

<<数据结构Java语言描述>>

图书基本信息

书名：<<数据结构Java语言描述>>

13位ISBN编号：9787302074151

10位ISBN编号：7302074151

出版时间：2003-11

出版时间：第1版 (2003年11月1日)

作者：贝利 (Duane A. Bailey)

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<数据结构Java语言描述>>

内容概要

本书为数据结构的教材，讲述如何用开放的、纯面向对象的Java作为描述语言来设计和实现传统的数据结构。

全书结构严谨，讲解清晰，提供了大量的示例，使读者不仅能学习数据结构的具体实现，而且抽象出一般的设计原则，掌握并灵活运用这些原则，将使读者受益匪浅。

本书可作为计算机及相关专业的数据结构课程的教材。
对于不熟悉Java语言的读者，建议先进行附录B的Java语言学习。

作者简介

作者：（美国）贝利（Duane A. Bailey）

书籍目录

Preface to First Edition Preface to the Second Edition 0 Introduction 0.1 Read Me 0.2 He Can ' t say That, Can He? 1 The Object-Oriented Method 1.1 Data Abstraction and Encapsulation 1.2 The Object Model 1.3 Object-Oriented Terminology 1.4 A Special-Purpose Class: A Bank Account 1.5 A General-Purpose Class: An Association 1.6 Sketching an Example: A Word List 1.7 Sketching an Example: A Rectangle Class 1.8 Interfaces 1.9 Who Is the User? 1.10 Conclusions 1.11 Laboratory: The Day of the Week Calculator 2 Comments, Conditions, and Assertions 2.1 Pre- and Postconditions 2.2 Assertions 2.3 Craftsmanship 2.4 Conclusions 2.5 Laboratory: Using Javadoc Commenting 3 Vectors 3.1 The Interface 3.2 Example: The Word List Revisited 3.3 Example: Word Frequency 3.4 The Implementation 3.5 Extensibility: A Feature 3.6 Example: L-Systems 3.7 Example: Vector-Based Sets 3.8 Example: The Matrix Class 3.9 Conclusions 3.10 Laboratory: The Silver Dollar Game 4 Design Fundamentals 4.1 Asymptotic Analysis Tools 4.1.1 Time and Space Complexity 4.1.2 Examples 4.1.3 The Trading of Time and Space 4.1.4 Back-of-the-Envelope Estimations 4.2 Self-Reference 4.2.1 Recursion 4.2.2 Mathematical Induction 4.3 Properties of Design 4.3.1 Symmetry 4.3.2 Friction 4.4 Conclusions 4.5 Laboratory: How Fast Is Java? 5 Sorting 5.1 Approaching the Problem 5.2 Selection Sort 5.3 Insertion Sort 5.4 Mergesort 5.5 Quicksort 5.6 Radix Sort 5.7 Sorting Objects 5.8 Ordering Objects Using Comparators 5.9 Vector-Based Sorting 5.10 Conclusions 5.11 Laboratory: Sorting with Comparators 6 A Design Method 6.1 The Interface-Based Approach 6.1.1 Design of the Interface 6.1.2 Development of an Abstract Implementation 6.1.3 Implementation 6.2 Example: development of Generators 6.3 Example: Playing Cards 6.4 Conclusions 7 Iterators 7.1 Java ' s Enumeration Interface 7.2 The Iterator Interface 7.3 Example: vector Iterators 7.4 Example: Rethinking Generators 7.5 Example: Filtering Iterators 7.6 Conclusions 7.7 Laboratory: The Two-Towers Problem 8 Lists 8.1 Example: A Unique Program 8.2 Example: Free Lists 8.3 Partial Implementation: Abstract Lists 8.4 Implementation: Singly Linked Lists 8.5 Implementation: Doubly Linked Lists 8.6 Implementation: Circularly Linked Lists 8.7 Implementation: Vectors 8.8 List Iterators 8.9 Conclusions 8.10 Laboratory: Lists with Dummy Nodes 9 Linear Structures 9.1 Stacks 9.1.1 Example: simulating Recursion 9.1.2 Vector-Based Stacks 9.1.3 List-Based Stacks 9.1.4 Comparisons 9.2 Queues 9.2.1 Example: Solving a Coin Puzzle 9.2.2 List-Based Queues 9.2.3 Vector-Based Queues 9.2.4 Array-Based Queues 9.3 Example: Solving Mazes 9.4 Conclusions 9.5 Laboratory: A Stack-Based Language 9.6 Laboratory: The Web Crawler 10 Ordered Structures 10.1 Comparable Objects Revisited 10.1.1 Example: Comparable Ratios 10.1.2 Example: Comparable Associations 10.2 Keeping Structures Ordered 10.2.1 The OrderedStructure Interface 10.2.2 The Ordered Vector and Binary Search 10.2.3 Example: Sorting Revisited 10.2.4 A Comparator-based Approach 10.2.5 The Ordered List 10.2.6 Example: The Modified Parking Lot 10.3 Conclusions 10.4 Laboratory: Computing the " Best Of " 11 Binary Trees 11.1 Terminology 11.2 Example: Pedigree Charts 11.3 Example: Expression Trees 11.4 Implementation 11.4.1 The BinaryTree Implementation 11.5 Example: An Expert System 11.6 Traversals of Binary Trees 11.6.1 Preorder Traversal 11.6.2 In-order Traversal 11.6.3 Postorder Traversal 11.6.4 Level-order Traversal 11.6.5 Recursion in Iterators 11.7 Property-Based Methods 11.8 Example: Huffman Compression 11.9 Example Implementation: Ahnentafel 11.10 Conclusions 11.11 Laboratory: Playing Gardner ' s Hex-a-Pawn 12 Priority Queues 12.1 The Interface 12.2 Example: Improving the Huffman Code 12.3 A Vector-Based Implementation 12.4 A Heap Implementation 12.4.1 Vector-Based Heaps 12.4.2 Example: Heapsort 12.4.3 Skew Heaps 12.5 Example: Circuit Simulation 12.6 Conclusions 12.7 Laboratory: Simulating Business 13 Search Trees 13.1 Binary Search Trees 13.2 Example: Tree Sort 13.3 Example: Associative Structures 13.4 Implementation 13.5 Splay Trees 13.6 Splay Tree Implementation 13.7 An Alternative: Red-Black Trees 13.8 Conclusions 13.9 Laboratory: Improving the BinarySearchTree 14 Maps 14.1 Example Revisited: The Symbol Table 14.2 The Interface 14.3 Simple Implementation: MapList 14.4 Constant time Maps: Hash Tables 14.4.1 Open Addressing 14.4.2 External chaining 14.4.3 Generation of Hash Codes 14.4.4 Hash Codes for Collection Classes 14.4.5 Performance Analysis 14.5 Ordered Maps and Tables 14.6

