

Contents

Part I Language and Logical Form

1 Atomic Statements	3
1.1 Vagueness and Ambiguity	3
1.2 Logical English	4
1.3 Names	5
1.4 Predicates	6
1.5 Descriptions	7
1.6 Atomic Statements	8
2 Compound Statements	13
2.1 Truth Functional Connectives	13
2.2 Statements with Multiple Connectives	17
2.3 Parenthesis Dropping Conventions	18
3 Quantified Statements	23
3.1 Logical Variables	23
3.2 Conditions	25
3.3 Open Descriptions	27
3.4 Quantifiers	28
4 Expressing Arguments	33
4.1 Arguments	33
4.2 Deductive and Inductive Arguments	34
4.3 Some Practical Suggestions	35
4.4 Expressing the Logical Structures of Arguments	38
5 Defining Data Structures	41
5.1 Properties of the Identity Relation	42
5.1.1 Identity is Reflexive, Symmetric, and Transitive	42
5.1.2 Leibniz's Law	42

5.2	Defining Data Structures	43
5.2.1	Sets.....	43
5.2.2	Bags	45
5.2.3	Sequences	46
5.2.4	Relations	47
5.2.5	Functions.....	48
5.2.6	Stacks.....	50
5.2.7	1-Dimensional Arrays.....	51
6	Expressing Problem Specifications	53
6.1	Functional Problem Specifications	53
6.2	What Can Go Wrong with Problem Specifications.....	55
6.3	Expressing Problem Specifications with Logical English.....	59
7	Expressing Program Designs	67
7.1	Pseudocode for Instructions.....	67
7.1.1	Atomic Instructions	68
7.1.2	Compound Instructions.....	68
7.2	Pseudocode for Algorithms	71
Part II Material Truth		
8	Truth for Statements with at Most One Connective	79
8.1	The Laws of Excluded Middle and Noncontradiction	80
8.2	Atomic Statements.....	80
8.3	Truth Functional Connectives.....	81
8.3.1	Negation.....	82
8.3.2	Conjunction.....	82
8.3.3	Disjunction: Inclusive and Exclusive	83
8.3.4	Material Equivalence	83
8.3.5	Material Implication	85
8.4	Conditions	86
8.5	Quantified Statements.....	87
8.6	Summary of Material Truth Conditions	88
8.7	Some Applications.....	89
8.7.1	Short Cut Evaluation: “cand” and “cor”.....	89
8.7.2	Bitwise Extensions of Truth Functions	90
9	Truth for Statements with Multiple Connectives	95
9.1	Compound Statements	95
9.1.1	Computing Truth Values of Compound Statements.....	96
9.1.2	Reducing the Need for Parentheses.....	98
9.2	How to Make Truth Tables for Compound Statements.....	100

- 9.2.1 The Four Parts of a Truth Table 100
- 9.2.2 Organizing a Truth Table Calculation 101
- 9.3 Reading Truth Tables from Right to Left 103
- 10 Tracing Program Execution..... 105**
 - 10.1 Tracing Program Execution Forwards 105
 - 10.1.1 Event Trace Tables 106
 - 10.1.2 Value Trace Tables 107
 - 10.1.3 Tracing with Complex Conditions 108
 - 10.2 Tracing Program Execution Backwards 109
- Part III Logical Truth**
- 11 Truth Functional Forms 115**
 - 11.1 Overview 115
 - 11.2 Truth Functional Forms 117
 - 11.3 Interpretations of Truth Functional Forms 118
 - 11.4 Truth Under an Interpretation 119
 - 11.5 Truth Functional Truth, Falsity, and Contingency 119
 - 11.6 Using Forms to Find the Logical Status of English Statements 121
 - 11.7 Application to Simplifying Statements and Conditions 124
- 12 Truth Functional Properties of Program Designs 129**
 - 12.1 The General Form of Decision Tables..... 129
 - 12.2 Limited Entry Decision Tables 130
 - 12.2.1 The Basic Form 130
 - 12.2.2 Simplifying Tables with “Don’t Care”Condition Entries.. 131
 - 12.3 Extended Entry Decision Tables..... 133
 - 12.4 Decision Tables and Other Control Structures 135
 - 12.5 Consistency, Completeness, and Redundancy of Decision Tables 136
- 13 Quantified Forms 139**
 - 13.1 How Truth Functional Logic Differs from Quantificational Logic 139
 - 13.2 Well Formed Forms 140
 - 13.3 Interpretations of Wffs..... 142
 - 13.4 Truth of Wffs in an Interpretation 143
 - 13.5 Logical Truth of Wffs 145
 - 13.6 Using Wffs to Determine the Logical Status of English Statements..... 148
 - 13.7 Conditions Revisited 149
 - 13.8 Summary of Classifications 151

