

# Contents

<i>Preface to the Second Edition</i>	<i>page xv</i>
<i>Preface to the First Edition</i>	<i>xvii</i>

## Part I Judgments and Rules

1 Abstract Syntax	3
1.1 Abstract Syntax Trees	3
1.2 Abstract Binding Trees	6
1.3 Notes	10
2 Inductive Definitions	12
2.1 Judgments	12
2.2 Inference Rules	12
2.3 Derivations	14
2.4 Rule Induction	15
2.5 Iterated and Simultaneous Inductive Definitions	17
2.6 Defining Functions by Rules	18
2.7 Notes	19
3 Hypothetical and General Judgments	21
3.1 Hypothetical Judgments	21
3.2 Hypothetical Inductive Definitions	24
3.3 General Judgments	26
3.4 Generic Inductive Definitions	27
3.5 Notes	28

## Part II Statics and Dynamics

4 Statics	33
4.1 Syntax	33
4.2 Type System	34
4.3 Structural Properties	35
4.4 Notes	37

---

5	Dynamics	39
5.1	Transition Systems	39
5.2	Structural Dynamics	40
5.3	Contextual Dynamics	42
5.4	Equational Dynamics	44
5.5	Notes	46
6	Type Safety	48
6.1	Preservation	48
6.2	Progress	49
6.3	Run-Time Errors	50
6.4	Notes	52
7	Evaluation Dynamics	53
7.1	Evaluation Dynamics	53
7.2	Relating Structural and Evaluation Dynamics	54
7.3	Type Safety, Revisited	55
7.4	Cost Dynamics	56
7.5	Notes	57

### Part III Total Functions

8	Function Definitions and Values	61
8.1	First-Order Functions	61
8.2	Higher-Order Functions	62
8.3	Evaluation Dynamics and Definitional Equality	65
8.4	Dynamic Scope	66
8.5	Notes	67
9	System <b>T</b> of Higher-Order Recursion	69
9.1	Statics	69
9.2	Dynamics	70
9.3	Definability	71
9.4	Undefinability	73
9.5	Notes	75

### Part IV Finite Data Types

10	Product Types	79
10.1	Nullary and Binary Products	79
10.2	Finite Products	81
10.3	Primitive Mutual Recursion	82
10.4	Notes	83

11	Sum Types	85
11.1	Nullary and Binary Sums	85
11.2	Finite Sums	86
11.3	Applications of Sum Types	88
11.4	Notes	91

## Part V Types and Propositions

12	Constructive Logic	95
12.1	Constructive Semantics	95
12.2	Constructive Logic	96
12.3	Proof Dynamics	100
12.4	Propositions as Types	101
12.5	Notes	101
13	Classical Logic	104
13.1	Classical Logic	105
13.2	Deriving Elimination Forms	109
13.3	Proof Dynamics	110
13.4	Law of the Excluded Middle	111
13.5	The Double-Negation Translation	113
13.6	Notes	114

## Part VI Infinite Data Types

14	Generic Programming	119
14.1	Introduction	119
14.2	Polynomial Type Operators	119
14.3	Positive Type Operators	122
14.4	Notes	123
15	Inductive and Coinductive Types	125
15.1	Motivating Examples	125
15.2	Statics	128
15.3	Dynamics	130
15.4	Solving Type Equations	131
15.5	Notes	132

## Part VII Variable Types

16	System <b>F</b> of Polymorphic Types	137
16.1	Polymorphic Abstraction	137
16.2	Polymorphic Definability	140
16.3	Parametricity Overview	142
16.4	Notes	144

17	Abstract Types	146
17.1	Existential Types	146
17.2	Data Abstraction	149
17.3	Definability of Existential Types	150
17.4	Representation Independence	151
17.5	Notes	153
18	Higher Kinds	154
18.1	Constructors and Kinds	155
18.2	Constructor Equality	156
18.3	Expressions and Types	157
18.4	Notes	158

### Part VIII Partiality and Recursive Types

19	System <b>PCF</b> of Recursive Functions	161
19.1	Statics	162
19.2	Dynamics	163
19.3	Definability	165
19.4	Finite and Infinite Data Structures	167
19.5	Totality and Partiality	167
19.6	Notes	169
20	System <b>FPC</b> of Recursive Types	171
20.1	Solving Type Equations	171
20.2	Inductive and Coinductive Types	172
20.3	Self-Reference	174
20.4	The Origin of State	176
20.5	Notes	177

### Part IX Dynamic Types

21	The Untyped $\lambda$ -Calculus	181
21.1	The $\lambda$ -Calculus	181
21.2	Definability	182
21.3	Scott's Theorem	184
21.4	Untyped Means Uni-Typed	186
21.5	Notes	187
22	Dynamic Typing	189
22.1	Dynamically Typed PCF	189
22.2	Variations and Extensions	192
22.3	Critique of Dynamic Typing	194
22.4	Notes	195

23 Hybrid Typing	198
23.1 A Hybrid Language	198
23.2 Dynamic as Static Typing	200
23.3 Optimization of Dynamic Typing	201
23.4 Static versus Dynamic Typing	203
23.5 Notes	204