Example:Document Indexing 14.7 Conclusions 14.8 Laboratory:The Soundex Name Lookup System15 Graphs
15.1 Terminology 15.2 The Graph Interface 15.3 Implementations 15.3.1 Abstract Classes Reemphasized 15.3.2
Adjacency Matrices 15.3.3 Adjacency Lists 15.4 Examples:Common Graph Algorithms 15.4.1 Reachability
15.4.2 Topological Sorting 15.4.3 Transitive Closure 15.4.4 All Pairs Minimum Distance 15.4.5 Greedy
Algorithms 15.5 Conclusions 15.6 Laboratory:Converting Between UnitsA Answers A.1 Solutions to Self Check
Problems A.2 Solutions to Odd-Numbered ProblemsB Beginning with Java B.1 A First Program B.2 Declarations
B.2.1 Primitive Types B.2.2 Reference Types B.3 Important Classes B.3.1 The ReadStream Class B.3.2 The
PrintStream Class B.3.3 Strings B.4 Control Constructs B.4.1 Conditional Statements B.4.2 Loops B.5 Methods
B.6 Inheritance and Subtyping B.6.1 Inheritance B.6.2 Subtyping B.6.3 Interfaces and Abstract Classes B.7 Use
of the Assert Command B.8 Use of the Keyword ProtectedC Collections C.1 Collection Class Features C.2
Parallel Features C.3 ConversionD Documentation D.1 Structure Package Hierarchy D.2 PrinciplesE
Environments E.1 Downloading Software E.2 Creating Libraries E.3 Creating Project StationeryF further
ReadingG GlossaryIndex

<<数据结构Java语言描述>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>