- 14 Logical Equivalence** 153
 - 14.1 Truth Functional Equivalence 154
 - 14.2 Applications of Truth Functional Equivalence 155
 - 14.2.1 Equivalence and Simplification of Program Instructions 155
 - 14.2.2 Equivalence and Simplification of SQL Select Instructions 158
 - 14.3 Logical Equivalence 159
 - 14.3.1 Logical Equivalence of Wffs 159
 - 14.3.2 Logical Equivalence of Conditions 160
 - 14.4 Applications of Logical Equivalence 163
 - 14.4.1 Simplification of Problem Specifications 163
 - 14.4.2 Detecting Redundancy 163

- 15 Logical Implication and Validity** 165
 - 15.1 Logical Implication and Validity 166
 - 15.1.1 Logical Implication 166
 - 15.1.2 Logical Validity 166
 - 15.1.3 Soundness 166
 - 15.2 Determining Validity 169
 - 15.2.1 Degrees of Formality 169
 - 15.2.2 Arguments and Corresponding Conditionals 170
 - 15.2.3 Truth Functional Validity 171
 - 15.2.4 Logical Validity 177

- 16 Rules of Inference** 181
 - 16.1 Limitations of Truth Table Tests 181
 - 16.2 Rules of Inference 182
 - 16.2.1 Formal Rules of Inference 182
 - 16.2.2 Correct and Incorrect Rules of Inference 182
 - 16.3 Some Truth Functional Rules of Inference 183
 - 16.3.1 Examples of Correct Rules of Inference 183
 - 16.3.2 Some Properties of Rules of Inference 185
 - 16.3.3 Examples of Incorrect Rules of Inference (Formal Fallacies) 188

- 17 Proof** 189
 - 17.1 Kinds of Proof 190
 - 17.1.1 Different Senses of Proof 190
 - 17.1.2 Formal vs Informal Proofs 190
 - 17.2 Two Ways of Organizing Proofs 191

- 17.3 Three Proof Strategies 193
 - 17.3.1 Direct Proof 193
 - 17.3.2 Conditional Proof 195
 - 17.3.3 Indirect Proof 196
- 17.4 Applying Logic to Specific Subjects 198
 - 17.4.1 Using Instances Of General Forms 198
 - 17.4.2 Adding Nonlogical Justifications 199
- 18 Algorithmic Unsolvability Proofs 203**
 - 18.1 Algorithmic Solvability and Unsolvability 203
 - 18.2 The Halting Problem is Algorithmically Unsolvable 207
 - 18.3 Other Algorithmically Unsolvable Problems 208
- 19 Program Correctness Proofs 211**
 - 19.1 The Limits of Testing 211
 - 19.2 Proofs Expressed in English Prose 212
 - 19.3 Proofs Using Floyd’s Method of Invariant Assertions 216
 - 19.4 Rules of Inference Involving Algorithms 219
 - 19.4.1 Notation for Statements Involving Instructions 220
 - 19.4.2 The Assignment Rule 220
 - 19.4.3 Rules of Inference about Control Structures 222
 - 19.5 Proofs Using Rules of Inference 224
- 20 Above and Beyond this Book 233**
 - 20.1 Other Texts on Classical Logic 234
 - 20.1.1 Standard Texts 234
 - 20.1.2 Computer Oriented Texts 234
 - 20.2 Extensions of Classical Logic 234
 - 20.2.1 Floyd-Hoare Logic 234
 - 20.2.2 Temporal, Modal, and Dynamic Logics 235
 - 20.3 Rivals to Classical Logic 235
 - 20.4 Applications of Logic to Computing or Computing to Logic 235
 - 20.4.1 Logic Circuits 235
 - 20.4.2 3-Valued Logic and SQL 235
 - 20.4.3 More on Expressing and Using Specifications 235
 - 20.4.4 Logic Testing of Software 236
 - 20.4.5 Computability and Algorithmic Unsolvability 236
 - 20.4.6 AI and Computer Aided Reasoning 236
 - 20.4.7 Quantum Computing and Programming 237
 - 20.5 Programming Challenges 237

Solutions to Selected Exercises	241
Chapter 1	241
Chapter 2	243
Chapter 3	246
Chapter 4	249
Chapter 5	251
Chapter 6	254
Chapter 7	257
Chapter 8	259
Chapter 9	261
Chapter 10	265
Chapter 11	270
Chapter 12	274
Chapter 13	277
Chapter 14	280
Chapter 15	286
Chapter 16	291
Chapter 17	293
Chapter 18	299
Chapter 19	300
Sources and Bibliography	305
Index	309



<http://www.springer.com/978-1-84800-081-0>

Elementary Logic
For Software Development
Lover, R.
2008, XIV, 314 p., Softcover
ISBN: 978-1-84800-081-0