



Java Challenges

100+ Proven Tasks that Will
Prepare You for Anything

—

Michael Inden

Apress®

Java Challenges

100+ Proven Tasks that Will Prepare
You for Anything

Michael Inden

Apress®

Java Challenges: 100+ Proven Tasks that Will Prepare You for Anything

Michael Inden
Zurich, Switzerland

ISBN-13 (pbk): 978-1-4842-7394-4
<https://doi.org/10.1007/978-1-4842-7395-1>

ISBN-13 (electronic): 978-1-4842-7395-1

Copyright © 2022 by Michael Inden

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director, Apress Media LLC: Welmoed Spahr
Acquisitions Editor: Steve Anglin
Development Editor: Matthew Moodie
Coordinating Editor: Mark Powers
Copyeditor: Mary Behr

Cover designed by eStudioCalamar

Cover image by Michael Inden

Distributed to the book trade worldwide by Apress Media, LLC, 1 New York Plaza, New York, NY 10004, U.S.A. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a **Delaware** corporation.

For information on translations, please e-mail booktranslations@springernature.com; for reprint, paperback, or audio rights, please e-mail bookpermissions@springernature.com.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at www.apress.com/bulk-sales.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at www.apress.com/9781484273944. For more detailed information, please visit www.apress.com/source-code.

Printed on acid-free paper

Table of Contents

About the Author	xvii
About the Technical Reviewer	xix
Preface	xxi
Chapter 1: Introduction.....	1
1.1 Structure of the Chapters.....	1
1.1.1 Introduction	1
1.1.2 Exercises	1
1.1.3 Solutions.....	2
1.2 Basic Structure of the Eclipse Project.....	3
1.3 Basic Framework for Unit Tests	5
1.4 Note on Programming Style	6
1.4.1 Thoughts on Source Code Compactness	6
1.4.2 Example 1	7
1.4.3 Example 2.....	8
1.4.4 Thoughts on <code>final</code> and <code>var</code>	9
1.4.5 Notes on Method Visibility	10
1.4.6 Block Comments in Listings	11
1.4.7 Thoughts on Formatting	11
1.5 Trying Out the Examples and Solutions.....	11
1.6 Let's Go: Discovering Java Challenge	12

