CAMBRIDGE
Cambridge University Press & Assessment
978-0-521-84887-9 — Automata Theory with Modern Applications
James A. Anderson
Table of Contents
More Information

Contents

	Preface	*page* vii
1	**Introduction**	1
1.1	Sets	1
1.2	Relations	6
1.3	Functions	12
1.4	Semigroups	16
2	**Languages and codes**	23
2.1	Regular languages	23
2.2	Retracts (Optional)	29
2.3	Semiretracts and lattices (Optional)	34
3	**Automata**	37
3.1	Deterministic and nondeterministic automata	37
3.2	Kleene's Theorem	52
3.3	Minimal deterministic automata and syntactic monoids	72
3.4	Pumping Lemma for regular languages	84
3.5	Decidability	86
3.6	Pushdown automata	89
3.7	Mealy and Moore machines	99
4	**Grammars**	114
4.1	Formal grammars	114
4.2	Chomsky normal form and Greibach normal form	138
4.3	Pushdown automata and context-free languages	149
4.4	The Pumping Lemma and decidability	162
5	**Turing machines**	169
5.1	Deterministic Turing machines	169

© in this web service Cambridge University Press & Assessment www.cambridge.org

CAMBRIDGE
Cambridge University Press & Assessment
978-0-521-84887-9 — Automata Theory with Modern Applications
James A. Anderson
Table of Contents
More Information

5.2 Nondeterministic Turing machines and acceptance of context-free languages 190
5.3 The halting problem for Turing machines 194
5.4 Undecidability problems for context-free languages 200

6 A visual approach to formal languages 210
6.1 Introduction 210
6.2 A minimal taste of word combinatorics 212
6.3 The spectrum of a word with respect to a language 214
6.4 The spectral partition of Σ^+ and the support of L 216
6.5 Visualizing languages 217
6.6 The sketch parameters of a language 221
6.7 Flag languages 223
6.8 Additional tools from word combinatorics 224
6.9 Counting primitive words in a regular language 225
6.10 Algorithmic sketching of regular languages 227

7 From biopolymers to formal language theory 231
7.1 Introduction 231
7.2 Constructing new words by splicing together pairs of existing words 232
7.3 The motivation from molecular biology 233
7.4 Splicing rules, schemes, systems, and languages 236
7.5 Every splicing language is a regular language 239
7.6 The syntactic monoid of a regular language L allows an effective determination of the set of all splicing rules that respect L 241
7.7 It is algorithmically decidable whether a given regular language is a reflexive splicing language 242

Appendix A Cardinality 245

Appendix B Co-compactness Lemma 247

References 249
Further reading 251
Index 253

© in this web service Cambridge University Press & Assessment www.cambridge.org