

全国高等教育自学考试应试指导丛书
中国计算机函授学院图书编写中心 组编

最新版

计算机及应用
（本科）

数据结构 自考应试指导

胡学钢 主编



南京大学出版社

中国计算机函授学院图书编写中心 组编

全国高等教育自学考试应试指导丛书

计算机及应用专业(本科)

数据结构自考应试指导

主 编 胡学钢

南 京 大 学 出 版 社

内 容 简 介

“数据结构”是计算机专业重要的专业基础课程,是提高软件设计水平以及学习后续课程所必需的基础。课程中涉及到软件设计中常见的几种数据结构及其在计算机内存中的表示(或存储)和各种操作的实现,以及软件设计中常用的排序和查找运算。

本书是针对全国自学考试计算机及应用专业《数据结构》(黄刘生主编)教材编写的辅导材料,按原教科书的章节对各部分内容展开辅导,包括线性表、栈、队列、串、数组和广义表、树和二叉树、图、排序、查找和文件等。各章中除了采用通俗易懂的语言系统地介绍有关概念、方法外,还通过大量例题及其分析展示了课程中内容的应用及其学习方法,因而容易激发学生的学习兴趣,具有较好的学习效果。最后部分给出了涉及面广泛的模拟考试试卷以供学生自测,并提供了详细的分析和讲解。

本书除了可以作为全国计算机及应用专业(独立本科段)自学考试的辅导教材外,也可以作为其他学习“数据结构”课程的辅导教材。

图书在版编目(CIP)数据

数据结构自考应试指导/胡学钢主编. —南京:南京大学出版社,2004.4

ISBN 7-305-04259-5

I. 数... II. 胡... III. 数据结构-高等教育-自学考试-自学参考资料 IV. TP311.12

中国版本图书馆 CIP 数据核字(2004)第 035170 号

书 名 数据结构自考应试指导
主 编 胡学钢
责任编辑 史德芬
出版发行 南京大学出版社
地 址 南京汉口路 22 号 邮编 210093 电话 025-83593695
印 刷 合肥学苑印务公司
经 销 全国各地新华书店
开 本 787×1092 1/16 印张 15 字数 355 千字
版 次 2004 年 5 月第 1 版 2004 年 5 月第 1 次印刷
定 价 19.00 元
书 号 ISBN 7-305-04259-5/TP·278

声明:(1)版权所有,侵权必究。

(2)本丛书若有质量问题,可向经销商调换。

组编前言

国家教育部考试中心决定,从 2000 年开始,全国高等教育自学考试正式使用新编的大纲和教材。

为适应新调整的考试计划及密切配合新大纲、新教材开展助学辅导,中国计算机函授学院利用多年积累的自考教学辅导资源和经验,全面系统地剖析了有关各门课程新大纲和教材的内容体系,重新组织编写了一套《全国高等教育自学考试应试指导丛书》推向全国,以满足考生之急需,适应社会之需要。

这套丛书堪称“通关必读”,丛书的作者在书中融入了自己多年从事自考教学辅导的直接经验,他们既是本专业的教授,又是自考辅导的专家,二者集于一身,使该套丛书极具实用性和针对性。他们精心组织、细心筹划、用心编撰,从而确保该套丛书质量上乘。

编写该套丛书的指导思想是切实解决考生自学应试中的三个问题:

- (1)在自学过程中起到答疑解惑作用,帮助考生顺利阅读,掌握教材内容;
- (2)帮助考生抓住课程重点、难点,不入迷津;
- (3)帮助考生理清课程主线,建立清晰的知识结构体系,在掌握知识点的前提下,沉着应战,顺利过关。

对于广大应试者而言,请一位好“教师”,找一位好“辅导”,尤为重要,这套《自学考试指导丛书》,可望成为你攻克一门又一门课程、克服一个又一个难关的良师益友,帮助你扫清学习中的障碍,增强你的必胜信心,伴随你走向成功的彼岸。

我们真诚地为广大考生奉献这份精品、真品。愿广大考生早成夙愿!