Part I: Fundamentals 13

Chapter 2: Mathematical Problems 15

2.1 Introduction..... 15

 2.1.1 Short Introduction to Division and Modulo 15

 2.1.2 Short Introduction to Divider 17

 2.1.3 Short Introduction to Prime Numbers..... 17

 2.1.4 Roman Numbers..... 20

 2.1.5 Rules..... 20

 2.1.6 Examples 21

 2.1.7 Noteworthy 21

 2.1.8 Number Games..... 22

 2.1.9 Perfect Numbers..... 22

 2.1.10 Armstrong Numbers 22

 2.1.11 Algorithm for a Simple Checksum 23

2.2 Exercises..... 24

 2.2.1 Exercise 1: Basic Arithmetic (★☆☆☆☆)..... 24

 2.2.2 Exercise 2: Number as Text (★★☆☆☆)..... 25

 2.2.3 Exercise 3: Perfect Numbers (★★☆☆☆)..... 26

 2.2.4 Exercise 4: Prime Numbers (★★☆☆☆)..... 27

 2.2.5 Exercise 5: Prime Number Pairs (★★☆☆☆) 27

 2.2.6 Exercise 6: Checksum (★★☆☆☆)..... 28

 2.2.7 Exercise 7: Roman Numbers (★★★☆☆) 28

 2.2.8 Exercise 8: Combinatorics (★★☆☆☆) 29

 2.2.9 Exercise 9: Armstrong Numbers (★★☆☆☆) 29

 2.2.10 Exercise 10: Max Change Calculator (★★★★☆) 30

 2.2.11 Exercise 11: Related Numbers (★★☆☆☆)..... 31

 2.2.12 Exercise 12: Prime Factorization (★★★★☆) 31

2.3 Solutions 32

 2.3.1 Solution 1: Basic Arithmetic (★☆☆☆☆)..... 32

 2.3.2 Solution 2: Number as Text (★★☆☆☆)..... 36

2.3.3 Solution 3: Perfect Numbers (★★☆☆☆)	38
2.3.4 Solution 4: Prime Numbers (★★☆☆☆)	41
2.3.5 Solution 5: Prime Number Pairs (★★☆☆☆)	44
2.3.6 Solution 6: Checksum (★★☆☆☆)	50
2.3.7 Solution 7: Roman Numbers (★★★★☆)	51
2.3.8 Solution 8: Combinatorics (★★☆☆☆)	57
2.3.9 Solution 9: Armstrong Numbers (★★☆☆☆)	62
2.3.10 Solution 10: Max Change Calculator (★★★★☆)	66
2.3.11 Solution 11: Related Numbers (★★☆☆☆)	68
2.3.12 Solution 12: Prime Factorization (★★★★☆)	70
Chapter 3: Recursion	75
3.1 Introduction	75
3.1.1 Mathematical Examples	75
3.1.2 Algorithmic Examples	79
3.1.3 Steps When Multiplying the Digits of a Number	85
3.1.4 Typical Problems	86
3.2 Exercises	90
3.2.1 Exercise 1: Fibonacci (★★☆☆☆)	90
3.2.2 Exercise 2: Process Digits (★★☆☆☆)	91
3.2.3 Exercise 3: GCD (★★☆☆☆)	91
3.2.4 Exercise 4: Reverse String (★★☆☆☆)	92
3.2.5 Exercise 5: Array Sum (★★☆☆☆)	93
3.2.6 Exercise 6: Array Min (★★☆☆☆)	93
3.2.7 Exercise 7: Conversions (★★☆☆☆)	94
3.2.8 Exercise 8: Exponential Function (★★☆☆☆)	95
3.2.9 Exercise 9: Pascal's Triangle (★★☆☆☆)	96
3.2.10 Exercise 10: Number Palindromes (★★★★☆)	96
3.2.11 Exercise 11: Permutations (★★★★☆)	97
3.2.12 Exercise 12: Count Substrings (★★☆☆☆)	97
3.2.13 Exercise 13: Ruler (★★☆☆☆)	98

TABLE OF CONTENTS

- 3.3 Solutions 99
 - 3.3.1 Solution 1: Fibonacci (★★☆☆☆)..... 99
 - 3.3.2 Solution 2: Process Digits (★★☆☆☆) 101
 - 3.3.3 Solution 3: GCD (★★☆☆☆) 103
 - 3.3.4 Solution 4: Reverse String (★★☆☆☆) 106
 - 3.3.5 Solution 5: Array Sum (★★☆☆☆) 108
 - 3.3.6 Solution 6: Array Min (★★☆☆☆)..... 110
 - 3.3.7 Solution 7: Conversions (★★☆☆☆) 112
 - 3.3.8 Solution 8: Exponential Function (★★☆☆☆) 116
 - 3.3.9 Solution 9: Pascal’s Triangle (★★☆☆☆) 120
 - 3.3.10 Solution 10: Number Palindromes (★★★★☆)..... 124
 - 3.3.11 Solution 11: Permutations (★★★★☆) 129
 - 3.3.12 Solution 12: Count Substrings (★★☆☆☆) 133
 - 3.3.13 Solution 13: Ruler (★★☆☆☆) 138

