

C++数据抽象和问题求解



[C++数据抽象和问题求解 下载链接1](#)

著者:卡拉拉

出版者:清华大学出版社

出版时间:2013-11-1

装帧:平装

isbn:9787302335993

这本经典、畅销的数据结构教材详细介绍了数据抽象的基础知识，强调作为面向对象方法基础原理的规范和实施之间的区别。书中使用的软件工程原则和概念以及UML图便于增强学生的理解。

本书特点

- ◆ 详细介绍了数据抽象，强调规范和实现之间的区别
- ◆ 广泛介绍了各种面向对象的编程技术

- ◆ 重点是核心的数据结构，而不是非必要的C++语言语法
- ◆ 说明了类和ADT在问题解决过程中的作用
- ◆ 诠释了ADT的主要应用，如查找航班图、事件驱动的模拟和八皇后问题
- ◆ 大部分章节中的例子都使用了标准模板库(STL)
- ◆ 介绍了递归
- ◆ 附录中提供了基本的C++语法，以帮助学生从其他语言转换为C++

作者介绍:

Frank M. Carrano是美国罗得岛大学(University of Rhode Island)计算机科学系荣誉退休教授，于1969年获得美国锡拉丘兹大学计算机专业博士学位。他的兴趣包括数据结构、计算机科学教育、社会问题的计算处理和数值计算。Carrano教授对计算机科学高年级本科课程的设计和交付特别感兴趣，曾撰写了多本著名的计算机科学高年级本科生教科书。

Timothy Henry是美国罗得岛大学计算机科学系副教授，1986年获得美国欧道明大学(Old Dominion University)计算机专业硕士学位，2001年获得美国罗得岛大学应用数学专业博士学位。从2000年至今一直保有美国PMI的项目管理专家(Project Management Professional, PMP)认证资格。他教授的课程有：数据结构与抽象、编程语言基础、操作系统与网络、计算机系统基础、计算机科学项目、文件系统取证等。研究的领域有：计算机和数字取证、交互式3D图形关系、传感器网络。

目录: 第1章 数据抽象：墙	1
1.1 面向对象的概念	2
1.1.1 面向对象分析与设计	2
1.1.2 面向对象解决方案的特征	3
1.2 获得更好的解决方案	4
1.2.1 内聚	5
1.2.2 耦合	5
1.3 规范	6
1.3.1 操作契约	7
1.3.2 特殊情况	8
1.3.3 抽象	9
1.3.4 信息隐藏	10
1.3.5 最小且完整的接口	11
1.4 抽象数据类型	12
1.4.1 设计ADT	14
1.4.2 涉及其他ADT的ADT	17
1.5 ADT包	18
1.5.1 确定行为	18
1.5.2 指定数据和操作	19
1.5.3 ADT的模板接口	22
1.5.4 使用ADT包	24
C++片段1 C++类	29
C1.1 待解决的问题	30

