

Contents

1 Introduction	1
Overview	1
What This Book Is About	1
What This Book Tries to Do	5
What This Book Tries Not to Do	6
The Exercises	7
Further Reading	7
Some Advice	7

Part 1 Languages and Machines

2 Elements of Formal Languages	11
Overview	11
Alphabets	11
Strings	12
Functions that Apply to Strings	12
Useful Notation for Describing Strings	13
Formal Languages	14
Methods for Defining Formal Languages	15
Set Definitions of Languages	15
Decision Programs for Languages	17
Rules for Generating Languages	19
Formal Grammars	22
Grammars, Derivations, and Languages	22
The Relationship Between Grammars and Languages	26
Phrase Structure Grammars and the Chomsky Hierarchy	26
Formal Definition of PSGs	26
Derivations, Sentential Forms, Sentences, and " $L(G)$ "	27
The Chomsky Hierarchy	30
A Type 0 Grammar: Computation as Symbol Manipulation	33
Exercises	35

3 Syntax, Semantics, and Ambiguity	37
Overview	37
Syntax Vs. Semantics	37
Derivation Trees	38
Parsing	39
Ambiguity	42
Exercises	45
4 Regular Languages and Finite State Recognisers	49
Overview	49
Regular Grammars	49
Some Problems with Grammars	51
Finite State Recognisers and Finite State Generators	51
Creating an FSR	51
The Behaviour of the FSR	51
The FSR as Equivalent to the Regular Grammar	56
Non-determinism in FSRs	59
Constructing Deterministic FSRs	61
The DFSR as Equivalent to the Non-DFSR	63
A Simple Deterministic Decision Program	68
Minimal FSRs	68
Constructing a Minimal FSR	70
Why Minimisation Works	73
The General Equivalence of Regular Languages and FSRs	76
Observations on Regular Grammars and Languages	77
Exercises	78
5 Context Free Languages and Pushdown Recognisers	81
Overview	81
Context Free Grammars and Context Free Languages	81
Changing G Without Changing $L(G)$	82
The Empty String (ϵ)	82
Chomsky Normal Form	85
Pushdown Recognisers	89
The Stack	90
Constructing a Non-deterministic PDR	92
Example NPDRs, M_3 and M_{10}	93
Deterministic PDRs	96
M_3^d , a Deterministic Version of M_3	97
More DPDRs	99
Deterministic and Non-deterministic CFLs	99
Every Regular Language Is a Deterministic CFL	100
The Non-deterministic CFLs	101
A Refinement to the Chomsky Hierarchy in the	
Case of CFLs	103
The Equivalence of the CFLs and the PDRs	103

Observations on CFGs and CFLs	104
Exercises	105
6 Important Features of Regular and Context Free Languages	107
Free Languages	107
Overview	107
Closure Properties of Languages	107
Closure Properties of the Regular Languages	108
Complement	108
Union	109
Intersection	110
Concatenation	112
Closure Properties of the Context Free Languages	112
Union	113
Concatenation	115
Intersection	116
Complement	117
Chomsky's Hierarchy Is Indeed a Proper Hierarchy	118
The "Repeat State Theorem"	118
A Language that Is Context Free but Not Regular	120
The " $uvwx$ y" Theorem for CFLs	122
$\{a^i b^j c^k : i \geq 1\}$ Is Not Context Free	128
The "Multiplication Language" Is Not Context Free	129
Preliminary Observations on the Scope of the Chomsky Hierarchy	130
Exercises	131
7 Phrase Structure Languages and Turing Machines	133
Overview	133
The Architecture of the Turing Machine	133
"Tapes" and the "Read/Write Head"	133
Blank Squares	134
TM "Instructions"	135
TMs Defined	136
The Behaviour of a TM	136
TMs as Language Recognisers	139
Regular Languages	139
Context Free Languages	140
TMs Are More Powerful than PDRs	143
Introduction to (TM) Computable Languages	145
The TM as the Recogniser for the Context Sensitive Languages	146
Constructing a Non-deterministic TM for Reduction	
Parsing of a Context Sensitive Language	146
The Generality of the Construction	150
The TM as the Recogniser for the Type 0 Languages	152

Amending the Reduction Parsing TM to Deal with Type 0 Productions	152
Dealing with the Empty String	153
The TM as the Recogniser for All Types in the Chomsky Hierarchy	155
Decidability: A Preliminary Discussion	155
<i>Deciding</i> a Language	155
<i>Accepting</i> a Language	157
End of Part 1	158
Exercises	159
 Part 2 Machines and Computation	
8 Finite State Transducers	163
Overview	163
Finite State Transducers	163
FSTs and Language Recognition	164
FSTs and Memory	164
FSTs and Computation	167
Simple Multiplication	167
Addition and Subtraction	168
Simple Division and Modular Arithmetic	171
The Limitations of the FST	172
Restricted FST Multiplication	173
FSTs and Unlimited Multiplication	176
FSTs as Unsuitable Models for Real Computers	176
Exercises	178
 9 Turing Machines as Computers	 179
Overview	179
Turing Machines and Computation	179
TMs and Arbitrary Binary Multiplication	180
Some Basic TM Operations	180
The “ADD” TM	181
The “MULT” TM	185
TMs and Arbitrary Integer Division	190
The “SUBTRACT” TM	191
The “DIV” TM	193
Logical Operations	194
TMs and the Simulation of Computer Operations	196
Exercises	199
 10 Turing’s Thesis and the Universality of the Turing Machine	 203
Overview	203