中国计算机函授学院
图书出版中心

编者的话

全国开考的计算机及应用专业(独立本科段),为广大学子掌握计算机科学与技术领域相关知识的读者提供了一个重要途径。然而,要想实现既定的目标,通过自学完成各门课程的学习是需要付出艰辛劳动的,因为很少有课程能轻易掌握。

在计算机及应用专业(独立本科段)的课程中,“数据结构”是重要的专业基础课程,是提高软件设计水平以及学习后续课程所必需的基础。课程中介绍了软件设计中常见的线性表、栈和队列、串、数组和广义表、树和二叉树、图、文件等数据结构及其在计算机内存中的存储结构和各种操作的实现,介绍了软件设计中常用的排序和查找方法,并讨论有关运算的性能。通过对这些内容的学习,将使学子能熟练地掌握各种常用结构的特性,各种运算的实现方法及其性能,并能在实际应用中根据具体问题的要求设计出合理的数据结构和运算。

然而,由于该课程中的内容多,而且许多内容较抽象,特别是其中大量的算法以及所用到的递归技术,以及缺乏有效的实验条件,使学习数据结构课程的难度较大。对于以自学为主的考生来说,学习数据结构课程更是困难重重。虽然教材较多,但由于受篇幅所限,效果不理想。

本书是针对全国自学考试计算机及应用专业(独立本科段)《数据结构》(黄刘生主编)教材编写的辅导材料。在接受编写任务之后,笔者有些惶恐,唯恐耽误了读者的学习。为了能对得起读者,笔者就编写风格、内容选择、深度及广度等方面作了较长时间的考虑,并和编写组几位同事作了认真的讨论,结合以往的教学实践和辅导自学考试考生的经验,终于编写出了现在的这本辅导书。尽管如此,仍对此书的效果不敢说有十分的把握。

本书总体上是按原教科书的章节对各部分内容展开辅导。各章中除了采用通俗易懂的语言系统地介绍有关概念、方法外,还通过大量例题及其分析展示了课程中内容的应用及其学习方法,因而容易激发学子的学习兴趣,收到良好的学习效果。这是本书的特色之一。最后部分给出了涉及面广的模拟考试试卷以供学子自测,并提供了详细的分析和讲解。全书的主要内容如下:

第1章介绍了数据结构课程的研究内容、特点以及一些所需的基础知识和学习方法,有助于对整个课程的学习。

第2章介绍了线性表的逻辑结构和运算,重点讨论了顺序表和链表结构及其基本运算实现,并给出了大量例题及分析讲解。

第3章介绍了栈、队列这两种结构的逻辑结构和运算,顺序栈、链栈、顺序(循环)队列和链队列及其基本运算实现。

第4章介绍了串的有关概念、运算和存储结构等内容。

第5章介绍了多维数组的逻辑结构、运算、顺序存储和压缩存储等内容,介绍了广义表的概念、运算和存储结构等。

第6章介绍了树的逻辑结构、存储结构和运算,介绍了二叉树的概念、性质、存储结构,

重点讨论了二叉树的遍历运算方法及其应用,讨论了线索二叉树的概念、用途和有关运算,并给出了相应的例题及分析,另外还介绍了哈夫曼树的有关知识及其应用。

第7章介绍了图的有关概念,介绍了两种常用的存储结构,重点讨论了两种遍历算法的方法和应用,最后以通俗的方式介绍了图的几个应用算法。

第8章重点讨论了各种排序的方法、算法及其性能分析。

第9章所介绍的查找是软件设计中常用的基本运算,本章讨论了在顺序表、树表和散列表等几种表结构上的查找方法实现及其性能。

第10章对有关文件方面的基本知识作了简要介绍。

最后给出了四套模拟试卷以供考生自测,每套模拟考试试卷都有40题左右,涉及面大,是考生自测及复习的重要内容,其后的试卷分析则更详细、深入地介绍了所学知识及其应用的方法。

本书由几位作者共同合作完成,具体分工如下:

合肥工业大学计算机与信息学院胡学钢负责全书的策划、组织,并编写第1章、第2章、第4章至第7章、第10章和模拟试卷及解析,同时对全书作了审校;张晶编写第3章;周红鹃编写第8章;阙夏编写第9章。

