Contents

Preface ix
Introduction 1

A Algorithms 11
Introduction to Part A 13

1 Specifying Algorithms 15
  1.1 Case Study: Searching an Array 15
  1.2 Declarative Interface for Array Search 18
  1.3 Assertions 19
  1.4 Completing the Specification for Array Search 25
  1.5 Correctness of Code 27
  1.6 Additional Reading 29

2 Verifying Algorithms: Basic Techniques 31
  2.1 Some Programs for Array Searching 31
  2.2 Correctness Statements 35
  2.3 Simple Assignment Statements 36
  2.4 Substitution into Assertions 38
  2.5 Using Mathematical Facts 40
  2.6 Formal Proofs and Proof Tableaux 42
  2.7 Sequencing 43
  2.8 If Statements 46
  2.9 While Statements 49
  2.10 Termination of Loops 53
  2.11 Local Variables 55
  2.12 Discussion 57
  2.13 Additional Reading 58
## Contents

3 Verifying Algorithms: Some Examples 61
   3.1 Searching an Array 61
   3.2 Minimal Entry of an Array 65
   3.3 Powers 67
   3.4 Division 69
   3.5 Binary Search in a Sorted Array 71
   3.6 The Number of Distinct Values in an Array 76
   3.7 Additional Reading 79

4 Additional Verification Techniques 81
   4.1 For Loops 81
   4.2 Array-Component Assignment Statements 83
   4.3 Do-While Loops 88
   4.4 Sorting an Array 91
   4.5 Combining Correctness Statements 99
   4.6 Additional Exercises 102
   4.7 Additional Reading 105

B Data Representations 107

Introduction to Part B 109

5 Data Representation: A Case Study 111
   5.1 Informal Specification 111
   5.2 Formal Specification 113
   5.3 A Simple Implementation 114
   5.4 Program Organization 116
   5.5 Verification of a Data Representation 119
   5.6 A Caching Implementation 122
   5.7 Additional Reading 124

6 Data Representation: Additional Examples 125
   6.1 Traversing a Sparse Set 125
   6.2 Sparse Arrays 131
   6.3 Sequences 136
   6.4 Bit-String Representation of Sets 138
   6.5 Reachability in a Directed Graph 143
   6.6 Sorting a Partially Ordered Set 146
   6.7 Additional Reading 152
## Contents

### D Unimplementable Specifications

*Introduction to Part D* 247

12 A Taste of Computability Theory 249
- 12.1 The Halting Problem 249
- 12.2 The Church-Turing Thesis 251
- 12.3 Unsolvability by Reduction 252
- 12.4 Additional Reading 254

### Appendices

A Programming Language Reference 259
- A.1 Lexical Conventions 259
- A.2 Basic Types and Constants 261
- A.3 Strings 263
- A.4 Variable Declarations 263
- A.5 Enumerated and Structure Types, Defined Type Names 264
- A.6 Expressions 264
- A.7 Some Library Functions 266
- A.8 Statements 268
- A.9 Function Definitions 271
- A.10 More Library Functions 272
- A.11 Classes 275
- A.12 Program Structure 275
- A.13 Grammar 275
- A.14 Additional Reading 280

B Hints for Selected Exercises 281

Index 287