- Chapter 4: Strings..... 141**
 - 4.1 Introduction..... 141
 - 4.1.1 The Interface CharSequence 141
 - 4.1.2 The Class String..... 142
 - 4.1.3 The Classes StringBuffer and StringBuilder 143
 - 4.1.4 The Class Character..... 145
 - 4.1.5 Examples Related to Character and String..... 146
 - 4.2 Exercises..... 150
 - 4.2.1 Exercise 1: Number Conversions (★★☆☆☆) 150
 - 4.2.2 Exercise 2: Joiner (★☆☆☆☆)..... 151
 - 4.2.3 Exercise 3: Reverse String (★★☆☆☆) 151
 - 4.2.4 Exercise 4: Palindrome (★★★★☆) 152
 - 4.2.5 Exercise 5: No Duplicate Chars (★★★★☆)..... 153
 - 4.2.6 Exercise 6: Remove Duplicate Letters (★★★★☆) 153
 - 4.2.7 Exercise 7: Capitalize (★★☆☆☆)..... 153
 - 4.2.8 Exercise 8: Rotation (★★☆☆☆)..... 154

4.2.9 Exercise 9: Well-Formed Braces (★★☆☆☆)	155
4.2.10 Exercise 10: Anagram (★★☆☆☆)	155
4.2.11 Exercise 11: Morse Code (★★☆☆☆)	156
4.2.12 Exercise 12: Pattern Checker (★★★☆☆)	157
4.2.13 Exercise 13: Tennis Score (★★★☆☆)	157
4.2.14 Exercise 14: Version Numbers (★★☆☆☆)	158
4.2.15 Exercise 15: Conversion strToLong (★★☆☆☆)	159
4.2.16 Exercise 16: Print Tower (★★★☆☆)	159
4.3 Solutions	160
4.3.1 Solution 1: Number Conversions (★★☆☆☆)	160
4.3.2 Solution 2: Joiner (★☆☆☆☆)	164
4.3.3 Solution 3: Reverse String (★★☆☆☆)	166
4.3.4 Solution 4: Palindrome (★★★☆☆)	169
4.3.5 Solution 5: No Duplicate Chars (★★★☆☆)	173
4.3.6 Solution 6: Remove Duplicate Letters (★★★☆☆)	175
4.3.7 Solution 7: Capitalize (★★☆☆☆)	177
4.3.8 Solution 8: Rotation (★★☆☆☆)	182
4.3.9 Solution 9: Well-Formed Braces (★★☆☆☆)	184
4.3.10 Solution 10: Anagram (★★☆☆☆)	186
4.3.11 Solution 11: Morse Code (★★☆☆☆)	187
4.3.12 Solution 12: Pattern Checker (★★★☆☆)	191
4.3.13 Solution 13: Tennis Score (★★★☆☆)	193
4.3.14 Solution 14: Version Numbers (★★☆☆☆)	198
4.3.15 Solution 15: Conversion strToLong (★★☆☆☆)	201
4.3.16 Solution 16: Print Tower (★★★☆☆)	206
Chapter 5: Arrays	211
5.1 Introduction	211
5.1.1 One-Dimensional Arrays	212
5.1.2 Multidimensional Arrays	225
5.1.3 Typical Errors	234