本书除了可以作为全国计算机及应用专业(独立本科段)自学考试的辅导教材外,也可以作为其他学习“数据结构”课程的辅导教材。

由于水平及时间所限,书中一定存在错误和不足之处,敬请批评指正。

编者

2004年4月

目 录

第一部分 内容概要与典型题解	(1)
第1章 概论	(2)
1.1 本章内容概述	(2)
1.2 “数据结构”的作用、特点及学习方法	(2)
1.3 本章内容的学习和辅导	(5)
1.4 复习题	(9)
第2章 线性表	(10)
2.1 概述	(10)
2.2 线性表的定义和运算	(11)
2.3 顺序表	(13)
2.4 链表	(24)
第3章 栈和队列	(39)
3.1 栈	(39)
3.2 队列	(42)
第4章 串	(49)
4.1 串的定义和运算	(49)
4.2 串的存储和运算实现	(49)
第5章 多维数组和广义表	(52)
5.1 数组	(52)
5.2 广义表	(55)
第6章 树	(59)
6.1 概述	(59)
6.2 树和二叉树的基本内容	(60)
6.3 二叉树的遍历	(67)

6.4 线索二叉树	(84)
6.5 树和森林	(92)
6.6 哈夫曼树	(100)
第7章 图	(107)
7.1 概述	(107)
7.2 图的基本概念和存储结构	(108)
7.3 图的遍历算法及其应用	(113)
7.4 图的应用	(125)
第8章 排序	(140)
8.1 基本内容及学习方法	(140)
8.2 插入排序	(141)
8.3 交换排序	(145)
8.4 选择排序	(150)
8.5 归并排序	(156)
第9章 查找表	(158)
9.1 基本内容及学习方法	(158)
9.2 顺序表的查找	(159)
9.3 树表的查找	(164)
9.4 散列表的查找	(172)
第10章 文件	(176)
10.1 概述	(176)
10.2 常见文件组织形式	(177)

第二部分 模拟试卷与参考答案

模拟试卷一	(181)
模拟试卷一参考答案	(183)
模拟试卷二	(194)
模拟试卷二参考答案	(197)
模拟试卷三	(208)

模拟试卷三参考答案	(210)
模拟试卷四	(217)
模拟试卷四参考答案	(220)

第一部分

内容概要与典型题解

在这一部分中,以考试大纲规定的考核知识点为纲,用最简捷的文字简明扼要地阐述了各知识点的基本概念、原理和方法,并围绕相关知识点组织了大量典型例题,以增强读者对概念的理解和提高解题能力。

读者可将这部分内容作为复习提纲来使用,它针对性强,能帮助考生从繁杂的内容中理清头绪,从而在复习迎考冲刺阶段做到事半功倍。

第 1 章 概 论

1.1 本章内容概述

本章是对“数据结构”课程的一个概述,主要介绍与整个课程有关的概念、术语、算法及其描述语言和算法分析等。由于本章是学习后续内容所需要的,因此,尽管考试大纲只要求达到“了解”,但为了后面内容学习的需要,还是希望大家能够深入理解。虽然本章是入门性的内容,但由于概念较多,并且相对较抽象,因此,还是有一定的学习难度。

考虑到本章内容及整个课程的特点,在本章的辅导中将介绍以下一些内容:

① 整个课程的作用、特点及学习方法。如果能了解课程的特点和作用,将对学习有很大的帮助。

② 本课程的重点、难点及学习方法简介。

③ 如何理解本章的有关概念。

④ 算法描述语言和分析。

通过对这些内容的学习,希望使初学者不仅能理解“数据结构”课程的研究内容、在计算机领域中的作用和特点,还能理解有关的概念和术语,掌握算法的描述和分析的方法,为后续课程的学习打好基础。

1.2 “数据结构”的作用、特点及学习方法

1. “数据结构”课程在计算机专业中的作用

“数据结构”这门课程有什么作用?如何学好这一课程?怎样才算学好了?下面从运用计算机解决实际问题的过程来谈谈本课程在计算机领域,尤其是软件设计中的作用。

(1) 用计算机解决实际问题的过程

在用计算机解决一个实际问题时,通常要涉及到以下几个步骤:

① 建立模型:一般情况下,由于将要面临的实际应用问题可能会千差万别,如大家所熟悉的工资表的处理问题,学生成绩管理问题,电话号码查询问题,从一组数据中选择若干元素的排列、组合,在国际象棋棋盘上放置八个皇后,以使相互间不被吃掉的八皇后问题等。这些问题无论是所涉及到的数据方面,还是其操作要求方面可能都存在一定的差异。尽管如此,许多应用问题之间还是具有一定的相似之处。例如,虽然工资表和学生成绩表的具体

