 515
The Functional Facet 515
The Imperative Facet 518
Exercises 520
13.2 ACTION SEMANTICS OF A CALCULATOR 522
Semantic Functions 523
Semantic Equations 524
A Sample Calculation 528
Exercises 530
13.3 THE DECLARATIVE FACET AND WREN 531
The Programming Language Wren 534
Exercises 540
13.4 THE REFLECTIVE FACET AND PELICAN 541
The Reflective Facet and Procedures 545
Procedures Without Parameters 547
Procedures With A Parameter 548
Recursive Definitions 550
Translating to Action Notation 551
Exercises 558
13.5 LABORATORY: TRANSLATING INTO ACTION NOTATION 559
Exercises 563
13.6 FURTHER READING 563

Appendix A
**LOGIC PROGRAMMING WITH PROLOG 565**
Prolog 566
BNF Syntax for Prolog 568
A Prolog Example 569
Predefined Predicates 571
Recursion in Prolog 572
Control Aspects of Prolog 574
Lists in Prolog 575
Sorting in Prolog 581
The Logical Variable 582
Equality and Comparison in Prolog 583
Input and Output Predicates 585

Appendix B
**FUNCTIONAL PROGRAMMING WITH SCHEME 587**
Lisp 588
Scheme Syntax 589
Functions on S-expressions 590
Lists in Scheme 591
Syntax for Functions 592
Scheme Evaluation 593
Special Forms 596
Defining Functions in Scheme 596
Recursive Definitions 598
Lambda Notation 599
Recursive Functions on Lists 599
Scope Rules in Scheme 603
Proving Correctness in Scheme 605
Higher-Order Functions 606
Currying 608
Tail Recursion 609

Bibliography 611

Index 625