Turing's Thesis	203
Coding a TM and Its Tape as a Binary Number	205
Coding Any TM	206
Coding the Tape	208
The Universal Turing Machine	208
<i>UTM</i> 's Tapes	209
The Operation of <i>UTM</i>	211
Some Implications of <i>UTM</i>	211
Non-deterministic TMs	213
Converting a Non-deterministic TM into a 4-tape	
Deterministic TM	214
The Four Tapes of the Deterministic Machine, <i>D</i>	214
The Systematic Generation of the Strings of	
Quintuple Labels	216
The Operation of <i>D</i>	222
The Equivalence of Non-deterministic and	
Four-tape Deterministic TMs	222
Converting a Multi-tape TM into a Single-tape TM	222
Example: Representing Three Tapes as One	223
The Operation of the Single-tape Machine, <i>S</i>	226
The Equivalence of Deterministic Multi-tape	
and Deterministic Single-tape TMs	227
The Linguistic Implications of the Equivalence of	
Non-deterministic and Deterministic TMs	228
Exercises	228
 11 Computability, Solvability, and the Halting Problem	 231
Overview	231
The Relationship Between Functions, Problems,	
Solvability, and Decidability	231
Functions and Computability	231
Problems and Solvability	232
Decision Problems and Decidability	234
The Halting Problem	234
<i>UTM_H</i> Partially Solves the Halting Problem	235
<i>Reductio ad Absurdum</i> Applied to the Halting	
Problem	235
The Halting Problem Shown to Be Unsolvable	239
Some Implications of the Unsolvability of the	
Halting Problem	240
Computable Languages	242
An Unacceptable (Non-computable) Language	243
An Acceptable, but Undecidable, Language	245
Languages and Machines	245
Exercises	247

12 Dimensions of Computation	249
Overview	249
Aspects of Computation: Space, Time, and Complexity	249
Non-deterministic TMs Viewed as Parallel Processors	251
Parallel Computations and Time	252
A Brief Look at an Unsolved Problem of Complexity	254
A Beginner's Guide to the "Big O"	254
Predicting the Running Time of Algorithms	255
Linear Time	256
Logarithmic Time	257
Polynomial Time	260
Exponential Time	267
The Implications of Exponential Time Processes	268
Is P Equal to NP?	269
Observations on the Efficiency of Algorithms	270
End of Part 2	271
Exercises	271

Part 3 Computation and Logic

13 Boolean Logic and Propositional Logic	275
Overview	275
Boolean Logic	275
Boolean Logic Operators	275
Boolean Logic for Problem Solving	276
Boolean Logic and Computing	278
Propositional Logic	281
Propositions	281
Implication and Equivalence	282
Rules of Inference	284
Problem Solving and Reasoning in Propositional Logic	285
Using Truth Tables to Prove Things in Propositional Logic	286
Observations on Propositional Logic	288
Exercises	289
14 First Order Predicate Logic	291
Overview	291
Predicate Logic	291
Predicates	292
Functions	293
"Sentences" Revisited: Well-formed Formulae	294
The "First Orderness" of First Order Logic	295

Quantifiers	296
The Existential Quantifier	296
The Universal Quantifier	298
A “Blocks World” Example of FOPL Representation	299
The Semantics of FOPL: <i>Interpretation</i>	300
Problem Solving and Inference in FOPL	301
Rules of Inference for FOPL	301
Solving Problems by Classical FOPL Reasoning	303
The Nature of FOPL	304
Conclusions	305
Exercises	305
15 Logic and Computation	307
Overview	307
A Computational Form of FOPL	307
Getting Rid of \forall	308
Getting Rid of \exists	309
Conjunctive Normal Form Databases	312
Resolution	314
The Role of Unification in Resolution	314
How to Do Resolution	316
The Efficiency of Resolution	322
Why Resolution Works	322
Logic in Action	323
Languages, Machines, and Logic	325
Exercises	326
Solutions to Selected Exercises	327
Chapter 2	327
Chapter 3	329
Chapter 4	329
Chapter 5	331
Chapter 6	333
Chapter 7	333
Chapter 8	333
Chapter 9	334
Chapter 10	334
Chapter 11	334
Chapter 12	335
Chapter 13	337
Chapter 14	338
Chapter 15	338
Further Reading	341
Index	343

Introduction to Languages, Machines and Logic
Computable Languages, Abstract Machines and Formal
Logic

Parkes, A.P.

2002, XI, 351 p., Softcover

ISBN: 978-1-85233-464-2