信息(栏目)不同,但如果将两个表中的每个人的工资信息和成绩信息看作一个整体,则这两个表结构之间就具有了某些共性。另外,从操作方面来看,虽然对这两个表的操作肯定存在差异,但也一定存在一些相同的基本操作。例如,查询一个人的工资信息和成绩信息,修改有关信息等,类似的例子有很多。

正因为许多不同的问题之间存在着某些共性,因而可以将一个具体的问题用这些共性的形式描述出来,这就是通常所说的建立模型。建立问题的模型通常包括所描述问题中的数据对象及其关系的描述、问题求解的要求及方法等方面。建立问题模型有这样的好处:因为所涉及到的许多基本模型在有关的课程中已有介绍,因而通过建立模型,就可以将一个具体的问题转换为所熟悉的模型,然后借助于这一模型来实现。“数据结构”、“离散数学”等许多计算机专业的课程中以及许多数学课程中就介绍了许多模型。例如,要描述一个群体中个体之间的关系时,可以采用“数据结构”和“离散数学”中所介绍的图结构。要描述一个工程内的关系或进展情况时,可以采用“数据结构”中所介绍的 AOV 网或 AOE 网等。即使没有现成的模型或算法可以用,构思问题模型以及研究问题求解也是计算机科学与技术领域的一个重要的工作。

② 构造求解算法:在建立了模型之后,一个具体的实际应用问题就转变成了一个用模型所描述的抽象问题。借助于这一模型以及已有的知识(如数据结构中有关图结构的基本知识),可以相对容易地描述出原问题的求解方法,即算法。从某种意义上说,该算法不仅能实现原问题的求解,而且还可能实现许多类似的具体问题的求解,尽管这些具体问题的背景及其描述形式可能存在较大的差异。

③ 选择存储结构:在构造出求解算法之后,就需要考虑在计算机上实现求解了。为此,需要做两方面的工作,其一是选择合适的存储结构,以便将问题所涉及到的数据(包括数据中的基本对象及对象之间的关系)存储到计算机中。不同的存储形式对问题的求解实现有较大的影响,所占用的存储空间也可能有较大的差异。

④ 编写程序:在选择了存储结构之后,就可以做实现求解的另一项工作,即编写程序了。存储形式和问题要求决定了编写程序的方法。

⑤ 测试:在编写出完整的程序之后,需要经过测试才能交付使用。

(2) “数据结构”在软件设计中的作用

在介绍了用计算机求解问题的过程之后,下面来看看“数据结构”课程与此有什么关系。事实上,本课程涉及到上述求解过程中的大多数步骤:

① 与建立模型的关系:“数据结构”课程中介绍了许多基本的数据结构模型及其运算实现。例如,线性表、栈和队列、树和二叉树、图、二叉排序树、堆等。通过学习,不仅可以掌握这些基本内容及其应用,还能根据实际问题选择或设计合适的模型。

② 与算法设计的关系:课程中对每种结构都讨论了相应的基本运算的实现,并且其中的许多算法是非常经典的,掌握这些基本运算的实现方法有助于进行更为复杂的算法设计。

③ 与选择存储结构的关系:课程中对每种结构都讨论了其具体存储结构及其对运算实现的影响。例如,在第2章中所介绍的对顺序表作插入和删除运算,平均需要移动表中一半的元素,而采用链表结构则不需要移动元素。通过对这些内容的学习和比较,可使学生熟练地选择合理的存储结构。

④ 与编程之间的关系:在实现各结构的算法时,涉及到许多具有代表性的设计方法。

通过对这些方法的学习,有助于编程技术的提高。

综上所述,“数据结构”对提高软件设计水平有较大的影响。也正因为如此,这一课程在计算机专业课程中具有极其重要的作用。几乎绝大多数学校和研究单位的计算机专业硕士研究生入学考试都将这一课程定为考试课程之一。

2. “数据结构”的主要内容和特点

“数据结构”是计算机专业重要的专业技术基础课程。在该课程中,主要介绍软件设计中常用的基本技术,主要包括软件设计中所涉及到的各种数据结构的结构、存储实现及其运算等。通过这些内容的学习,使学习者不仅能掌握这些数据结构的组织形式,而且还能掌握在各种存储形式下实现运算的方法以及各种存储形式实现运算的性能特点和影响。由此可见,如果熟练地掌握了本课程所介绍的这些内容,不仅可以有效地提高程序设计的技术,而且对后续课程的学习有很大的帮助。

