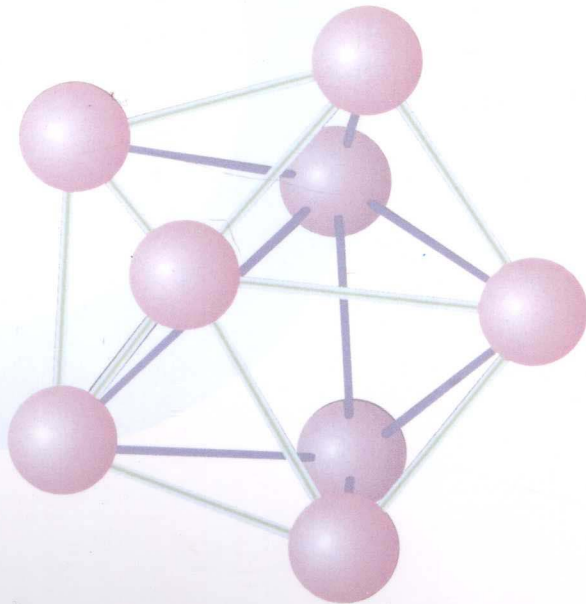




高等职业院校规划教材·计算机应用技术系列

数据结构学习指导与实训

李 波 张亦辉 主编



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE



高等职业院校规划教材·计算机应用技术系列

数据结构学习指导与实训

李 波 张亦辉 主编

周朋红 李婷婷 牛文琦 李 琳 参编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书是《数据结构》(李波、张亦辉主编)的配套学习指导教材,融入了山东省“数据结构”精品课程在教学方法、内容上的先进思想,与主教材配合使用,是主教材的扩展和补充。

全书共分8章,主要内容包括顺序表、链式表、栈、队列、串、多维数组、广义表、树、图、查找和内部排序。除第1章和第8章外,每章内容都含有“学习目标”、“本章要点”、“典型例题”、“补充习题”、“综合实训”几个部分。通过“典型例题”和“补充练习”可以提高学生的理论水平;而通过“综合实训”则可以让学生对本章内容有一个更加系统的认识,促进学生由重理论知识轻动手实践到理论与实践并重的转变。

本书适合高职高专计算机相关专业的学生使用。

图书在版编目(CIP)数据

数据结构学习指导与实训/李波,张亦辉主编. —北京:
中国铁道出版社,2009.10
(高等职业院校规划教材. 计算机应用技术系列)
ISBN 978-7-113-10668-3

I. 数… II. ①李…②张… III. 数据结构—高等学校:
技术学校—教学参考资料 IV. TP311.12

中国版本图书馆CIP数据核字(2009)第188363号

书 名: 数据结构学习指导与实训

作 者: 李 波 张亦辉 主编

策划编辑: 秦绪好 翟玉峰

责任编辑: 翟玉峰

编辑部电话: (010) 63583215

编辑助理: 邱雪姣

封面设计: 付 巍

封面制作: 李路

责任印制: 李 佳

出版发行: 中国铁道出版社(北京市宣武区右安门西街8号 邮政编码: 100054)

印 刷: 河北省遵化市胶印厂

版 次: 2009年12月第1版 2009年12月第1次印刷

开 本: 787mm×1092mm 1/16 印张: 9.75 字数: 237千

印 数: 4 000册

书 号: ISBN 978-7-113-10668-3/TP·3603

定 价: 16.00元

版权所有 侵权必究

本书封面贴有中国铁道出版社激光防伪标签,无标签者不得销售

凡购买铁道版的图书,如有缺页、倒页、脱页者,请与本社计算机图书批销部调换。

高等职业院校规划教材·计算机应用技术系列

主 任：石 冰

副主任：徐 红 张亦辉

委 员：（按姓氏音序排列）

安丰彩	崔凤磊	黄丽民	李 敏	梁国浚
刘 学	毛红旗	彭丽英	曲桂东	孙学农
王明晶	王兴宝	王秀红	王宜贵	王 玉
徐新艳	燕居怀	杨 云	张序政	赵 敬
赵吉兴				

