Contents

1 Why Mathematics? 1
1.1 What Is Mathematics? 2
1.2 A Historical Perspective 2
1.3 On the Superiority of Formal Reasoning 4
1.4 The Mathematics of Computing 4
1.4.1 The Analysis of Programs 5
1.4.2 Formal Specification of Requirements 6
1.4.3 Reasoning About Programs 6
1.4.4 The Limits of Computing 7
1.4.5 A Physical Embodiment of Mathematics 8
1.5 Terminology 8

2 Propositional Logic 9
2.1 What Is a Proposition? 9
2.2 The Language of Discourse 10
2.3 Model Theory 15
2.3.1 Truth Tables: Propositional Operators 16
2.3.2 Properties of Propositional Sentences 22
2.3.3 Conclusion 28
2.4 Proof Theory 29
2.4.1 What Is a Proof? 29
2.4.2 Example Axioms and Rules of Inference 30
2.4.3 Why Proofs? 47
2.5 Comparing Truth Tables and Proofs 47
2.5.1 Completeness 48
2.5.2 Soundness 48
2.6 More About Propositional Logic 48
2.6.1 Boolean Algebra 48
2.6.2 Doing with Less 50
2.6.3 Doing with Even Less 51
2.6.4 What if the Axioms Are Wrong? 52

Bibliografische Informationen
http://d-nb.info/1021361712

digitalisiert durch DEUTSCHE NATIONAL BIBLIOTHEK

2.7 Some Notes 53
2.7.1 Notation and Symbols 53
2.7.2 Proof Elegance 54
2.7.3 How to Come up with a Proof 55
2.7.4 Not All Proofs Are Equal 55

3 Predicate Calculus 57
3.1 The Limits of Propositional Logic 57
3.2 The Language of Discourse 58
3.3 The Use of Variables 61
3.3.1 Free Variables 62
3.3.2 Substitution of Variables 64
3.4 Axiomatisation of Predicate Logic 67
3.4.1 Universal Quantification 67
3.4.2 Existential Quantification 69
3.4.3 Existential Quantification and Equality 76
3.5 Beyond Predicate Calculus 76

4 Sets 79
4.1 What Are Sets? 79
4.2 Queries About Sets 80
4.3 Comparing Sets 81
4.3.1 Subsets 81
4.3.2 Set Equality 83
4.4 Constructing Sets 84
4.4.1 Finite Sets 84
4.4.2 The Empty Set 85
4.4.3 Set Comprehensions 86
4.5 Set Operators 90
4.5.1 Set Union 92
4.5.2 Set Intersection 93
4.5.3 Set Complement 95
4.5.4 Set Difference 97
4.5.5 Other Properties of the Set Operators 98
4.5.6 Generalised Set Operators 99
4.6 Sets as a Boolean Algebra 102
4.7 More About Types 104
4.7.1 Types and Sets 104
4.7.2 On Why Types Are Desirable: Russell's Paradox 108
4.8 Summary 110

5 Relations 111
5.1 An Informal View of Relations 111
5.2 Formalising Relations 112
5.2.1 The Type of a Relation 113
5.2.2 Some Basic Relations 114

5.2.3 Relational Equality 115
5.2.4 Combining Relations as Sets 115
5.3 Domain and Range 118
5.4 Building New Relations from Old Ones 119
5.4.1 Composition of Relations 119
5.4.2 Relational Inverse 123
5.4.3 Repeated Relations 127
5.4.4 Closure 131
5.5 Other Relational Operators 134
5.6 Beyond Binary Relations 138
5.7 Summary 139

6 Classifying Relations 141
6.1 Classes of Relations 141
6.1.1 Totality 141
6.1.2 Surjectivity 143
6.1.3 Injectivity 144
6.1.4 Functionality 146
6.1.5 Combining the Results 148
6.2 Relating a Type to Itself 149
6.2.1 Properties of These Relations 149
6.2.2 Order-Inducing Relations 152
6.2.3 Equivalence Relations 153
6.3 Summary 154

7 More Discrete Structures 157
7.1 Multisets 157
7.2 Sequences 160
7.3 Graph Theory 164

8 Defining New Structured Types 175
8.1 Notions and Notation 176
8.1.1 Simple Enumerated Types 176
8.1.2 More Elaborate Types 177
8.1.3 Self-Referential Types 179
8.1.4 Parametrised Types 181
8.2 Reasoning About New Types 184
8.2.1 Axiomatising New Types 184
8.2.2 A General Inductive Principle 188
8.2.3 Structural Induction 193
8.3 Using Structured Types 194
8.3.1 Three-Valued Logic 194
8.3.2 Processing Data 198
8.3.3 Lists 202
8.3.4 Binary Trees 206
8.4 Summary 210

9 Numbers 211
9.1 Natural Numbers 211
9.1.1 Defining the Counting Numbers 212
9.1.2 Defining the Arithmetic Operators 212
9.1.3 Strong Induction 222
9.1.4 Division and Prime Numbers 224
9.2 Beyond the Natural 234
9.2.1 Integers 235
9.2.2 The Rational Numbers 237
9.2.3 The Real Numbers 237
9.3 Cardinality 239
9.3.1 Counting with Finite Sets 239
9.3.2 Extending Cardinality to Infinite Sets 249
9.4 Summary 255

10 Reasoning About Programs 257
10.1 Correctness of Algorithms 258
10.1.1 Euclid's Algorithm 259
10.1.2 Sorted Binary Trees 263
10.2 Assigning Meaning to Programs 268
10.2.1 Numeric Expressions 268
10.2.2 Program Semantics 275
10.3 The Uncomputable 280
10.3.1 Counting Computer Programs 280
10.3.2 Sets of Numbers 281
10.3.3 Parsing Languages 282
10.3.4 The Halting Problem 283
10.4 Summary 285

Index 287