

《轻松学算法》

图书基本信息

书名：《轻松学算法》

13位ISBN编号：9787121291940

出版时间：2016-7

作者：赵焯

页数：412

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

《轻松学算法》

内容概要

《轻松学算法——互联网算法面试宝典》共分为12个章节，首先介绍了一些基础的数据结构，以及常用的排序算法和查找算法；其次介绍了两个稍微复杂一些的数据结构——树和图，还介绍了每种数据结构和算法的适用场景，之后是一些在工作与面试中的实际应用，以字符串、数组、查找等为例介绍了一些常见的互联网面试题及分析思路，便于读者了解这些思路，顺利地通过互联网公司的面试；最后介绍了一些常见的算法思想，便于读者对今后遇到的算法问题更轻易地想出解决方案。

《轻松学算法——互联网算法面试宝典》的讲解轻松有趣，易于读者把烦琐、枯燥的算法学习变为有趣、愉快的学习，把被动学习变为主动学习。《轻松学算法——互联网算法面试宝典》也介绍了一些会在工作面试中用到的算法。对于一些正在学习算法的人来说，《轻松学算法——互联网算法面试宝典》绝对是可以帮你轻松掌握算法的辅助资料；对于已经了解算法的人来说，可以从《轻松学算法——互联网算法面试宝典》中了解到这些算法是如何在实际工作中使用的。

《轻松学算法——互联网算法面试宝典》适合即将毕业的学生、初入职场的工程师及想补充基础算法知识的人学习，也适合作为一本互联网公司面试的参考书，更是一本不可多得的便于读者时常补习算法知识的收藏宝典。

书籍目录

第1章 数组、集合和散列表 1

1.1 要用就要提前想好的数据结构—数组 2

1.1.1 什么是数组 2

1.1.2 数组的存储结构 3

1.1.3 数组的特点 6

1.1.4 数组的适用场景 7

1.2 升级版数组—集合 8

1.2.1 什么是集合 8

1.2.2 集合的实现 8

1.2.3 集合的特点 13

1.2.4 集合的适用场景 13

1.2.5 数组与变长数组的性能 14

1.3 数组的其他应用—散列表 14

1.3.1 什么是散列表 15

1.3.2 对散列表函数产生冲突的解决办法 16

1.3.3 散列表的存储结构 17

1.3.4 散列表的特点 18

1.3.5 散列表的适用场景 20

1.3.6 散列表的性能分析 21

1.4 小结 28

第2章 栈、队列、链表 29

2.1 汉诺塔游戏—栈 30

2.1.1 什么是汉诺塔 30

2.1.2 什么是栈 31

2.1.3 栈的存储结构 31

2.1.4 栈的特点 36

2.1.5 栈的适用场景 36

2.2 火爆的奶茶店—队列 37

2.2.1 什么是队列 37

2.2.2 队列的存储结构 38

2.2.3 队列的特点 43

2.2.4 队列的适用场景 44

2.3 用栈实现队列 45

2.3.1 用两个栈实现队列 46

2.3.2 两个队列实现栈 50

2.4 链表 53

2.4.1 什么是链表 54

2.4.2 链表的存储结构 54

2.4.3 链表的操作 55

2.4.4 链表的特点 66

2.4.5 链表的适用场景 66

2.4.6 链表的性能分析 67

2.4.7 面试举例：如何反转链表 68

2.5 链表其实也可以用数组模拟 69

2.5.1 静态链表 70

2.5.2 静态链表的实现 70

2.5.3 静态链表的特点 80

2.6	再谈汉诺塔	81
2.6.1	汉诺塔的移动原理	81
2.6.2	汉诺塔的递归实现	82
第3章	排序算法	84
3.1	算法基础	85
3.1.1	时间复杂度	85
3.1.2	空间复杂度	88
3.1.3	稳定性	88
3.2	快而简单的排序—桶排序	89
3.2.1	举个例子	89
3.2.2	什么是桶排序	90
3.2.3	桶排序的实现	90
3.2.4	桶排序的性能及特点	92
3.2.5	桶排序的适用场景	93
3.3	咕嘟咕嘟的冒泡排序	94
3.3.1	什么是冒泡排序	94
3.3.2	冒泡排序的原理	94
3.3.3	冒泡排序的实现	96
3.3.4	冒泡排序的特点及性能	99
3.3.5	冒泡排序的适用场景	99
3.3.6	冒泡排序的改进方案	100
3.4	最常用的快速排序	100
3.4.1	什么是快速排序	101
3.4.2	快速排序的原理	101
3.4.3	快速排序的实现	105
3.4.4	快速排序的特点及性能	107
3.4.5	快速排序的适用场景	108
3.4.6	快速排序的优化	108
3.5	简单的插入排序	109
3.5.1	什么是插入排序	110
3.5.2	插入排序的原理	110
3.5.3	插入排序的实现	112
3.5.4	插入排序的特点及性能	114
3.5.5	插入排序的适用场景	115
3.6	直接插入的改进—希尔排序	115
3.6.1	什么是希尔排序	116
3.6.2	希尔排序的原理	116
3.6.3	希尔排序的实现	118
3.6.4	希尔排序的特点及性能	120
3.6.5	希尔排序的适用场景	121
3.7	简单选择排序	121
3.7.1	什么是选择排序	122
3.7.2	简单选择排序的原理	122
3.7.3	简单选择排序的实现	123
3.7.4	选择排序的特点及性能	125
3.7.5	简单选择排序的优化	125
3.7.6	选择排序的适用场景	126
3.8	小结	126
第4章	搜索，没那么难	128