- C1.1.1 私有数据字段 31
- C1.1.2 构造函数和析构函数 32
- C1.1.3 方法 32
- C1.1.4 防止编译错误 33
- C1.2 实现解决方案 34
- C1.3 模板 35
- C1.4 继承 37
 - C1.4.1 基类和派生类 38
 - C1.4.2 重写基类方法 40
- C1.5 虚方法和抽象类 42
 - C1.5.1 虚方法 42
 - C1.5.2 抽象类 43
- 第2章 递归：镜子 45
 - 2.1 递归解决方案 46
 - 2.2 返回值的递归 48
 - 2.2.1 递归值函数：n的阶乘 49
 - 2.2.2 箱式跟踪 52
 - 2.3 执行动作的递归 55
 - 2.4 递归与数组 62
 - 2.4.1 逆置数组项 63
 - 2.4.2 折半查找 64
 - 2.4.3 查找数组中的最大值 68
 - 2.4.4 查找数组中第k个最小值 69
 - 2.5 组织数据 71
 - 2.6 更多示例 75
 - 2.6.1 Fibonacci数列(兔子繁殖) 75
 - 2.6.2 组织游行队伍 78
 - 2.6.3 从n个事物中选出k个 79
 - 2.7 递归和效率 81
- 第3章 基于数组的实现 91
 - 3.1 办法 92
 - 3.1.1 核心方法 93
 - 3.1.2 使用大小固定的数组 93
 - 3.2 ADT包的基于数组的实现 94
 - 3.2.1 头文件 95
 - 3.2.2 定义核心方法 96
 - 3.2.3 测试核心方法 98
 - 3.2.4 实现更多方法 101
 - 3.2.5 删除项的方法 103
 - 3.2.6 测试 106
 - 3.3 在实现中使用递归 107
 - 3.3.1 getIndexOf方法 107
 - 3.3.2 getFrequencyOf方法 108
- C++片段2 指针、多态和内存分配 113
 - C2.1 变量的内存分配和方法的前期绑定 114
 - C2.2 需要解决的问题 115
 - C2.3 指针与程序的自由存储 116
 - C2.3.1 释放内存 118
 - C2.3.2 避免内存泄漏 119
 - C2.3.3 避免悬挂指针 122
 - C2.4 虚方法和多态 124
 - C2.5 数组的动态分配 126
- 第4章 基于链表的实现 129

- 4.1 预备知识 130
- 4.2 ADT包的基于链表的实现 133
 - 4.2.1 头文件 134
 - 4.2.2 定义核心方法 135
 - 4.2.3 实现更多方法 138
- 4.3 在基于链表的实现中使用递归 143
- 4.4 测试多个ADT实现 145
- 4.5 比较基于数组的实现和基于链表的实现 148
- 第5章 作为问题求解技术的递归 155
 - 5.1 定义语言 156
 - 5.1.1 语法知识基础 156
 - 5.1.2 两种简单的语言 158
 - 5.2 代数表达式 160
 - 5.2.1 代数表达式的类型 160
 - 5.2.2 前缀表达式 162
 - 5.2.3 后缀表达式 166
 - 5.2.4 完全括号化表达式 168
 - 5.3 回溯 168
 - 5.3.1 查找航线 168
 - 5.3.2 八皇后问题 173
 - 5.4 递归和数学归纳法的关系 179
 - 5.4.1 递归阶乘函数的正确性 179
 - 5.4.2 Hanoi塔的工作量 180
- 第6章 栈 189
 - 6.1 ADT栈 190
 - 6.1.1 在设计解决方案期间开发ADT 190
 - 6.1.2 ADT栈的规范 192
 - 6.2 栈的简单应用 197
 - 6.2.1 检查括号匹配 197
 - 6.2.2 识别语言中的字符串 199
 - 6.3 栈在代数表达式中的应用 200
 - 6.3.1 计算后缀表达式 201
 - 6.3.2 中缀表达式与后缀表达式的等价转换 202
 - 6.4 使用栈查找航班图 205
 - 6.5 栈和递归的关系 212
- C++片段3 异常 221
 - C3.1 背景知识 222
 - C3.2 断言 223
 - C3.3 抛出异常 224
 - C3.4 处理异常 227
 - C3.4.1 多个catch块 228
 - C3.4.2 未捕获的异常 229
 - C3.5 程序员定义的异常类 232
- 第7章 实现ADT栈 235
 - 7.1 基于数组的实现 236
 - 7.2 基于链表的实现 239
 - 7.3 在实现中使用异常 243
- 第8章 列表 247
 - 8.1 指定ADT列表 248
 - 8.2 使用列表操作 252
 - 8.3 ADT列表的模板接口 255
- 第9章 实现列表 259
 - 9.1 基于数组的ADT列表实现 260
 - 9.1.1 头文件 261