TABLE OF CONTENTS

- 5.2 Exercises 236
 - 5.2.1 Exercise 1: Even Before Odd Numbers (★★☆☆☆) 236
 - 5.2.2 Exercise 2: Flip (★★☆☆☆)..... 236
 - 5.2.3 Exercise 3: Palindrome (★★☆☆☆) 237
 - 5.2.4 Exercise 4: Inplace Rotate (★★★☆☆) 237
 - 5.2.5 Exercise 5: Jewels Board Init (★★★☆☆)..... 238
 - 5.2.6 Exercise 6: Jewels Board Erase Diamonds (★★★★☆) 239
 - 5.2.7 Exercise 7: Spiral Traversal (★★★★☆) 241
 - 5.2.8 Exercise 8: Add One to Array as Number (★★☆☆☆) 242
 - 5.2.9 Exercise 9: Sudoku Checker (★★★☆☆) 242
 - 5.2.10 Exercise 10: Flood Fill (★★☆☆☆)..... 244
 - 5.2.11 Exercise 11: Array Merge (★★☆☆☆)..... 245
 - 5.2.12 Exercise 12: Array Min and Max (★★☆☆☆)..... 245
 - 5.2.13 Exercise 13: Array Split (★★★☆☆)..... 246
 - 5.2.14 Exercise 14: Minesweeper Board (★★★☆☆) 248
- 5.3 Solutions 250
 - 5.3.1 Solution 1: Even Before Odd Numbers (★★☆☆☆)..... 250
 - 5.3.2 Solution 2: Flip (★★☆☆☆)..... 255
 - 5.3.3 Solution 3: Palindrome (★★☆☆☆)..... 260
 - 5.3.4 Solution 4: Inplace Rotate (★★★☆☆)..... 263
 - 5.3.5 Solution 5: Jewels Board Init (★★★☆☆) 269
 - 5.3.6 Solution 6: Jewels Board Erase Diamonds (★★★★☆)..... 279
 - 5.3.7 Solution 7: Spiral Traversal (★★★★☆)..... 292
 - 5.3.8 Solution 8: Add One to Array as Number (★★☆☆☆)..... 299
 - 5.3.9 Solution 9: Sudoku Checker (★★★☆☆)..... 302
 - 5.3.10 Solution 10: Flood Fill (★★☆☆☆) 310
 - 5.3.11 Solution 11: Array Merge (★★☆☆☆) 316
 - 5.3.12 Solution 12: Array Min and Max (★★☆☆☆)..... 321
 - 5.3.13 Solution 13: Array Split (★★★☆☆) 325
 - 5.3.14 Solution 14: Minesweeper Board (★★★☆☆)..... 332

Chapter 6: Date Processing	341
6.1 Introduction.....	341
6.1.1 The Enumerations DayOfWeek and Month	341
6.1.2 The Classes LocalDate, LocalTime, and LocalDateTime	342
6.1.3 The Class ZonedDateTime	345
6.1.4 The class ZonedDateTime.....	346
6.1.5 The Class Duration.....	347
6.1.6 The Class Period	349
6.1.7 Date Arithmetic.....	350
6.1.8 Formatting and Parsing	354
6.2 Exercises.....	356
6.2.1 Exercise 1: Leap Years (★☆☆☆☆)	356
6.2.2 Exercise 2: Basic Knowledge Date-API (★★☆☆☆).....	356
6.2.3 Exercise 3: Length of Month (★★☆☆☆)	357
6.2.4 Exercise 4: Time Zones (★★☆☆☆)	357
6.2.5 Exercise 5: Time Zone Calculation (★★☆☆☆)	357
6.2.6 Exercise 6: Calculations with LocalDate.....	358
6.2.7 Exercise 7: Calendar Output (★★★☆☆)	359
6.2.8 Exercise 8: Weekdays (★☆☆☆☆)	359
6.2.9 Exercise 9: Sundays and Leap Years (★★☆☆☆).....	361
6.2.10 Exercise 10: TemporalAdjuster (★★★☆☆)	361
6.2.11 Exercise 11: NthWeekdayAdjuster (★★★☆☆).....	362
6.2.12 Exercise 12: Payday TemporalAdjuster (★★★☆☆).....	362
6.2.13 Exercise 13: Formatting and Parsing (★★☆☆☆)	363
6.2.14 Exercise 14: Fault-Tolerant Parsing (★★☆☆☆)	364
6.3 Solutions	364
6.3.1 Solution 1: Leap Years (★☆☆☆☆).....	364
6.3.2 Solution 2: Basic Knowledge Date-API (★★☆☆☆)	366
6.3.3 Solution 3: Length of Month (★★☆☆☆).....	367
6.3.4 Solution 4: Time Zones (★★☆☆☆).....	368

TABLE OF CONTENTS

- 6.3.5 Solution 5: Time Zone Calculation (★★☆☆☆)..... 369
- 6.3.6 Solution 6: Calculations With LocalDate 371
- 6.3.7 Solution 7: Calendar Output (★★★☆☆)..... 374
- 6.3.8 Solution 8: Weekdays (★☆☆☆☆)..... 378
- 6.3.9 Solution 9: Sundays and Leap Years (★★☆☆☆)..... 381
- 6.3.10 Solution 10: TemporalAdjuster (★★★☆☆)..... 384
- 6.3.11 Solution 11: NthWeekdayAdjuster (★★★☆☆) 386
- 6.3.12 Solution 12: Payday TemporalAdjuster (★★★☆☆) 388
- 6.3.13 Solution 13: Formatting and Parsing (★★☆☆☆) 393
- 6.3.14 Solution 14: Fault-Tolerant Parsing (★★☆☆☆)..... 396