然而,本课程无论从内容的数量还是难度上都使得课程的学习需要花费较多的精力:

① 由于有多种数据结构,每种数据结构有多种存储形式,每一存储形式对运算的实现可能存在较大的差异,因此,需要掌握的内容较多。

② 因为课程中所涉及到的许多递归(递归定义和递归调用)以及动态变量是初学者所头疼的,因此,要真正掌握这些内容还是有较大的难度。

由此可见,使用适合于本课程的学习方法是非常必要的。

3. “数据结构”的学习方法

如前所述,本课程的内容较多,需要掌握的内容不仅较多,而且还具有一定的难度,因此需要有合适的方法。下面就有关的学习方法谈点自己的观点供参考。

(1) 了解本科段与专科段之间的差异

参加独立本科段的考生应该是经历了专科阶段学习的。与专科层次相比,计算机本科层次的要求自然要高,就“数据结构”课程而言,不仅要掌握大纲所涉及到的各知识点,还要领会其思想,以便能对给定的实际问题综合运用所学知识进行求解,设计出符合要求的数据结构和算法。

(2) 正确理解大纲

考试大纲是考试、命题的依据,因此考生要严格按照大纲的要求进行学习。通过阅读大纲,可以了解到课程的性质、任务、重点、难点及与其他课程之间的关系,从而做好必要的准备,为后面的学习奠定基础。对照大纲,考生不仅可以在学习时知道自己该学什么、学到什么程度,各项内容的分解等,而且还可以在复习时作为复习提纲和标准,以找出自己存在的问题。因此,考试大纲是整个学习阶段必不可少的学习工具。

(3) 掌握程序设计语言

按照所指定的教材,本课程主要介绍各种数据结构及其运算实现,因此要求考生必须掌握描述数据结构和算法的工具——程序设计语言及编程技术。按指定教材的要求,需要掌握 C 语言。

(4) 注意学习内容的联系

虽然本课程的学习内容较多,但如果组织得合理,则可以较方便地掌握。在学习每种结

构,特别是在复习时,应注意以下几个方面的内容及其联系:

- ① 逻辑定义及相关的术语。
- ② 在这种结构上所定义的运算。
- ③ 如何用程序设计语言中的结构来存储结构?
- ④ 在所选用的存储结构上,如何实现所定义的运算?
- ⑤ 在选定的结构上所实现的各算法的性能如何(主要是时间性能,在有些情况下还要涉及到空间性能)?

(5) 注意实践

及时做习题,以理解和巩固所学的内容。在编制算法时,应能根据实际问题的要求提出尽可能合理的数据结构,设计出有效的算法,并注意一题多解及比较各解之间的性能。这样既可以巩固所学内容,又能提高解决实际问题的能力,因而是必不可少的重要环节。

由于本课程中算法的执行过程较复杂,因此按传统方法阅读算法有一定的困难,这给验证算法的正确性增加了难度,从而影响到学习的效果。为此,建议有条件的考生尽可能多地在机器上验证算法。

(6) 及时复习和总结

每当一个完整的内容(指节、章及整个课程)学习完成时,要注意及时复习和总结。复习时,可按大纲所列出的要点进行。在学完每一章、节后,应按上述的线索作总结对比,学完全书后还要作全面系统的总结。

(7) 适当选用参考书

适当参考其他教材和辅导用书将是对自己学习的一个补充,有助于课程的学习。目前市场上有不少参考书,一般来说,各参考书籍的特点不同,但并非都适合于你,因此,需要有所选择。

1.3 本章内容的学习和辅导

1. 有关概念的理解

在教材第1章中介绍了许多概念,由于概念较多,并且对初学者来说较抽象,因而不易理解。下面就来介绍这些概念的学习。

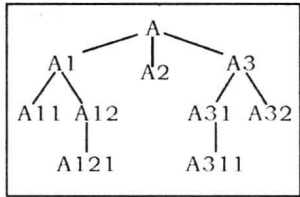
如教材所述,数据是指信息的载体,能被计算机识别、存储和加工处理。数据的形式较多,如前面提到的工资报表、学生成绩表,一个家族关系的表示形式,表示一个群体中个体之间关系的图形描述等,如图1-1所示。

编 号	姓 名	基本工资	奖 金	...	...

