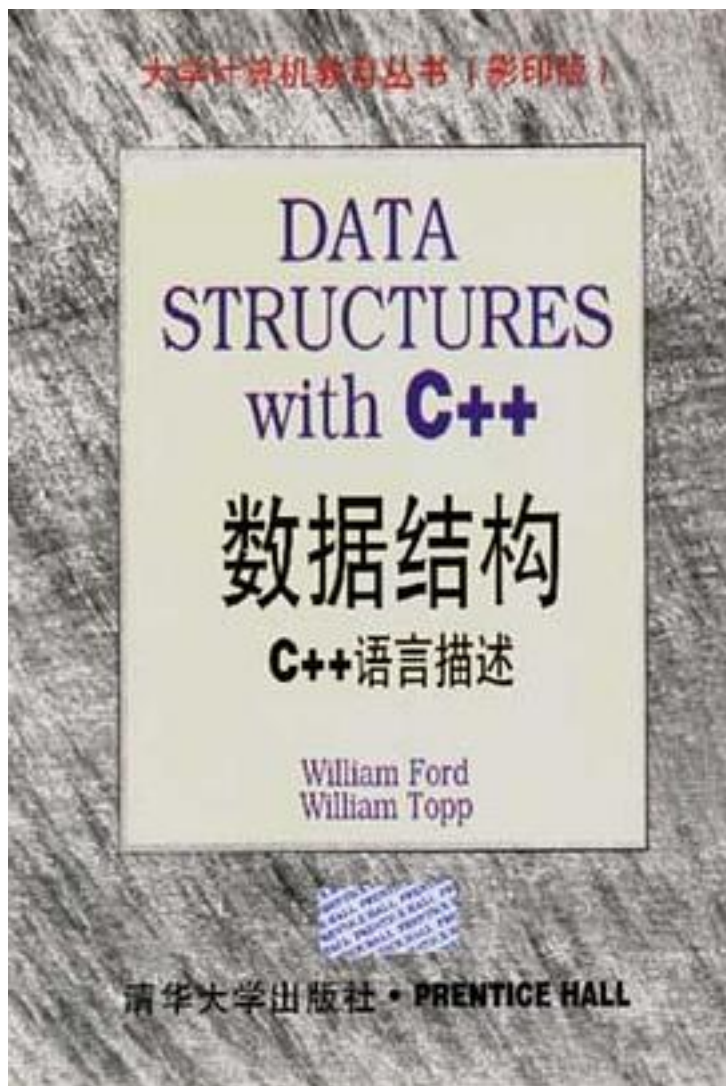


数据结构(C++语言描述)



[数据结构\(C++语言描述\)_下载链接1](#)

著者:(美)福特(Ford,W.)

出版者:清华大学出版社

出版时间:1997-03

装帧:平装

isbn:9787302024132

内容简介

本书从面向对象的视角介绍数据结构。内容从数据结构的基本原理到面向对象程序设计的方法。书内使用适应面极广的C++语言。全书14章分别为：1绪论；2基本数据类型；3抽象数据类型与类；4.集合类；5栈与队列；6.抽象运算符；7.类属数据类型；8.类与动态存储；9链表；10递归；11树；12继承与抽象类；13先进的非线性结构；14构建集合。书后附有练习答案。

作者介绍:

目录: Preface xvii
CHAPTER 1 INTRODUCTION
1.1 Abstract Data Types
ADT Format
1.2 C++ Classes and Abstract Types
Encapsulation and Information Hiding
Message Passing
1.3 Objects in C++ Applications
Application: The Circle Class
1.4 Object Design
Objects and Composition
C++ Geometric Classes
Objects and Inheritance
Inheritance in Programming
Ordered Lists and Inheritance
Software Reusability
SeqList and OrderedList Class Specifications
1.5 Applications with Class Inheritance
1.6 Object-Oriented Program Design
Problem Analysis/Program Definition
Design
Coding
Testing
Program Design Illustration: A Dice Graph
1.7 Program Testing and Maintenance
Object Testing
Control Module Testing
Program Maintenance and Documentation
1.8 The C++ Programming Language
1.9 Abstract Base Classes and Polymorphism
Polymorphism and Dynamic Binding
Written Exercises
CHAPTER 2 BASIC DATA TYPES
2.1 Integer Types
Computer Storage of Integers
Data in Memory