- Chapter 7: Basic Data Structures: Lists, Sets, and Maps 399**
- 7.1 Introduction..... 399
 - 7.1.1 The Interface Collection..... 399
 - 7.1.2 Lists and the Interface List<E> 400
 - 7.1.3 Sets and the Interface Set..... 401
 - 7.1.4 Key-Value Mappings and the Interface Map..... 401
 - 7.1.5 The Stack as a LIFO Data Structure..... 403
 - 7.1.6 The Queue as a FIFO Data Structure..... 404
- 7.2 Exercises..... 406
 - 7.2.1 Exercise 1: Set Operations (★★☆☆☆)..... 406
 - 7.2.2 Exercise 2: List Reverse (★★☆☆☆)..... 407
 - 7.2.3 Exercise 3: Remove Duplicates (★★☆☆☆)..... 407
 - 7.2.4 Exercise 4: Maximum Profit (★★★☆☆)..... 408
 - 7.2.5 Exercise 5: Longest Sequence (★★★☆☆)..... 408
 - 7.2.6 Exercise 6: Own Stack (★★☆☆☆) 409
 - 7.2.7 Exercise 7: Well-Formed Braces (★★☆☆☆) 409
 - 7.2.8 Exercise 8: Check a Magic Triangle (★★★☆☆) 410
 - 7.2.9 Exercise 9: Pascal’s Triangle (★★★☆☆)..... 411
 - 7.2.10 Exercise 10: Most Frequent Elements (★★☆☆☆)..... 411
 - 7.2.11 Exercise 11: Addition of Digits (★★★☆☆)..... 411

7.2.12 Exercise 12: Compound Key (★★☆☆☆)	412
7.2.13 Exercise 13: List Merge (★★☆☆☆)	413
7.2.14 Exercise 14: Excel Magic Select (★★☆☆☆)	413
7.3 Solutions	414
7.3.1 Solution 1: Set Operations (★★☆☆☆)	414
7.3.2 Solution 2: List Reverse (★★☆☆☆)	421
7.3.3 Solution 3: Remove Duplicates (★★☆☆☆)	425
7.3.4 Solution 4: Maximum Profit (★★★☆☆)	427
7.3.5 Solution 5: Longest Sequence (★★★☆☆)	431
7.3.6 Solution 6: Own Stack (★★☆☆☆)	436
7.3.7 Solution 7: Well-Formed Braces (★★☆☆☆)	438
7.3.8 Solution 8: Check a Magic Triangle (★★★☆☆)	444
7.3.9 Solution 9: Pascal's Triangle (★★★☆☆)	450
7.3.10 Solution 10: Most Frequent Elements (★★☆☆☆)	452
7.3.11 Solution 11: Addition of Digits (★★★☆☆)	455
7.3.12 Solution 12: Compound Key (★★☆☆☆)	460
7.3.13 Solution 13: List Merge (★★☆☆☆)	464
7.3.14 Solution 14: Excel Magic Select (★★☆☆☆)	467
Part II: More Advanced and Tricky Topics	475
Chapter 8: Recursion Advanced	477
8.1 Memoization	477
8.1.1 Memoization for Fibonacci Numbers	477
8.1.2 Memoization for Pascal's Triangle	480
8.2 Backtracking	482
8.2.1 n-Queens Problem	483
8.3 Exercises	488
8.3.1 Exercise 1: Towers of Hanoi (★★★☆☆)	488
8.3.2 Exercise 2: Edit Distance (★★★★☆)	490
8.3.3 Exercise 3: Longest Common Subsequence (★★★☆☆)	490

