

# Concrete Abstractions

Out of print; full text available for free at <http://www.gustavus.edu/+max/concrete-abstracts.html>

# Concrete Abstractions

## An Introduction to Computer Science Using Scheme

Max Hailperin  
*Gustavus Adolphus College*

Barbara Kaiser  
*Gustavus Adolphus College*

Karl Knight  
*Gustavus Adolphus College*



An Imprint of Brooks/Cole Publishing Company

**I(TP)<sup>®</sup>** An International Thomson Publishing Company

---

Pacific Grove • Albany • Belmont • Bonn • Boston • Cincinnati • Detroit  
Johannesburg • London • Madrid • Melbourne • Mexico City • New York  
Paris • Singapore • Tokyo • Toronto • Washington

Sponsoring Editor: *Suzanne Jeans*  
Marketing Team: *Nathan Wilbur, Michele Mootz*  
Editorial Assistant: *Kathryn Schooling*  
Production Manager: *Marlene Thom*  
Manuscript Editor: *Linda Thompson*  
Interior Design: *Merry Obrecht Sawdey*

Cover Design: *Vernon T. Boes*  
Cover Photo: *Craig Cowan. Courtesy of Couturier Gallery, Los Angeles, CA*  
Project Management and Typesetting: *Integre Technical Publishing Co., Inc.*  
Printing and Binding: *Webcom, Ltd.*

COPYRIGHT © 1999 by Brooks/Cole Publishing Company  
A division of International Thomson Publishing Inc.

ITP The ITP logo is a registered trademark used herein under license.

Java™ is a trademark of Sun Microsystems, Inc. All other products used herein are used for identification purposes only, and may be trademarks or registered trademarks of their respective owners.

*For more information, contact:*

BROOKS/COLE PUBLISHING COMPANY  
511 Forest Lodge Road  
Pacific Grove, CA 93950  
USA

International Thomson Publishing Europe  
Berkshire House 168-173  
High Holborn  
London WC1V 7AA  
England

Thomas Nelson Australia  
102 Dodds Street  
South Melbourne, 3205  
Victoria, Australia

Nelson Canada  
1120 Birchmount Road  
Scarborough, Ontario  
Canada M1K 5G4

International Thomson Editores  
Seneca 53  
Col. Polanco  
11560 México, D.F., México

International Thomson Publishing GmbH  
Königswinterer Strasse 418  
53227 Bonn  
Germany

International Thomson Publishing Asia  
221 Henderson Road  
#05-10 Henderson Building  
Singapore 0315

International Thomson Publishing Japan  
Hirakawacho Kyowa Building, 3F  
2-2-1 Hirakawacho  
Chiyoda-ku, Tokyo 102  
Japan

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transcribed, in any form or by any means— electronic, mechanical, photocopying, recording, or otherwise—without the prior written permission of the publisher, Brooks/Cole Publishing Company, Pacific Grove, California 93950.

Printed in Canada

10 9 8 7 6 5 4 3 2 1

#### Library of Congress Cataloging in Publication Data

Hailperin, Max.

Concrete abstractions : an introduction to computer science using  
Scheme / Max Hailperin, Barbara Kaiser, Karl Knight.

p. cm.

ISBN 0-534-95211-9 (alk. paper)

1. Computer science. 2. Abstract data types (Computer science)  
I. Kaiser, Barbara (Barbara K.). II. Knight, Karl (Karl  
W.). III. Title.

QA76.H296 1997

005.13'3—dc21

98-34080

CIP



# Contents

Preface	ix
PART I Procedural Abstraction	
CHAPTER 1	
Computer Science and Programming	3
1.1 What's It All About? / 3	
<i>Sidebar: Responsible Computer Use / 5</i>	
1.2 Programming in Scheme / 5	
1.3 An Application: Quilting / 15	
CHAPTER 2	
Recursion and Induction	22
2.1 Recursion / 22	
<i>Sidebar: Exponents / 28</i>	
2.2 Induction / 28	
2.3 Further Examples / 34	
2.4 An Application: Custom-Sized Quilts / 40	
CHAPTER 3	
Iteration and Invariants	48
3.1 Iteration / 48	
3.2 Using Invariants / 54	
3.3 Perfect Numbers, Internal Definitions, and Let / 58	
3.4 Iterative Improvement: Approximating the Golden Ratio / 61	
3.5 An Application: The Josephus Problem / 65	