C++ Representation of Integers
2.2 Character Types
ASCII Characters
2.3 Real Data Types
Real Number Representations
2.4 Enumerated Types
Implementing C++ Enumerated Types
2.5 Pointers
Pointer ADT
Pointer Values
2.6 The Array Type
the Built-In C++ Array Type
Storage of One-Dimensional Arrays
Array Bounds
Two-Dimensional Arrays
Storage of Two-Dimensional Arrays
2.7 String Literals and Variables
C++ Strings
Application: Reversing Names
2.8 Records
C++ Structures
2.9 Files
C++ Stream Hierarchy
2.10 Array and Record Applications
Sequential Search
Exchange Sort
Counting C++ Reserved Words
Written Exercises
Programming Exercises
CHAPTER 3 ABSTRACT DATA TYPES AND CLASSES
3.1 The User Type CLASS
Class Declaration
Constructor
Object Declaration
Class Implementation
Implementing a Constructor
Building Objects
3.2 Sample Classes
The Temperature Class
The Random Number Class
3.3 Objects and Information Passing
An Object as a Return Value
An Object as a Function Parameter
3.4 Arrays of Objects
The Default Constructor
3.5 Multiple Constructors
3.6 Case Study: Triangular Matrices
Upper Triangular Matrix Properties
Written Exercises
Programming Exercises
CHAPTER 4 COLLECTION CLASSES
4.1 Describing Linear Collections
Direct Access Collections
Sequential Access Collections

Generalized Indexing
4.2 Describing Nonlinear Collections
Group Collections
4.3 Analysis of Algorithms
Performance Criteria
Common Orders of Magnitude
4.4 The Sequential and Binary Search
Binary Search
4.5 The Basic Sequential List Class
List Modification Methods
Written Exercises
Programming Exercises
CHAPTER 5 STACKS AND QUEUES
5.1 Stacks
5.2 The Stack Class
5.3 Expression Evaluation
Postfix Evaluation
Application: A Postfix Calculator
5.4 Queues
5.5 The Queue Class
5.6 Priority Queues
A Priority Queue Class
5.7 Case Study: Event-Driven Simulation
Written Exercises
Programming Exercises
CHAPTER 6 ABSTRACT OPERATORS
6.1 Describing Operator Overloading
Client-Defined External Functions
Class Members
Friend Functions
6.2 Rational Number System
Representing Rational Numbers
Rational Number Arithmetic
Rational Number Conversion
6.3 The Rational Class
6.4 Rational Operators as Member Functions
Implementing the Rational Operators
6.5 The Rational Stream Operators as Friends
Implementing Rational Stream Operators
6.6 Converting Rational Numbers
Conversion to Object Type
Conversion from Object Type
6.7 Using Rational Numbers
Written Exercises
Programming Exercises
CHAPTER 7 GENERIC DATA TYPES
7.1 Template Functions
Template-Based Sort
7.2 Template Classes
Defining a Template Class
Declaring Template Class Objects
Defining Template Class Methods
7.3 Template List Classes
7.4 Infix Expression Evaluation