TABLE OF CONTENTS

- 8.3.4 Exercise 4: Way Out of a Labyrinth (★★★☆☆) 491
- 8.3.5 Exercise 5: Sudoku Solver (★★★★☆) 492
- 8.3.6 Exercise 6: Math Operator Checker (★★★★☆) 493
- 8.3.7 Exercise 7: Water Jug Problem (★★★☆☆) 495
- 8.3.8 Exercise 8: All Palindrome Substrings (★★★★☆) 496
- 8.3.9 Exercise 9: n-Queens Problem (★★★☆☆) 496
- 8.4 Solutions 497
 - 8.4.1 Solution 1: Towers of Hanoi (★★★☆☆) 497
 - 8.4.2 Solution 2: Edit Distance (★★★★☆) 506
 - 8.4.3 Solution 3: Longest Common Subsequence (★★★☆☆) 515
 - 8.4.4 Solution 4: Way Out of a Labyrinth (★★★☆☆) 519
 - 8.4.5 Solution 5: Sudoku Solver (★★★★☆) 523
 - 8.4.6 Solution 6: Math Operator Checker (★★★★☆) 535
 - 8.4.7 Solution 7: Water Jug Problem (★★★☆☆) 542
 - 8.4.8 Solution 8: All Palindrome Substrings (★★★★☆) 546
 - 8.4.9 Solution 9: n-Queens Problem (★★★☆☆) 552
- Chapter 9: Binary Trees 565**
 - 9.1 Introduction 565
 - 9.1.1 Structure, Terminology, and Examples of Use 566
 - 9.1.2 Binary Trees 566
 - 9.1.3 Binary Trees with Order: Binary Search Trees 567
 - 9.1.4 Traversals 570
 - 9.1.5 Balanced Trees and Other Properties 574
 - 9.1.6 Trees for the Examples and Exercises 576
 - 9.2 Exercises 579
 - 9.2.1 Exercise 1: Tree Traversal (★★☆☆☆) 579
 - 9.2.2 Exercise 2: Preorder, Inorder, and Postorder Iteratives (★★★★☆) 579
 - 9.2.3 Exercise 3: Tree Height (★★☆☆☆) 580
 - 9.2.4 Exercise 4: Lowest Common Ancestor (★★★☆☆) 580
 - 9.2.5 Exercise 5: Breadth-First (★★★☆☆) 581

9.2.6 Exercise 6: Level Sum (★★★★☆)	581
9.2.7 Exercise 7: Tree Rotate (★★★☆☆)	582
9.2.8 Exercise 8: Reconstruction (★★★☆☆)	583
9.2.9 Exercise 9: Math Evaluation (★★☆☆☆)	584
9.2.10 Exercise 10: Symmetry (★★☆☆☆)	584
9.2.11 Exercise 11: Check Binary Search Tree (★★☆☆☆)	586
9.2.12 Exercise 12: Completeness (★★★★★)	586
9.2.13 Exercise 13: Tree Printer (★★★★★)	588
9.3 Solutions	593
9.3.1 Solution 1: Tree Traversal (★★☆☆☆)	593
9.3.2 Solution 2: Preorder, Inorder, and Postorder Iteratives (★★★★☆)	596
9.3.3 Solution 3: Tree Height (★★☆☆☆)	607
9.3.4 Solution 4: Lowest Common Ancestor (★★★☆☆)	609
9.3.5 Solution 5: Breadth-First (★★★☆☆)	613
9.3.6 Solution 6: Level Sum (★★★★☆)	617
9.3.7 Solution 7: Tree Rotate (★★★☆☆)	621
9.3.8 Solution 8: Reconstruction (★★★☆☆)	626
9.3.9 Solution 9: Math Evaluation (★★☆☆☆)	634
9.3.10 Solution 10: Symmetry (★★☆☆☆)	636
9.3.11 Solution 11: Check Binary Search Tree (★★☆☆☆)	642
9.3.12 Solution 12: Completeness (★★★★★)	644
9.3.13 Solution 13: Tree Printer (★★★★★)	657
Chapter 10: Searching and Sorting	673
10.1 Introduction Search	673
10.1.1 Searching in Collections and Arrays	673
10.1.2 Binary Search with <code>binarySearch()</code>	675
10.2 Introduction Sort	676
10.2.1 Insertion Sort	676
10.2.2 Selection Sort	679
10.2.3 Merge Sort	682