- 4.1 最先想到的—顺序查找 129
 - 4.1.1 最先想到的 129
 - 4.1.2 顺序查找的原理与实现 129
 - 4.1.3 顺序查找的特点及性能分析 131
 - 4.1.4 顺序查找的适用场景 132
- 4.2 能不能少查点—二分查找 133
 - 4.2.1 某些特殊情况的查找需求 133
 - 4.2.2 二分查找的原理及实现 133
 - 4.2.3 二分查找的优化 137
 - 4.2.4 二分查找的特点及性能分析 138
 - 4.2.5 二分查找的适用场景 139
 - 4.2.6 我是来还债的—之前欠的二分插入排序 139
- 4.3 行列递增的矩阵查找—二分查找思维拓展 141
 - 4.3.1 一道题 142
 - 4.3.2 几个解法 142
 - 4.3.3 其他拓展 153
- 4.4 分块查找 154
 - 4.4.1 每次插入元素都要有序吗 154
 - 4.4.2 什么是分块查找 155
 - 4.4.3 分块查找的原理及实现 155
 - 4.4.4 分块查找的特点与性能分析 159
 - 4.4.5 分块查找的适用场景 160
- 4.5 查找算法小结 161
- 4.6 搜索引擎与倒排索引 162
 - 4.6.1 什么是搜索引擎 162
 - 4.6.2 倒排索引 162
 - 4.6.3 索引实例 163
 - 4.6.4 倒排索引的关键字提取 164
 - 4.6.5 商业索引的其他拓展 164
- 第5章 树 166
 - 5.1 树的定义及存储结构 167
 - 5.1.1 什么是树 167
 - 5.1.2 其他相关术语 168
 - 5.1.3 都有哪些树 170
 - 5.1.4 树的存储结构及实现 170
 - 5.2 二叉树 173
 - 5.2.1 什么是二叉树 173
 - 5.2.2 还有两种特殊的二叉树 173
 - 5.2.3 二叉树的实现 175
 - 5.2.4 二叉树的遍历 185
 - 5.2.5 完全二叉树 187
 - 5.3 二叉树的查找算法 188
 - 5.3.1 二叉查找树 188
 - 5.3.2 平衡二叉树 198
 - 5.4 B-树、B+树 202
 - 5.4.1 什么是B-树 203
 - 5.4.2 什么是B+树 204
 - 5.4.3 B-树、B+树的特点及性能分析 205
 - 5.5 在MySQL数据库中是如何应用B+树的 206

- 5.6 哈夫曼树 209
 - 5.6.1 几个术语 209
 - 5.6.2 什么是哈夫曼树 209
 - 5.6.3 哈夫曼树的构造 211
 - 5.6.4 哈夫曼编码及解码 213
- 5.7 堆 215
 - 5.7.1 什么是堆 215
 - 5.7.2 堆的基本操作 216
 - 5.7.3 堆的性能分析 221
 - 5.7.4 堆排序 222
- 5.8 红黑树 224
 - 5.8.1 什么是红黑树 224
 - 5.8.2 红黑树的特点与优势 224
- 第6章 图 226
 - 6.1 图的定义及相关术语 227
 - 6.1.1 什么是图 227
 - 6.1.2 图的分类 227
 - 6.1.3 图的相关术语 228
 - 6.2 图的表示与存储方式 229
 - 6.2.1 邻接矩阵 229
 - 6.2.2 邻接表 234
 - 6.3 更多的图 237
 - 6.3.1 连通图 238
 - 6.3.2 强连通图 238
 - 6.4 深度优先遍历与广度优先遍历 238
 - 6.4.1 深度优先遍历 239
 - 6.4.2 广度优先遍历 248
 - 6.4.3 两种遍历方法的对比 253
 - 6.5 最短路径 254
 - 6.5.1 带权图 254
 - 6.5.2 Dijkstra算法 255
 - 6.5.3 Floyd算法 269
- 第7章 字符串 272
 - 7.1 字符及字符串简介 273
 - 7.1.1 什么是字符 273
 - 7.1.2 什么是字符串 273
 - 7.2 字符的全排列 275
 - 7.2.1 问题描述及分析 275
 - 7.2.2 最先想到的 275
 - 7.2.3 利用字典序排列 278
 - 7.3 反转字符串 283
 - 7.3.1 问题描述及分析 283
 - 7.3.2 最先想到的 283
 - 7.3.3 对换反转法 285
 - 7.3.4 拓展—旋转字符串 287
 - 7.4 判断回文 288
 - 7.4.1 问题描述及分析 288
 - 7.4.2 对半判断 289
 - 7.5 寻找最大的回文子串 290

