

Contents

Preface

page ix

Part 1 Functions and Foundations

1 Introduction	3
1.1 Programming Languages	3
1.2 Goals	5
1.3 Programming Language History	6
1.4 Organization: Concepts and Languages	8
2 Computability	10
2.1 Partial Functions and Computability	10
2.2 Chapter Summary	16
Exercises	16
3 Lisp: Functions, Recursion, and Lists	18
3.1 Lisp History	18
3.2 Good Language Design	20
3.3 Brief Language Overview	22
3.4 Innovations in the Design of Lisp	25
3.5 Chapter Summary: Contributions of Lisp	39
Exercises	40
4 Fundamentals	48
4.1 Compilers and Syntax	48
4.2 Lambda Calculus	57
4.3 Denotational Semantics	67
4.4 Functional and Imperative Languages	76
4.5 Chapter Summary	82
Exercises	83

Part 2 Procedures, Types, Memory Management, and Control

5 The Algol Family and ML	93
5.1 The Algol Family of Programming Languages	93
5.2 The Development of C	99
5.3 The LCF System and ML	101
5.4 The ML Programming Language	103
5.5 Chapter Summary	121
Exercises	122

6 Type Systems and Type Inference	129
6.1 Types in Programming	129
6.2 Type Safety and Type Checking	132
6.3 Type Inference	135
6.4 Polymorphism and Overloading	145
6.5 Type Declarations and Type Equality	151
6.6 Chapter Summary	155
Exercises	156

7 Scope, Functions, and Storage Management	162
7.1 Block-Structured Languages	162
7.2 In-Line Blocks	165
7.3 Functions and Procedures	170
7.4 Higher-Order Functions	182
7.5 Chapter Summary	190
Exercises	191

8 Control in Sequential Languages	204
8.1 Structured Control	204
8.2 Exceptions	207
8.3 Continuations	218
8.4 Functions and Evaluation Order	223
8.5 Chapter Summary	227
Exercises	228

Part 3 Modularity, Abstraction, and Object-Oriented Programming

9 Data Abstraction and Modularity	235
9.1 Structured Programming	235
9.2 Language Support for Abstraction	242
9.3 Modules	252
9.4 Generic Abstractions	259
9.5 Chapter Summary	269
Exercises	271

10 Concepts in Object-Oriented Languages	277
10.1 Object-Oriented Design	277
10.2 Four Basic Concepts in Object-Oriented Languages	278

10.3	Program Structure	288
10.4	Design Patterns	290
10.5	Chapter Summary	292
10.6	Looking Forward: Simula, Smalltalk, C++, Java	293
	Exercises	294
11	History of Objects: Simula and Smalltalk	300
11.1	Origin of Objects in Simula	300
11.2	Objects in Simula	303
11.3	Subclasses and Subtypes in Simula	308
11.4	Development of Smalltalk	310
11.5	Smalltalk Language Features	312
11.6	Smalltalk Flexibility	318
11.7	Relationship between Subtyping and Inheritance	322
11.8	Chapter Summary	326
	Exercises	327
12	Objects and Run-Time Efficiency: C++	337
12.1	Design Goals and Constraints	337
12.2	Overview of C++	340
12.3	Classes, Inheritance, and Virtual Functions	346
12.4	Subtyping	355
12.5	Multiple Inheritance	359
12.6	Chapter Summary	366
	Exercises	367
13	Portability and Safety: Java	384
13.1	Java Language Overview	386
13.2	Java Classes and Inheritance	389
13.3	Java Types and Subtyping	396
13.4	Java System Architecture	404
13.5	Security Features	412
13.6	Java Summary	417
	Exercises	420
Part 4 Concurrency and Logic Programming		
14	Concurrent and Distributed Programming	431
14.1	Basic Concepts in Concurrency	433
14.2	The Actor Model	441
14.3	Concurrent ML	445
14.4	Java Concurrency	454
14.5	Chapter Summary	466
	Exercises	469

15 The Logic Programming Paradigm and Prolog	475
15.1 History of Logic Programming	475
15.2 Brief Overview of the Logic Programming Paradigm	476
15.3 Equations Solved by Unification as Atomic Actions	478
15.4 Clauses as Parts of Procedure Declarations	482
15.5 Prolog's Approach to Programming	486
15.6 Arithmetic in Prolog	492
15.7 Control, Ambivalent Syntax, and Meta-Variables	496
15.8 Assessment of Prolog	505
15.9 Bibliographic Remarks	507
15.10 Chapter Summary	507
Appendix A Additional Program Examples	509
A.1 Procedural and Object-Oriented Organization	509
<i>Glossary</i>	521
<i>Index</i>	525