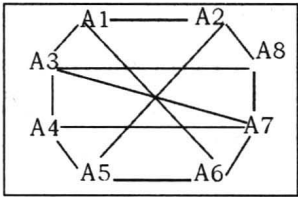
(a) 工资表示例

序号	学号	姓名	成绩	备注

(b) 成绩表示例



(c) 家族关系示例



(d) 群体间关系示例(连线表示相互认识关系)

图 1-1 数据示例

虽然这些数据的形式及运算存在较大的差异,但可以找出其中的共性部分:每一数据都是由多个具有独立意义的个体所组成的,每一数据中的个体之间存在着某些关系,对这些数据的运算也有一些相似部分。例如,在家族关系数据中,组成数据的基本个体是每个人,其中的人与人之间存在着多种关系,如父子关系、兄弟关系、祖先-后代关系等,其中有些关系是直接表示出来的,还有一些关系则是隐含的。对家族关系数据,通常要涉及到查询特定个体间的关系,插入和删除个体等。其他数据具有相似的组成,以及包含多个成员,成员间具有一定的关系,对这些数据需要进行特定的运算。

“数据结构”主要讨论这些具有共性的内容。为便于讨论,将这些共性部分分别给出有关描述,得到下面的概念:

① 数据元素:数据中具有独立意义的个体。例如,工资表中的每个人,成绩表中的每个学生,家族关系中的每个人等。在有些场合下,也称之为元素、记录、结点、顶点等。

② 字段(域):虽然将具有独立意义的个体用元素来表示,但在许多情况下还需要特定个体的具体信息,因而涉及到了元素的字段信息。字段是对元素的详细描述,通常情况下,元素包含多个字段信息。例如,在图 1-1 所示的成绩表中,每个元素包括序号、学号、姓名、成绩、备注五个字段,分别描述了每个学生在表中的序号,学生的学号,学生的姓名、成绩及备注信息。

③ 数据结构:在图 1-1 所示的几个数据示例中,各元素之间具有一定的构成形式,这些构成形式被称为数据结构。数据结构是指组成数据的元素之间的结构关系。在图 1-1 中,(a)、(b)中的元素依次排列,构成线性关系;(c)中的元素是按树的形式构成树型结构;(d)中的元素构成图结构。线性结构、树型结构和图结构是“数据结构”中要介绍的几类常见的数据结构。如果数据中的元素之间没有关系,则构成集合,这也是一种结构。

通常,称这几类结构为逻辑结构,因为仅考虑了元素之间的逻辑关系,而没有考虑到其

在计算机中的具体实现。

在用计算机解决涉及到数据结构的问题时,需要完成如下任务:

① 数据结构的存储:为所涉及到的数据结构选择一种存储形式,并将其存储到计算机中,这样就得到了数据结构在内存中的存储结构(有时也称为物理结构)。在教材的后面部分,你将看到每种逻辑结构可能会有多种存储结构。

② 数据结构运算的实现:在选择了数据结构的存储结构之后,就可以实现所给出的运算了。在“数据结构”课程中,运算的实现一般是以算法的形式给出的。在以 C 语言作为算法描述语言的教材中,算法大多以函数的形式给出。

由此可见,对一种数据结构,需要涉及到其逻辑结构、存储结构和运算三个方面。在以后的学习中,对每种结构都要注意这三方面的联系。为便于学习,下面将这三个方面展开为图1-2所示。



图 1-2 数据结构的几个方面及联系

由图 1-2 可知,逻辑结构是由定义给出的,运算则取决于逻辑结构;在选择了一种存储结构之后,需要在该存储结构上实现所给出的运算,从而得到了算法。

由于不同的存储形式对算法的时间性能、空间性能等有较大影响,即使是相同的存储结构也可能存在不同的算法实现,这就需要解决这样的问题:究竟是何种存储结构更为合适? 什么算法更有效? 这就需要对算法进行分析,这就是图 1-2 中最后所列出的内容,这将在后面作介绍。通过分析,可以知道所实现的算法的性能及所选择的存储结构是否符合要求。

2. 算法描述及分析

如前所述,对数据结构的运算是以算法的形式来描述的。关于算法的定义,由于教材中已给出了一个粗略的定义,因而此处不再讨论。读者应理解算法就是一段程序,该程序段对给定的输入可在有限的时间内产生出确定的输出结果。