TABLE OF CONTENTS

- 10.2.4 Quick Sort..... 684
- 10.2.5 Bucket Sort..... 687
- 10.2.6 Final Thoughts 688
- 10.3 Exercises..... 689
 - 10.3.1 Exercise 1: Contains All (★★☆☆☆)..... 689
 - 10.3.2 Exercise 2: Partitioning (★★★★☆)..... 689
 - 10.3.3 Exercise 3: Binary Search (★★☆☆☆)..... 690
 - 10.3.4 Exercise 4: Insertion Sort (★★☆☆☆)..... 691
 - 10.3.5 Exercise 5: Selection Sort (★★☆☆☆)..... 691
 - 10.3.6 Exercise 6: Quick Sort (★★☆☆☆)..... 692
 - 10.3.7 Exercise 7: Bucket Sort (★★☆☆☆)..... 693
 - 10.3.8 Exercise 8: Search in Rotated Data (★★★★☆) 693
- 10.4 Solutions 694
 - 10.4.1 Solution 1: Contains All (★★☆☆☆) 694
 - 10.4.2 Solution 2: Partitioning (★★★★☆) 696
 - 10.4.3 Solution 3: Binary Search (★★☆☆☆) 699
 - 10.4.4 Solution 4: Insertion Sort (★★☆☆☆) 704
 - 10.4.5 Solution 5: Selection Sort (★★☆☆☆) 705
 - 10.4.6 Solution 6: Quick Sort (★★☆☆☆) 707
 - 10.4.7 Solution 7: Bucket Sort (★★☆☆☆) 710
 - 10.4.8 Solution 8: Search in Rotated Data (★★★★☆)..... 712
- Chapter 11: Conclusion and Supplementary Literature..... 721**
 - 11.1 Conclusion 721
 - 11.1.1 Lessons Learned per Chapter..... 721
 - 11.1.2 Noteworthy 723
 - 11.2 Puzzles..... 725
 - 11.2.1 Gold Bags: Detect the Fake 725
 - 11.2.2 Horse Race: Determine Fastest Three Horses 727

11.3 Supplementary Literature	729
11.3.1 Introduction to Algorithms and Data Structures	730
11.3.2 Basic Books	730
11.3.3 Specializing in Interview Questions.....	731
11.3.4 Supplements for Job Interviews at Top Companies	731
Part III: Appendices	733
Appendix A: Quick Start for JShell	735
A.1 Java + REPL => jshell	735
A.1.1 Introductory Example	735
A.1.2 More Commands and Possibilities	736
A.1.3 Using Syntactic Specialties and Modern Java Features	737
A.1.4 More Complex Actions.....	738
A.1.5 Exiting the JShell.....	739
Appendix B: Short Introduction to JUnit 5.....	741
B.1 Writing and Running Tests	741
B.1.1 Example: A First Unit Test	741
B.1.2 Fundamentals of Writing and Running Tests	742
B.1.3 Handling Expected Exceptions with <code>assertThrows()</code>	746
B.2 Parameterized Tests with JUnit 5	748
B.2.1 Introduction to Parameterized Tests with JUnit 5.....	748
B.2.2 More Practical Parameterized Tests	750
B.2.3 JUnit Parameterized Tests with <code>@MethodSource</code>	751
Appendix C: Quick Start for O-notation	753
C.1 Estimations with O-notation	753
C.1.1 Complexity Classes	754
C.1.2 Complexity and Program Running Time	756
Bibliography	759
Index.....	761