- 9.1.2 实现文件 262
- 9.2 基于链表的ADT列表实现 266
 - 9.2.1 头文件 266
 - 9.2.2 实现文件 268
 - 9.2.3 在LinkedList的方法中使用递归 275
- 9.3 两种实现的比较 279
- 第10章 算法的效率 283
 - 10.1 什么是好的解决方案 284
 - 10.2 测量算法的效率 285
 - 10.2.1 算法的执行时间 286
 - 10.2.2 算法增长率 287
 - 10.2.3 分析与大O表示法 288
 - 10.2.4 正确分析问题 291
 - 10.2.5 查找算法的效率 293
- 第11章 排序算法及其效率 299
 - 11.1 基本排序算法 300
 - 11.1.1 选择排序 300
 - 11.1.2 起泡排序 303
 - 11.1.3 插入排序 305
 - 11.2 较快排序算法 307
 - 11.2.1 归并排序 307
 - 11.2.2 快速排序 312
 - 11.2.3 基数排序 319
 - 11.3 各种排序算法的比较 321
- C++片段4 类关系和重用 325
 - C4.1 回顾继承 326
 - C4.1.1 类的公有、私有和受保护部分 331
 - C4.1.2 公有、私有和受保护继承 332
 - C4.1.3 is-a和as-a关系 333
 - C4.2 包含：has-a关系 334
 - C4.3 回顾抽象基类 335
- 第12章 有序表及其实现 339
 - 12.1 指定ADT有序表 340
 - 12.1.1 ADT有序表的模板接口 342
 - 12.1.2 使用有序表的操作 343
 - 12.2 基于链表的实现 344
 - 12.2.1 头文件 344
 - 12.2.2 实现文件 345
 - 12.2.3 基于链表的实现的效率 348
 - 12.3 使用ADT列表的实现 348
 - 12.3.1 包含 349
 - 12.3.2 公有继承 352
 - 12.3.3 私有继承 356
- 第13章 队列和优先队列 363
 - 13.1 ADT队列 364
 - 13.2 ADT队列的简单应用 367
 - 13.2.1 读取字符串 367
 - 13.2.2 识别回文 368
 - 13.3 ADT优先队列 369
 - 13.4 应用：模拟 371
 - 13.5 面向位置和面向值的ADT 379
- 第14章 队列和优先队列的实现 387
 - 14.1 ADT队列的实现 388
 - 14.1.1 使用ADT列表的实现 388

- 14.1.2 基于链表的实现 390
- 14.1.3 基于数组的实现 394
- 14.1.4 比较实现 399
- 14.2 ADT优先队列的实现 400
- C++片段5 运算符重载和友元访问 405
- C5.1 重载运算符 406
 - C5.1.1 重载=进行赋值 408
 - C5.1.2 重载+进行连接 410
- C5.2 友元访问和<<的重载 411
- 第15章 树 415
 - 15.1 术语 416
 - 15.1.1 树的类型 417
 - 15.1.2 树的高度 419
 - 15.1.3 满二叉树、完全二叉树和平衡二叉树 421
 - 15.1.4 二叉树的最大和最小高度 422
 - 15.2 ADT二叉树 425
 - 15.2.1 二叉树的遍历 425
 - 15.2.2 二叉树的操作 428
 - 15.2.3 ADT二叉树的模板接口 430
 - 15.3 ADT二叉查找树 432
 - 15.3.1 二叉查找树的操作 433
 - 15.3.2 查找二叉查找树 434
 - 15.3.3 创建二叉查找树 435
 - 15.3.4 遍历二叉查找树 437
 - 15.3.5 二叉查找树操作的效率 437
- 第16章 树的实现 443
 - 16.1 二叉树中的节点 444
 - 16.1.1 基于数组的表示 444
 - 16.1.2 基于链表的表示 446
 - 16.2 ADT二叉树基于链表的实现 447
 - 16.2.1 头文件 447
 - 16.2.2 实现 450
 - 16.3 ADT二叉查找树基于链表的实现 458
 - 16.3.1 ADT二叉查找树操作的算法 458
 - 16.3.2 BinarySearchTree类 469
 - 16.4 在文件中保存二叉查找树 471
 - 16.5 树排序 474
 - 16.6 一般树 474
- C++片段6 迭代器 479
- C6.1 迭代器 480
 - C6.1.1 常见的迭代器操作 481
 - C6.1.2 使用迭代器操作 482
 - C6.1.3 实现迭代器 483
- C6.2 迭代器的高级功能 485
- 第17章 堆 489
 - 17.1 ADT堆 490
 - 17.2 堆的基于数组的实现 493
 - 17.2.1 基于数组的堆操作的算法 494
 - 17.2.2 实现 498
 - 17.3 ADT优先队列的堆实现 502
 - 17.4 堆排序 504
- 第18章 字典及其实现 511
 - 18.1 ADT字典 512
 - 18.2 可能的实现 517