(1) 算法描述语言

算法可采用多种描述语言来描述,如自然语言、计算机语言或某些伪语言。各种描述语言对问题的描述能力上是存在差异的。例如,自然语言较为灵活,但不够严谨,而计算机语言虽然严格,但由于语法方面的限制,使得灵活性不足。因此,许多教材中采用的是以一种计算机语言为基础,适当添加某些功能或放宽某些限制而得到一种类语言,这些类语言既具有计算机语言的严格性,又具有其灵活性,同时也容易上机实现,因此被广泛接受。目前的许多《数据结构》教材就是采用的类 PASCAL、类 C++ 或类 C 语言作为算法描述语言。本教材选用的是 C 语言作为算法描述语言,所涉及到的算法是以 C 的函数形式给出的。为充分理解教材中的算法,以及为了能熟练地写出问题求解算法,要求读者要熟练掌握所采用的描述语言。

(2) 算法分析

如前所述,为了掌握算法的有关性能,需要对算法进行分析。通过分析,不仅可以知道

算法的有关性能,而且由此还可知道所选择的存储结构是否符合要求。

在“数据结构”中,大多数算法分析是针对算法的时间性能进行的。算法的时间性能是以时间复杂度来衡量的。所谓算法的时间复杂度是指运行算法所需时间。很显然,为了使算法的时间复杂度便于比较,不宜采用在某个具体机器上所运行的时间的形式来表示。对此,一般是以算法中基本语句的执行次数来衡量的。然而,在实际应用时,基本语句的执行次数的精确计算是困难的,同时也是不必要的,因为许多算法中的语句的执行次数要取决于输入数据,可能会有多种复杂的情况。为便于计算,对时间复杂度大多采用一种近似的形式来描述,即采用基本语句执行次数的数量级来表示。所谓数量级是这样定义的:如果变量 n 的函数 $f(n)$ 和 $g(n)$ 满足: $\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = \text{常数 } k (k \neq \infty, 0)$, 则称 $f(n)$ 和 $g(n)$ 是同一数量级的,并用 $f(n) = O(g(n))$ 的形式来表示。

由定义可知,两个函为同一数量级,强调的是在 n 趋向无穷大时,两者“接近”。例如, $\frac{n(n+1)}{2}$ 和 n^2 是同一数量级的,因为 $= \lim_{n \rightarrow \infty} \frac{\frac{n(n+1)}{2}}{n^2} = \frac{1}{2}$ 为一个常数。

通过求解算法中语句执行次数的数量级所得到的时间复杂度,忽略了其中的“细微”部分,得到了时间性能的近似描述,这一描述有助于对算法时间性能的简要了解。

【例 1.1】求解下列各程序段的时间复杂度。

① for ($i = 1; i < n; i++$) { $x++$; }

【解】虽然从循环语句的内部执行来看,该程序段中基本语句的执行数多于循环体的执行次数,但两者是同一数量级的。因此,该问题的求解可以以其循环体的执行次数的数量级的求解来实现。由于循环体执行 $n-1$ 次,因此其时间复杂度为 $O(n)$ 。

② for ($i = 1; i < n; i++$)
 for ($j = 1; j \leq i; j++$) { $x++$; }

【解】该问题同样以其循环体的执行次数的数量级求解来实现。由于是双重循环,并且内层循环的循环次数不是常数,因此其计算稍微有些麻烦。计算方法如下:

$i = 1$ 时,内层循环执行 1 次($j = 1$ 到 1);

$i = 2$ 时,内层循环执行 2 次($j = 1$ 到 2);

$i = 3$ 时,内层循环执行 3 次($j = 1$ 到 3);

.....

$i = n-1$ 时,内层循环执行 $n-1$ 次($j = 1$ 到 $n-1$);

因此,最内层循环体共执行 $n(n-1)/2$ 次,因此其时间复杂度为 $O(n^2)$ 。

③ $i = 1;$
 while ($i < n$) $i * = 2;$

【解】对许多初学者来说,该问题的求解有些麻烦,求解方法如下:

由语句可知,循环次数和 i 之间的对应关系如下:

次数	1	2	3	4	...	...	k
I 值	2^1	2^2	2^3	2^4	...	...	2^k

假设最后一项即 $2^k = n$, 则可知 $k = \log_2 n$ 。由此可知,该语句段的时间复杂度为 $O(\log_2 n)$ 。