

# 《算法（第4版）》 阅读计划

——算法群阅读计划 (第 1 期)

领读人: Scruel

本书特色: 涉及常用算法, 均有Java代码的完整实现, 在算法和数据结构相结合, 在讲解的同时还举例在工程上的常见应用, 所以本书既可作为一本入门书, 也不失为一本可随取随看的算法应用手册。适合读者: 程序员

第一周答疑时间安排: 周日 18:00 - 20:00

《算法（第4版）》是一本较为易读的书籍, 定位于中级, 全书对于数学公式略有涉及, 大名鼎鼎的《算法导论》初学者可能没多久就放弃了（可能是晦涩难懂, 可能是.....）, 但同样是大部头的本书, 却并没有想象中的那么大部头, 作者对全书的一些概念尝试以更为易懂表达来方法, 图文并茂。并且本书资源丰富, 适合入门及自学。本阅读计划为总结而出, 旨在指导阅读, 会不断修改和完善, 欢迎提出您的宝贵意见。（图友们轻喷。。。一起进步啊咳咳。。。）

总阅读时间长度（预估）: 2 - 3 月

每天阅读用时: 2 小时

## 相关资源

### 1. [算法\(第4版\)官网](#)

可下载API、部分习题源代码、补充习题的源代码等

### 2. Coursera配套课程

[Algorithms, Part I](#)

[Algorithms, Part II](#)

### 3. [图灵社区勘误](#)

推荐在买来后首先到图灵社区对书籍勘误手工校对, 以免在阅读过程中遇到已经被解决的问题而耽误时间（当然也可自己尝试发现错误）。

## 阅读规划

### 第1章 基础

阅读时长: 1 ~ 2周

#### 1. Java 基础(1.1 - 1.3.1.3)

本部分内容对于有一定 Java 基础的同学来说可以较为轻松地应对, 学习中查漏补缺; 未曾接触过Java的同学同样可以此入门, 以已学语言为基础, 练习每一节的内容以及选择性练习课后习题, 学习 Java 语法, 但无需特别关注语言特性。

重点内容

1. 自己实现二分查找的递归和非递归做法（隔天）。

2. 了解使用 API 的目的、如何使用 API 以及定义 API 的注意事项。
3. 掌握Java 中的 String 字符串、数组操作。
4. 面向对象基础，学习封装、继承、多态三大特性。

#### 难点内容

1. 二分查找的递归及非递归实现
2. 面向对象基础

## 2. 数据结构及算法初步(1.3.1.3 后)

本部分开始正式涉及到算法部分，之前的部分务必要读懂读透，然后可在接下来的阅读中实战练习。其间仍有部分 Java 相关内容，遇到了就学习即可。因为，即使作者已经非常小心，尽量避免使用 Java 特性，但在遇到 Java 代码时，仍可能会有阅读障碍，尤其是面向对象部分，同时，希望大家在理解算法的基础上，用自己熟悉的语言写出算法。

#### 重点内容

1. 掌握并实现背包、栈、队列，掌握三者的特性及应用场景，了解 Java 中对应 API。
2. 理解算法分析中常见名词含义及用法。
3. 以实验方式分析和优化算法，学会比较算法优劣。
4. 内存管理。
5. union 并查集算法分析及优化。

#### 难点内容

1. 背包、栈、队列的数据结构特性及应用场景。
2. 算法分析。
3. union-find、quick-union 算法。

#### 补充

1. 《算法导论》第 3 章
2. 了解常见算法的复杂度
3. 尝试解决八皇后等经典递归-回溯问题

## 第2章 排序

排序算法是大部分算法的基础，有时候遇到看似无法解决的编程问题，一定会有人对你说，试试排序后再做，因此，掌握好排序算法尤为重要，由于排序算法实际上并不复杂，推荐所有内容都完成亲自实现。

