

算法导论

xxx

October 27, 2018

译者序

我从 1994 年开始每年都为本科生讲授“算法设计与分析”课程，粗略地统计一下，发现至今已有 5 000 余名各类学生听过该课。算法的重要性不言而喻，因为不管新概念、新方法、新理论如何引人注目，信息的表示与处理总是计算技术（含软件、硬件、应用、网络、安全、智能等）永恒的话题。信息处理的核心是算法。在大数据时代，设计高效地算法显得格外重要。

当初，为了教好这门基础必修课，提高教学质量，我觉得应该从教学内容的改革入手，具体来说，采用的教程应该与国际一流大学接轨。1997 年访美期间，在 Stanford 大学了解到他们采用的教程是 Thomas H.Cormen 等人著的《Introduction to Algorithms》，于是从 Stanford 书店买了一本带回来，从第二年便开始改用该教材。至今，15 年过去了，我们一直追随其变迁。

前言

在计算机出现之前，就有算法。现在有了计算机，就需要更多的算法，算法是计算机的核心。

本书提供了对当代计算机算法研究的一个全面、综合的介绍。书中给出了多个算法，并对它进行了深入的分析，使得这些算法的设计和分析易于被各个层次的读者所理解。我们力求在不牺牲分析的深度和数学严密性的前提下，给出深入浅出的说明。

本书每一章都给出了一个算法、一种算法设计技术、一个应用领域或一个相关主题。我们强调将算法的效率作为一种设计标准，对书中的所有算法，都给出了关于其运行时间的详细分析。

本书主要供本科生和研究生的算法或数据结构课程使用。因为书中讨论了算法设计中的工程问题及其数学性质，所以，本书也可以供专业技术人员自学之用。

0.1 致使用本书的学生

希望本教程能为学生提供关于算法这一领域的有趣介绍。我们力求使书中给出的每一个算法都易于理解和有趣。为了在学生遇到不熟悉或比较困难的算法时提供帮助，我们逐个步骤地描述每一个算法。

本书是一本大部头著作，学生所修的课程可能只讲授其中的一部分。我们试图使它能成为一本现在对学生有用的教材，并在其将来的职业生涯中，也能成为一本案头的数学参考或工程实践手册。

Contents

译者序	iii
前言	v
0.1 致使用本书的学生	v
1 基础知识	1
2 算法基础	3
2.1 插入算法	3

Chapter 1

基础知识

Chapter 2

算法基础

本章将介绍一个贯穿本书的框架，后续算法设计与分析都是在这个框架中进行的。这一部分内容基本上是独立的，但也有对第 3 章和第 4 章中一些内容的引用。

首先，我们考察求解第 1 章中引入的排序问题的插入排序算法。我们定义一种对已经编写过计算机程序的读者来说应该熟悉的“伪代码”，并用它来表示我们将如何说明算法。然后，在说明了插入排序算法后，我们将证明该算法能正确地排序并分析其运行时间。这种分析引入了一种记号，该记号关注时间如何随着将被排序的项数而增加。在讨论完插入排序之后，我们引入用于算法设计的分而治之，并使用这种方法开发一个称谓归并排序的算法。最后，我们分析归并排序的运行时间。

2.1 插入算法

我们的第一个算法求解第 1 章引入的排序算法：

输入： n 个数一个序列 $\langle a_1, a_2, \dots, a_n \rangle$ 。

输出：输入序列的一个排序 $\langle a'_1, a'_2, \dots, a'_n \rangle$ 。

我们排序的数也称为关键词。虽然概念上我们在排序一个序列，但是输入是以 n 个元素的数组的形式出现的。

伪代码与真代码的另一个区别是伪代码通常不关心软件工程的问题。为了更简洁地表达算法的本质，常常忽略数据抽象、模块性和错误处理的问题。

```
INSERTION-SORT(A)
  for j = 2 to A.length
    key = A[j]
    //Insert A[j] into the sorted sequence A[1..j - 1].
    i = j - 1
    while i > 0 and A[i] > key
      A[i + 1] = A[i]
      i = i - 1
    A[i + 1] = key
```