Written Exercises

Programming Exercises

CHAPTER 8 CLASSES AND DYNAMIC MEMORY

8.1 Pointers and Dynamic Data Structures

The Memory Allocation Operator New

Dynamic Array Allocation

The Memory Deallocation Operator Delete

8.2 Dynamically Allocated Objects

Deallocating Object Data: The Destructor

8.3 Assignment and Initialization

Assignment Issues

Overloading the Assignment Operator

The This Pointer

Initialization Issues

Creating a Copy Constructor

8.4 Safe Arrays

The Array Class

Memory Allocation for the Array Class

Array Bounds Checking and the Overloaded [] Operator

Converting an Object to a Pointer

Using the Array Class

8.5 A String Class

String Class Implementation

8.6 Pattern Matching

The Find Process

Pattern Matching Algorithm

Analysis of the Pattern Matching Algorithm

8.7 Integral Sets

Sets of Integral Types

C++ Bit Handling Operators

Representing Set Elements

The Sieve of Eratosthenes

Set Class Implementation

Written Exercises

Programming Exercises

CHAPTER 9 LINKED LISTS

Describing a Linked List

Chapter 9 Overview

9.1 The Node Class

Declaring a Node Type

Implementing the Node Class

9.2 Building Linked Lists

Creating a Node

Inserting a Node: InsertFront

Traversing a Linked List

Inserting a Node: InsertRear

Application: Student Graduation List

Creating an Ordered List

Application: Sorting with Linked Lists

9.3 Designing a Linked List Class

Linked List Data Members

Linked List Operations

9.4 The LinkedList Class

9.5 Implementing the LinkedList Class

9.6 Implementing Collections with Linked Lists
Linked Queues
Implementing Queue Methods
Linked SeqList Class
Implementing SeqList Data Access Methods
Application: Comparing SeqList Implementations
9.7 Case Study: A Print Spooler
Implementing the Spooler Update Method
Spooler Evaluation Methods
9.8 Circular Lists
Circular Node Class Implementation
Application: Solving the Josephus Problem
9.9 Doubly Linked Lists
Application: Doubly Linked List Sort
DNode Class Implementation
9.10 Case Study: Window Management
The Window List
WindowList Class Implementation
Written Exercises
Programming Exercises
CHAPTER 10 RECURSION
10.1 The Concept of Recursion
Recursive Definitions
Recursive Problems
10.2 Designing Recursive Functions
10.3 Recursive Code and the Runtime Stack
The Runtime Stack
10.4 Problem-Solving with Recursion
Binary Search
Combinatorics: The Committee Problem
Combinatorics: Permutations
Maze Handling
Maze Class Implementation
10.5 Evaluating Recursion
Written Exercises
Programming Exercises
CHAPTER 11 TREES
Tree Terminology
Binary Trees
11.1 Binary Tree Structure
Designing a TreeNode Class
Building a Binary Tree
11.2 Designing TreeNode Functions
Recursive Tree Traversals
11.3 Using Tree Scan Algorithms
Application: Visiting Tree Nodes
Application: Tree Print
Application: Copying and Deleting Trees
Application: Upright Tree Printing
11.4 Binary Search Trees
The Key in a Binary Search Tree Node
Operations on a Binary Search Tree
Declaring a Binary Search Tree ADT
11.5 Using Binary Search Trees

- Duplicate Nodes
- 11.6 The BlnSTree Implementation
- List Operations
- 11.7 Case Study: Concordance
- Written Exercises
- Programming Exercises
- CHAPTER 12 INHERITANCE AND ABSTRACT CLASSES
- 12.1 A Vlew of Inheritance
- Class Inherltance Terminology
- 12.2 Inheritance In C++
- Constructors and Derived Classes
- What Cannot Be Inherited
- 12.3 Polymorphism and Vlrtual Functlons
- Demonstratlng Polymorphlsm
- Appllcation: Geometrcl Figures and Virtual Methods
- Virtual Melhods and the Destructor
- 12.4 Abstract Base Classes
- Abstract Base Class-List
- Derlving SeqLlst from Abstract Base Class List
- 12.5 Iterators
- The Iterator Abstract Base Class
- Derlving Llst Iterators
- Building the SeqLlst Iterator
- Array Iterator
- Application: Merglng Sorted Runs
- Arraylterator Implementation
- 12.6 Ordered Llsts
- OrderedLlst Class Implementation
- 12.7 Heterogeneous Llsts
- Heterogeneous Arrays
- Heterogeneous Linked Lists
- Wrltten Exercises
- Programmlng Exerelses
- CHAPTER 13 ADVANCED NONLINEAR STRUCTURES
- 13.1 Array-Based Binary Trees
- Application: The Tournament Sort
- 13.2 Heaps
- The Heap as a List
- The Heap Class
- 13.3 Implementlng the Heap Class
- Appllcation: Heap Sort
- 13.4 Priorlty Queues
- Appllcation: Long Runs
- 13.5 AVL Trees
- AVL Tree Nodes
- 13.6 The AVL Tree Class
- Memory Allocation for the AVLTree
- Evaluatlng AVL Trees
- 13.7 Trce Iterators
- The Inorder Iterator
- Inorderlteralor Class Implementation
- Appllcation: TreeSort
- 13.8 Graphs
- Connected Components

13.9 The Graph Class
Declaring a Graph ADT
Graph Class Implementatlon
Graph Traversals
Appllcatlons
Reachabillty and Warshall's Algorlthm
Writlen Exercises
Programmng Exercises
CHAPTER 14 ORGANIZING COLLECTIONS
14.1 Basic Array Sorting Algorithms
The Selection Sort
The Bubble Sort
The Insertion Sort
14.2 QuickSort
QuickSort Description
QuickSort Algorithm
Comparison of Array Sort Algorithms
14.3 Hashing
Keys and a Hash Function
Hashing Functions
Other Hash Methods
Collision Resolution
14.4 Hash Table Class
Application: String Frequency
HashTable Class Implementation
HashTableIterator Class Implementation
14.5 The Performance of Searching Methods
14.6 Binary Files and External Data Operations
Binary Files
The BinFile Class
External File Searching
External File Sort
Long Run MergeSort
14.7 Dictionaries
Writlen Exercises
Programming Exercises
APPENDIX ANSWERS TO SELECTED EXERCISES
BIBLIOGRAPHY
• • • • • (收起)

[数据结构\(C++语言描述\) 下载链接1](#)

标签

数据结构

计算机

算法

计算机科学

C++

data-structure

评论

[数据结构\(C++语言描述\) 下载链接1](#)

书评

[数据结构\(C++语言描述\) 下载链接1](#)