## **Part X Subtyping**

24 Structural Subtyping	207
24.1 Subsumption	207
24.2 Varieties of Subtyping	208
24.3 Variance	211
24.4 Dynamics and Safety	215
24.5 Notes	216
25 Behavioral Typing	219
25.1 Statics	220
25.2 Boolean Blindness	226
25.3 Refinement Safety	228
25.4 Notes	229

## **Part XI Dynamic Dispatch**

26 Classes and Methods	235
26.1 The Dispatch Matrix	235
26.2 Class-Based Organization	238
26.3 Method-Based Organization	239
26.4 Self-Reference	240
26.5 Notes	242
27 Inheritance	245
27.1 Class and Method Extension	245
27.2 Class-Based Inheritance	246
27.3 Method-Based Inheritance	248
27.4 Notes	249

## **Part XII Control Flow**

28 Control Stacks	253
28.1 Machine Definition	253
28.2 Safety	255
28.3 Correctness of the <b>K</b> Machine	256
28.4 Notes	259

29	Exceptions	260
29.1	Failures	260
29.2	Exceptions	262
29.3	Exception Values	263
29.4	Notes	264
30	Continuations	266
30.1	Overview	266
30.2	Continuation Dynamics	268
30.3	Coroutines from Continuations	269
30.4	Notes	272

### **Part XIII Symbolic Data**

31	Symbols	277
31.1	Symbol Declaration	277
31.2	Symbol References	280
31.3	Notes	282
32	Fluid Binding	284
32.1	Statics	284
32.2	Dynamics	285
32.3	Type Safety	286
32.4	Some Subtleties	287
32.5	Fluid References	288
32.6	Notes	289
33	Dynamic Classification	291
33.1	Dynamic Classes	291
33.2	Class References	293
33.3	Definability of Dynamic Classes	294
33.4	Applications of Dynamic Classification	295
33.5	Notes	296

### **Part XIV Mutable State**

34	Modernized Algol	301
34.1	Basic Commands	301
34.2	Some Programming Idioms	306
34.3	Typed Commands and Typed Assignables	307
34.4	Notes	310
35	Assignable References	313
35.1	Capabilities	313
35.2	Scoped Assignables	314

---

35.3 Free Assignables	316
35.4 Safety	318
35.5 Benign Effects	320
35.6 Notes	321
36 Lazy Evaluation	323
36.1 PCF By-Need	323
36.2 Safety of PCF By-Need	326
36.3 FPC By-Need	328
36.4 Suspension Types	329
36.5 Notes	331

## Part XV Parallelism

37 Nested Parallelism	335
37.1 Binary Fork-Join	335
37.2 Cost Dynamics	338
37.3 Multiple Fork-Join	341
37.4 Bounded Implementations	342
37.5 Scheduling	346
37.6 Notes	348
38 Futures and Speculations	350
38.1 Futures	350
38.2 Speculations	351
38.3 Parallel Dynamics	352
38.4 Pipelining with Futures	354
38.5 Notes	356

## Part XVI Concurrency and Distribution

39 Process Calculus	359
39.1 Actions and Events	359
39.2 Interaction	361
39.3 Replication	363
39.4 Allocating Channels	364
39.5 Communication	366
39.6 Channel Passing	369
39.7 Universality	371
39.8 Notes	372
40 Concurrent Algol	375
40.1 Concurrent Algol	375
40.2 Broadcast Communication	378
40.3 Selective Communication	380

---

40.4 Free Assignables as Processes	382
40.5 Notes	383
41 Distributed Algol	385
41.1 Statics	385
41.2 Dynamics	388
41.3 Safety	390
41.4 Notes	391
<b>Part XVII Modularity</b>	
42 Modularity and Linking	395
42.1 Simple Units and Linking	395
42.2 Initialization and Effects	396
42.3 Notes	398
43 Singleton Kinds and Subkinding	399
43.1 Overview	399
43.2 Singletons	400
43.3 Dependent Kinds	402
43.4 Higher Singletons	405
43.5 Notes	407
44 Type Abstractions and Type Classes	409
44.1 Type Abstraction	410
44.2 Type Classes	412
44.3 A Module Language	414
44.4 First- and Second-Class	418
44.5 Notes	419
45 Hierarchy and Parameterization	422
45.1 Hierarchy	422
45.2 Abstraction	425
45.3 Hierarchy and Abstraction	427
45.4 Applicative Functors	429
45.5 Notes	431
<b>Part XVIII Equational Reasoning</b>	
46 Equality for System T	435
46.1 Observational Equivalence	435
46.2 Logical Equivalence	439
46.3 Logical and Observational Equivalence Coincide	440
46.4 Some Laws of Equality	443
46.5 Notes	444



---

47	Equality for System <b>PCF</b>	445
47.1	Observational Equivalence	445
47.2	Logical Equivalence	446
47.3	Logical and Observational Equivalence Coincide	446
47.4	Compactness	449
47.5	Lazy Natural Numbers	452
47.6	Notes	453
48	Parametricity	454
48.1	Overview	454
48.2	Observational Equivalence	455
48.3	Logical Equivalence	456
48.4	Parametricity Properties	461
48.5	Representation Independence, Revisited	464
48.6	Notes	465
49	Process Equivalence	467
49.1	Process Calculus	467
49.2	Strong Equivalence	469
49.3	Weak Equivalence	472
49.4	Notes	473

## Part XIX Appendices

A	Background on Finite Sets	477
	<i>Bibliography</i>	479
	<i>Index</i>	487