UNDERGRADUATE TEXTS IN COMPUTER SCIENCE

Editors
David Gries
Fred B. Schneider

UNDERGRADUATE TEXTS IN COMPUTER SCIENCE

Beidler, Data Structures and Algorithms

Bergin, Data Structure Programming

Brooks, C Programming: The Essentials for Engineers and Scientists

Brooks, Problem Solving with Fortran 90

Dandamudi, Introduction to Assembly Language Programming

Grillmeyer, Exploring Computer Science with Scheme

Jalote, An Integrated Approach to Software Engineering, Second Edition

Kizza, Ethical and Social Issues in the Information Age

Kozen, Automata and Computability

Merritt and Stix, Migrating from Pascal to C++

Pearce, Programming and Meta-Programming in Scheme

Zeigler, Objects and Systems

Dexter C. Kozen

Automata and Computability

Springer

Dexter C. Kozen
Department of Computer Science
Cornell University
Ithaca, NY 14853-7501
USA

Series Editors
David Gries
Department of Computer Science
415 Boyd Studies Research Center
The University of Georgia
Athens, Georgia 30602
USA

Fred B. Schneider
Department of Computer Science
Cornell University
4115C Upson Hall
Ithaca, NY 14853-7501
USA

On the cover: **Cover photo taken by John Still/Photonica.**

With 1 figure.

Library of Congress Cataloging-in-Publication Data
Kozen, Dexter, 1951-
Automata and computability/Dexter C. Kozen.
p. cm. — (Undergraduate texts in computer science)
Includes bibliographical references and index.
ISBN 978-1-4612-7309-7 ISBN 978-1-4612-1844-9 (eBook)
DOI 10.1007/978-1-4612-1844-9
1. Machine theory. 2. Computable functions. I. Title.
II. Series.
QA267.K69 1997
511.3—dc21 96-37409

Printed on acid-free paper.

9 8 7

ISBN 978-1-4612-7309-7

springeronline.com

To Juris

Preface

These are my lecture notes from CS381/481: Automata and Computability Theory, a one-semester senior-level course I have taught at Cornell University for many years. I took this course myself in the fall of 1974 as a first-year Ph.D. student at Cornell from Juris Hartmanis and have been in love with the subject ever since.

The course is required for computer science majors at Cornell. It exists in two forms: CS481, an honors version; and CS381, a somewhat gentler-paced version. The syllabus is roughly the same, but CS481 goes deeper into the subject, covers more material, and is taught at a more abstract level. Students are encouraged to start off in one or the other, then switch within the first few weeks if they find the other version more suitable to their level of mathematical skill.

The purpose of the course is twofold: to introduce computer science students to the rich heritage of models and abstractions that have arisen over the years; and to develop the capacity to form abstractions of their own and reason in terms of them.

The course is quite mathematical in flavor, and a certain degree of previous mathematical experience is essential for survival. Students should already be conversant with elementary discrete mathematics, including the notions of set, function, relation, product, partial order, equivalence relation, graph, and tree. They should have a repertoire of basic proof techniques at their disposal, including a thorough understanding of the principle of mathematical induction.

The material covered in this text is somewhat more than can be covered in a one-semester course. It is also a mix of elementary and advanced topics. The basic course consists of the lectures numbered 1 through 39. Additionally, I have included several supplementary lectures numbered A through K on various more advanced topics. These can be included or omitted at the instructor's discretion or assigned as extra reading. They appear in roughly the order in which they should be covered.

At first these notes were meant to supplement and not supplant a textbook, but over the years they gradually took on a life of their own. In addition to the notes, I depended on various texts at one time or another: Cutland [30], Harrison [55], Hopcroft and Ullman [60], Lewis and Papadimitriou [79], Machtey and Young [81], and Manna [82]. In particular, the Hopcroft and Ullman text was the standard textbook for the course for many years, and for me it has been an indispensable source of knowledge and insight. All of these texts are excellent references, and I recommend them highly.

In addition to the lectures, I have included 12 homework sets and several miscellaneous exercises. Some of the exercises come with hints and/or solutions; these are indicated by the annotations "H" and "S," respectively. In addition, I have annotated exercises with zero to three stars to indicate relative difficulty.

I have stuck with the format of my previous textbook [72], in which the main text is divided into more or less self-contained lectures, each 4 to 8 pages. Although this format is rather unusual for a textbook, I have found it quite successful. Many readers have commented that they like it because it partitions the subject into bite-sized chunks that can be covered more or less independently.

I owe a supreme debt of gratitude to my wife Frances for her constant love, support, and superhuman patience, especially during the final throes of this project. I am also indebted to the many teachers, colleagues, teaching assistants, and students who over the years have shared the delights of this subject with me and from whom I have learned so much. I would especially like to thank Rick Aaron, Arash Baratloo, Jim Baumgartner, Steve Bloom, Manuel Blum, Amy Briggs, Ashok Chandra, Wilfred Chen, Allan Cheng, Francis Chu, Bob Constable, Devdatt Dubhashi, Peter van Emde Boas, Allen Emerson, András Ferencz, Jeff Foster, Sophia Georgiakaki, David Gries, Joe Halpern, David Harel, Basil Hayek, Tom Henzinger, John Hopcroft, Nick Howe, Doug Ierardi, Tibor Janosi, Jim Jennings, Shyam Kapur, Steve Kautz, Nils Klarlund, Peter Kopke, Vladimir Kotlyar, Alan Kwan, Georges Lauri, Michael Leventon, Jake Levirne, David Liben-Nowell, Yvonne Lo, Steve Mahaney, Nikolay Mateev, Frank McSherry, Albert Meyer, Bob Milnikel, Francesmary Modugno, Anil Nerode, Damian Niwiński, David de la Nuez, Dan Oberlin, Jens Palsberg, Rohit

Parikh, David Pearson, Paul Pedersen, Vaughan Pratt, Zulfikar Ramzan, Jon Rosenberger, Jonathan Rynd, Erik Schmidt, Michael Schwartzbach, Amitabh Shah, Frederick Smith, Kjartan Stefánsson, Colin Stirling, Larry Stockmeyer, Aaron Stump, Jurek Tiuryn, Alex Tsow, Moshe Vardi, Igor Walukiewicz, Rafael Weinstein, Jim Wen, Dan Wineman, Thomas Yan, Paul Zimmons, and many others too numerous to mention. Of course, the greatest of these is Juris Hartmanis, whose boundless enthusiasm for the subject is the ultimate source of my own.

I would be most grateful for suggestions and criticism from readers.

Note added for the third printing. I am indebted to Chris Jeuell for pointing out several typographical errors, which have been corrected in this printing.

Ithaca, New York   Dexter C. Kozen

Contents

Pushdown Automata and Context-Free Languages

Turing Machines and Effective Computability

Homework Sets

Miscellaneous Exercises

Hints and Solutions