阅读时长： 1 ~ 2周

#### 重点内容

1. 选择排序、插入排序、希尔排序
2. 归并排序
3. 快速排序
4. 优先队列、堆排序
5. 算法比较及分析

#### 难点内容

1. 三向切分的快速排序

2. 算法分析部分
3. 算法在具体场景中的选择及应用

注：这里的 3、4 点皆为为后部分章节中的难点，不再于后文难点内容部分列出。这两点并不易掌握。

## 第3章 查找

---

本章涉及到了树这一数据结构。本书的红黑树讲解由于图片分析详尽，可能会较其他书籍容易一些，但仍然会在理解和应用上遇到一些问题，树的掌握程度在一定程度上决定了图的学习曲线，尤其是图遍历部分。哈希散列表可以算是算法设计中最常用的一个工具，了解其实现可以帮助我们更好地应用该算法。

阅读时长： 2 周

### 重点内容

1. 符号表（关联性数组）的顺序及二分查找
2. 二叉查找树
3. 2-3 树及红黑树实现（平衡树）
4. 散列表

### 难点内容

1. 二叉查找树的删除
2. 红黑树实现
3. 散列表的几种实现形式

### 补充

1. hashcode 实现原理
2. 2-3-4 树实现，了解其他的平衡树
3. 树的前中后序遍历的非递归及递归实现

## 第4章 图

---

地图、电路、任务调度，生活中的图无处不在（嗯，你也可以画个图），而如何更快地在图中搜索和连接，则是长久以来人们一直在探究的问题，这一部分的消化吸收可能是最为耗时的，更需耐心。

阅读时长： 2 周

### 重点内容

1. 深度广度优先搜索
2. 连通分量
3. 有向图
4. 拓补排序
5. 最小生成树
6. 最短路径

### 难点内容

1. 有向图中的环
2. 图查找算法结合堆排序的应用
3. Prim 和 Kruskal 算法的区别及应用

#### 4. Dijkstra 算法

##### 补充

1. 了解 Java 中的 GC 机制
2. 了解 FloydWarshall、SPFA 等图论算法
3. 复习并查集算法

## 第5章 字符串

---

字符串操作是最常用的，正则表达式也同样常用，而数据压缩更是无处不在，所以要谈实用性，可能本章内容是最为实用，作者在本章也列出了大量的应用点。当然啦，作为第五章是有理由的，本章的内容需要坚实的基础。

阅读时长：2 周

##### 重点内容

1. KMP 算法
2. 前缀后缀数组
3. 字符串排序
4. 正则表达式
5. 数据压缩
6. 了解词典序

##### 难点内容

1. KMP 算法
2. 正则表达式及状态机
3. 霍夫曼压缩算法、LZW 算法

##### 补充

1. 学习 KMP 算法并实现
2. 了解 AC 自动机
3. 了解常见的压缩算法实现
4. 了解 Java 中字符串的 substring 方法的变化及原因
5. 了解数据加密

## 第6章 背景

---

本章开篇就介绍了算法各种方向，并带来了算法方面的更多应用，对于一些内容一笔带过也意味着需要我们自己去搜集相关资料，继续更为深度的学习。

阅读时长：1 ~ 2周

##### 重点内容

1. 算法无处不在
2. 后缀数组
3. 网络流算法
4. NP - 完全

##### 难点内容

1. B-Tree
2. 网络流相关算法

补充

1. 了解图灵机
2. [《迷茫的旅行商》](#)
3. 本章的粒子碰撞程序在官网中提供了源码，不妨可以下载运行一试。
4. 选择你的下一步算法之路。

## 其他建议

---

本书同其他计算机书籍甚至是所有书籍一样，都适合且需要多遍阅读，如果感觉本书的内容较为简单，可以选择《算法导论》等书（不仅限于纯算法类）继续学习。

本书并未涵盖数论以及动态规划等部分，而这些部分恰恰是算法中比较难于掌握的，建议在算法学习的基础上选择一些算法题练习，可选择阅读《挑战程序设计竞赛》、《算法竞赛入门经典》等书，继续学习、练习和巩固。

阅读本书后，对常见的数据库的实现会有一定的认识，对工程上的一些优化方案能有一定的见解，但这些还远远不够，算法之路充满挑战，且分支也很多，刷题、阅读论文，持续不断一步一步的学习，才可能一窥算法奥秘。

附卢晓南老师推荐的算法期刊（欢迎推荐）：

*Journal of ACM,*  
*ACM Trans. on Algorithms,*  
*ACM Trans. on Computation.*  
更偏数学的有  
*SIAM Journal on Computing,*  
*Algorithmica.*