CHAPTER 4	
Orders of Growth and Tree Recursion	75
4.1 Orders of Growth / 75	
<i>Sidebar: Selection Sort / 77</i>	
<i>Sidebar: Merge Sort / 78</i>	
<i>Sidebar: Merging / 79</i>	
<i>Sidebar: Logarithms / 82</i>	
4.2 Tree Recursion and Digital Signatures / 83	
<i>Sidebar: Modular Arithmetic / 87</i>	
4.3 An Application: Fractal Curves / 95	
CHAPTER 5	
Higher-Order Procedures	109
5.1 Procedural Parameters / 109	
5.2 Uncomputability / 113	
<i>Sidebar: Alan Turing / 116</i>	
5.3 Procedures That Make Procedures / 118	
5.4 An Application: Verifying ID Numbers / 120	
PART II Data Abstraction	
CHAPTER 6	
Compound Data and Data Abstraction	133
6.1 Introduction / 133	
6.2 Nim / 135	
6.3 Representations and Implementations / 143	
<i>Sidebar: Nim Program / 144</i>	
<i>Sidebar: Game State ADT Implementation / 152</i>	
6.4 Three-Pile Nim / 153	
6.5 An Application: Adding Strategies to Nim / 156	
<i>Sidebar: Type Checking / 157</i>	
CHAPTER 7	
Lists	167
7.1 The Definition of a List / 167	
7.2 Constructing Lists / 169	
7.3 Basic List Processing Techniques / 172	
7.4 List Processing and Iteration / 179	
7.5 Tree Recursion and Lists / 182	

7.6	An Application: A Movie Query System / 187	
	<i>Sidebar: Is There More to Intelligence Than the Appearance of Intelligence?</i> / 202	
CHAPTER 8		
	Trees	212
8.1	Binary Search Trees / 212	
8.2	Efficiency Issues with Binary Search Trees / 220	
	<i>Sidebar: Privacy Issues</i> / 225	
8.3	Expression Trees / 226	
8.4	An Application: Automated Phone Books / 229	
CHAPTER 9		
	Generic Operations	243
9.1	Introduction / 243	
9.2	Multiple Representations / 244	
9.3	Exploiting Commonality / 253	
9.4	An Application: Computer Graphics / 262	
CHAPTER 10		
	Implementing Programming Languages	278
10.1	Introduction / 278	
10.2	Syntax / 279	
	<i>Sidebar: The Expressiveness of EBNF</i> / 285	
10.3	Micro-Scheme / 289	
10.4	Global Definitions: Mini-Scheme / 303	
10.5	An Application: Adding Explanatory Output / 311	
PART III Abstractions of State		
CHAPTER 11		
	Computers with Memory	333
11.1	Introduction / 333	
11.2	An Example Computer Architecture / 333	
11.3	Programming the SLIM / 340	
	<i>Sidebar: What Can Be Stored in a Location?</i> / 342	
	<i>Sidebar: SLIM's Instruction Set</i> / 348	
11.4	Iteration in Assembly Language / 349	
11.5	Recursion in Assembly Language / 357	
11.6	Memory in Scheme: Vectors / 361	
11.7	An Application: A Simulator for SLIM / 367	

CHAPTER 12	
Dynamic Programming	379
12.1 Introduction / 379	
12.2 Revisiting Tree Recursion / 380	
12.3 Memoization / 388	
12.4 Dynamic Programming / 398	
12.5 Comparing Memoization and Dynamic Programming / 406	
12.6 An Application: Formatting Paragraphs / 406	
CHAPTER 13	
Object-based Abstractions	420
13.1 Introduction / 420	
13.2 Arithmetic Expressions Revisited / 421	
13.3 RA-Stack Implementations and Representation Invariants / 432	
<i>Sidebar: Strings and Characters / 433</i>	
13.4 Queues / 446	
13.5 Binary Search Trees Revisited / 453	
13.6 Dictionaries / 472	
CHAPTER 14	
Object-oriented Programming	486
14.1 Introduction / 486	
14.2 An Object-oriented Program / 487	
14.3 Extensions and Variations / 511	
14.4 Implementing an Object-oriented Prog. System / 517	
14.5 An Application: Adventures in the Land of Gack / 543	
CHAPTER 15	
Java, Applets, and Concurrency	577
15.1 Introduction / 577	
15.2 Java / 578	
15.3 Event-Driven Graphical User Interfaces in Applets / 599	
15.4 Concurrency / 616	
<i>Sidebar: Nested Calls to Synchronized Methods and Deadlock / 625</i>	
15.5 An Application: Simulating Compound Interest / 632	
APPENDIX Nonstandard Extensions to Scheme	645
Bibliography	649
Index	653