- 18.2.1 ADT字典的基于数组的有序实现 519
- 18.2.2 ADT字典的二叉查找树实现 521
- 18.3 选择实现 523
- 18.4 散列 529
 - 18.4.1 散列函数 532
 - 18.4.2 解决冲突 534
 - 18.4.3 散列的效率 539
 - 18.4.4 如何确立散列函数 542
 - 18.4.5 字典遍历：散列的低效操作 543
 - 18.4.6 使用散列和分离链实现ADT字典 544
- 第19章 平衡查找树 551
 - 19.1 平衡查找树 552
 - 19.2 2-3树 553
 - 19.2.1 遍历2-3树 555
 - 19.2.2 查找2-3树 556
 - 19.2.3 在2-3树中插入数据 558
 - 19.2.4 从2-3树中删除数据 562
 - 19.3 2-3-4树 567
 - 19.3.1 查找和遍历2-3-4树 569
 - 19.3.2 在2-3-4树中插入数据 569
 - 19.3.3 从2-3-4树中删除数据 572
 - 19.4 红-黑树 573
 - 19.4.1 查找和遍历红-黑树 575
 - 19.4.2 红-黑树的插入和删除 575
 - 19.5 AVL树 577
- 第20章 图 583
 - 20.1 术语 584
 - 20.2 将图作为ADT 587
 - 20.3 图的遍历 591
 - 20.3.1 深度优先查找 592
 - 20.3.2 广度优先查找 593
 - 20.4 图的应用 595
 - 20.4.1 拓扑排序 595
 - 20.4.2 生成树 598
 - 20.4.3 最小生成树 600
 - 20.4.4 最短路径 603
 - 20.4.5 回路 606
 - 20.4.6 一些复杂问题 608
- 第21章 外部存储中的数据处理 615
 - 21.1 了解外部存储 616
 - 21.2 排序外部文件的数据 618
 - 21.3 外部字典 624
 - 21.3.1 确定外部文件的索引 626
 - 21.3.2 外部散列 629
 - 21.3.3 B-树 632
 - 21.3.4 遍历 639
 - 21.3.5 多索引 640
- C++片段7 标准模板库 647
 - C7.1 STL容器 648
 - C7.1.1 STL容器适配器 649
 - C7.1.2 顺序容器 650
 - C7.1.3 关联容器 654
 - C7.2 STL算法 657
- 附录A 回顾C++基础 659

附录B 编程中的重要主题 697
附录C 统一建模语言 719
附录D 软件生命周期 727
附录E 数学归纳法 733
附录F 算法验证 737
附录G C++文件基础 741
附录H C++头文件和标准函数 751
附录I C++文档系统 755
附录J ASCII字符代码 757
附录K 针对Java编程人员的C++知识 759
附录L 针对Python编程人员的C++
知识 767
· · · · · (收起)

[C++数据抽象和问题求解 下载链接1](#)

标签

C++

计算机科学

评论

好于90%的同类书籍

个人觉得这本书内容有点混乱，说是c++的数据结构其实也没用到什么c++的语法，然后具体算法又不如之前看过的c版本简单清晰，还不如自己去github上扒点c++的数据结构看看

[C++数据抽象和问题求解 下载链接1](#)

书评

[C++数据抽象和问题求解 下载链接1](#)