- 7.5.1 问题描述及分析 290
- 7.5.2 遍历实现 291
- 7.6 将字符串转换为数字 293
 - 7.6.1 问题描述及分析 293
 - 7.6.2 解决 293
- 7.7 判断字符串包含的问题 297
 - 7.7.1 问题描述及分析 297
 - 7.7.2 非常简单的解决思路 297
 - 7.7.3 利用排序进行优化 299
 - 7.7.4 投机取巧的素数方案 302
 - 7.7.5 用散列表进行实现 304
 - 7.7.6 用位运算进行实现 305
- 第8章 数组还有好多玩法 308
 - 8.1 从数组中找出其和为指定值的两个数 309
 - 8.1.1 问题描述及分析 309
 - 8.1.2 最简单的办法 309
 - 8.1.3 一切为了速度 310
 - 8.1.4 排序总是好的选择 311
 - 8.1.5 还能更好 313
 - 8.2 找出连加值最大的子数组 315
 - 8.2.1 问题描述及分析 315
 - 8.2.2 暴力穷举法 316
 - 8.2.3 动态规划法 319
 - 8.2.4 问题拓展 321
 - 8.3 数组正负值排序 323
 - 8.3.1 问题描述及分析 323
 - 8.3.2 最直观的解法 324
 - 8.3.3 借鉴简单插入排序 325
 - 8.3.4 借鉴快速排序 327
 - 8.3.5 拓展 329
 - 8.4 将数组随机打乱顺序 329
 - 8.4.1 问题描述及分析 329
 - 8.4.2 随便糊弄一下 330
 - 8.4.3 插入排序的思想又来了 331
 - 8.5 数组赋值 333
 - 8.5.1 问题描述及分析 333
 - 8.5.2 分解计算 333
 - 8.6 寻找旋转数组的拐点 335
 - 8.6.1 问题描述及分析 335
 - 8.6.2 最简单的方法 336
 - 8.6.3 利用“有序” 337
 - 8.7 荷兰国旗问题 339
 - 8.7.1 问题描述及分析 339
 - 8.7.2 排序法 340
 - 8.7.3 快速排序带给人的灵感 340
- 第9章 查找又来了 344
 - 9.1 出现次数超过一半的数字 345
 - 9.1.1 问题描述及分析 345
 - 9.1.2 排序法 345

- 9.1.3 散列表 347
- 9.1.4 删除法 349
- 9.1.5 更优的解法 349
- 9.2 寻找缺少的数字 352
 - 9.2.1 问题描述及分析 352
 - 9.2.2 借助快速排序 352
 - 9.2.3 借助散列表实现 354
 - 9.2.4 投机取巧法 355
 - 9.2.5 思路拓展 357
- 9.3 在10亿个数中找出最大的1万个数 357
 - 9.3.1 问题描述及分析 357
 - 9.3.2 拍脑袋想问题 358
 - 9.3.3 借助快速排序 358
 - 9.3.4 不想都放入内存 358
 - 9.3.5 传说中的大顶堆 359
 - 9.3.6 拓展—找出数组中第k大的数 359
- 第10章 更多 363
 - 10.1 不使用额外的空间交换两个数 364
 - 10.1.1 问题描述 364
 - 10.1.2 分析问题 364
 - 10.1.3 解决问题 364
 - 10.2 拿乒乓球的问题 365
 - 10.2.1 问题描述 365
 - 10.2.2 分析问题 365
 - 10.2.3 解决问题 365
- 第11章 实现一些集合类 367
 - 11.1 栈 (Stack) 的实现 368
 - 11.1.1 实现前的思考 368
 - 11.1.2 实现栈 368
 - 11.1.3 参考JDK的实现 372
 - 11.2 变长数组 (ArrayList) 的实现 372
 - 11.2.1 实现前的思考 372
 - 11.2.2 实现变长数组 373
 - 11.2.3 参考JDK的实现 380
 - 11.3 散列表 (HashMap) 的实现 381
 - 11.3.1 实现前的思考 381
 - 11.3.2 实现散列表 381
 - 11.3.3 参考JDK的实现 389
- 第12章 方向 390
 - 12.1 算法的一些常用思想 391
 - 12.1.1 分治法 391
 - 12.1.2 动态规划 391
 - 12.1.3 贪心算法 391
 - 12.1.4 回溯法 392
 - 12.1.5 分支限界法 392
 - 12.2 新兴算法 392
 - 12.2.1 加密算法 392
 - 12.2.2 商业算法 393
 - 12.3 其他算法 393

12.3.1 基数估计算法 393

12.3.2 蚁群算法 394

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:www.tushu000.com