出版说明

自 2002 年全国职业教育工作会议以来, 全国各地区、各部门认真贯彻《国务院关于大力推进职业教育改革与发展的决定》(国发[2002] 16 号), 加强了对职业教育工作的领导和支持, 以就业为导向改革与发展职业教育逐步成为社会共识。2005 年, 在北京召开的全国职业教育工作会议上, 国务院总理温家宝提出, 在今后一个时期, “教育结构调整总的方向是, 普及和巩固义务教育, 大力发展职业教育, 提高高等教育质量”, “把基础教育、职业教育和高等教育放在同等重要位置”。此次讲话精神将职业教育的地位提到了一个新的高度, 为大力发展职业教育奠定了思想基础, 指明了方向。

作为高等职业教育的重要组成部分, 计算机教育和教学也面临着“以就业为导向”的重要转变和改革。为顺应高等职业教育改革和发展的趋势, 配合高等院校的教学改革和教材建设, 中国铁道出版社联合全国知名职业教育专家和各大职业院校推出了《高等职业院校规划教材》系列丛书。

本套系列教材编写的主要指导思想:

- (1) 定位明确。整套教材贯穿了“以就业为导向”的思想, 面向就业, 突出实际应用。
- (2) 内容先进。教材合理安排经典知识和实际应用的内容, 补充了新知识、新技术和新设备。
- (3) 取舍合理。以高等职业教育的培养目标为依据, 注重教材的科学性、实用性和通用性, 尽量满足同类专业院校的需求。
- (4) 体系得当。以岗位职业标准为依据设计教材的体系, 体现岗位技能要求, 紧密结合生产实际, 强化实践环节, 培养创新精神。
- (5) 风格优良。在编写方式和配套建设中体现建设“立体化”精品教材体系的宗旨。为主要课程配套了电子教案、教学大纲、学习指导、习题解答、素材库、案例库、试题库等相关教学资源。

本套教材在编写过程中参考了《中国高职院校计算机教育课程体系 2007》(英文简称为 CVC 2007) 中各专业课程体系的参考方案, 并根据专业类别划分系列, 分为计算机应用技术系列、信息管理技术系列、多媒体技术系列、网络技术系列、软件技术系列、电子商务系列等若干子系列。在本系列丛书的编写和出版过程中, 得到了各专业领域知名职业教育专家以及全国各大高等职业院校的大力支持, 在此表示衷心感谢。希望本系列丛书的出版能为我国高等职业院校计算机教育改革起到良好的推动作用, 欢迎使用本系列教材的老师和同学提出意见和建议, 书中如有不妥之处, 敬请批评指正。

中国铁道出版社



“数据结构”是计算机程序设计的重要理论知识,是计算机相关专业必修的一门重要基础课程和核心课程。本书是由中国铁道出版社出版的“高等职业院校规划教材·计算机应用技术系列”《数据结构》(张亦辉、李波主编)的配套学习指导教材,也是山东省“数据结构”精品课程的立体化教材,主要面向高职高专和应用型本科计算机相关专业的学生。

秉承“以应用为主体、培养实践能力”的思想,本书在强调理论知识的理解和运用的同时,更加注重结合高职高专和应用型本科教学要求,体现对学生应用技能的培养。

全书共分 8 章。内容分别为:数据结构的相关概念及算法分析,线性结构中的顺序表和链表,栈和队列的原理及应用,其他线性数据结构,包括多维数组和广义表,树和二叉树,图的原理和基本操作,查找的概念和应用,排序的概念和算法。

除第 1 章和第 8 章外,每章内容都含有“学习目标”、“本章要点”、“典型例题”、“补充习题”、“综合实训”几个部分。其中,“学习目标”指出了本章各知识点的学习要求,可以作为学生自主学习的评价指标;“本章要点”简明扼要地讲解了本章的重点知识和难点知识;本章中典型的考试题目均放在“典型例题”中,并配有详细的求解过程,主教材中难度较大的课后习题也在本部分予以解析说明;“补充习题”是对典型例题的有益补充,知识点的覆盖面更广,相对于“典型例题”难度稍低,可以用来课下练习;“综合实训”是对主教材实验内容的扩充,其难度更大,程序结构更完整,通过实训,可以使学生对本章涉及的数据结构的类型、特点、操作等有一个整体的认识和把握。本书所有章节均配有习题答案,供学生练习时参考。习题答案请登录 <http://edu.tqbooks.net> 网站进行下载。

本书具有以下特点:

(1) 融入山东省“数据结构”精品课程在教学方法、教学内容上的先进思想,内容的组织和编排以应用为主线,与主教材《数据结构》相互配合使用,是主教材的扩展和补充。

(2) 各章都配有“典型例题”,详细讲解本章中典型的考试题目,有助于学生加深对基础理论知识的理解和掌握。

(3) 各章均提供了大量的补充习题,并提供了习题答案,方便学生课下复习、巩固。

(4) 在主教材实验指导的基础上,每章均设计了综合实训。通过实训可以让学生对本章内容有一个系统的认识,并促进学生由理论学习到动手实践的转变。

(5) 为方便学生学习,本书的程序部分均采用 Turbo C 语言描述,学生可以很方便地将书中的算法进行上机测试。

作为主教材的配套学习指导教材，推荐将“典型习题”用做课堂讲授，将“补充习题”用做课下练习，将“综合实训”用做阶段性实训或大作业。其中，实训的课时数安排约为 20 课时，教师可以根据实际学时数、学生情况等自行调整进度和教学内容。

本书由济南铁道职业技术学院信息工程系李波、张亦辉主编，周朋红、李婷婷、牛文琦、李琳参加了部分章节的编写。济南铁道职业技术学院信息工程系以及兄弟院校的领导和教师在本书编写过程中给予了大力支持，在此表示感谢。

尽管我们尽了最大努力，但时间紧迫，书中难免存在疏漏之处，敬请读者批评指正。

编 者

2009 年 10 月

第 1 章 绪论	1
1.1 学习目标	1
1.2 本章要点	1
1.3 典型例题	2
第 2 章 线性表	6
2.1 学习目标	6
2.2 本章要点	6
2.3 典型例题	8
2.4 补充习题	15
2.4.1 单选题	15
2.4.2 判断题	17
2.4.3 应用题	17
2.5 综合实训	18
2.5.1 顺序表综合实训	18
2.5.2 单链表综合实训	23
第 3 章 栈和队列	29
3.1 学习目标	29
3.2 本章要点	29
3.3 典型例题	30
3.4 补充习题	34
3.4.1 单选题	34
3.4.2 判断题	36
3.5 综合实训	36
3.5.1 顺序栈	36
3.5.2 链式栈	39
3.5.3 循环队列	42
3.5.4 链式队列	46
第 4 章 其他线性数据结构	50
4.1 学习目标	50
4.2 本章要点	50
4.3 典型例题	52
4.4 补充习题	54
4.4.1 单选题	54

4.4.2 判断题.....	56
4.4.3 填空题.....	57
4.4.4 应用题.....	57
4.5 综合实训.....	58
第 5 章 树和二叉树.....	65
5.1 学习目标.....	65
5.2 本章要点.....	65
5.3 典型例题.....	67
5.4 补充习题.....	77
5.4.1 单选题.....	77
5.4.2 判断题.....	78
5.4.3 填空题.....	78
5.5 综合实训.....	79
第 6 章 图.....	84
6.1 学习目标.....	84
6.2 本章要点.....	84
6.3 典型例题.....	86
6.4 补充习题.....	95
6.4.1 单选题.....	95
6.4.2 填空题.....	97
6.4.3 应用题.....	97
6.5 综合实训.....	98
第 7 章 查找.....	107
7.1 学习目标.....	107
7.2 本章要点.....	107
7.3 典型例题.....	109
7.4 补充习题.....	118
7.4.1 单选题.....	118
7.4.2 填空题.....	119
7.4.3 应用题.....	120
7.5 综合实训.....	120
第 8 章 内部排序.....	131
8.1 学习目标.....	131
8.2 本章要点.....	131
8.3 典型例题.....	132
8.4 综合实训.....	138

第 1 章 绪 论

随着计算机技术的发展, 计算机处理的数据越来越庞大和复杂, 为了有效地组织和管理这些数据, 编制出高质量的程序以及高效率地使用计算机, 就必须深入研究这些数据的特性以及它们之间的关系。这就是“数据结构”这门学科形成和发展的背景。数据结构主要研究非数值应用问题中, 数据之间的逻辑关系和对数据的操作, 同时还研究如何将具有逻辑关系的数据按一定的存储方式存放在计算机内。分析数据之间的逻辑关系和确定数据在计算机内的存储结构是程序设计前两个必须完成的任务。

1.1 学 习 目 标

学习本章内容, 需要做到以下几点:

- ① 理解数据、数据元素、数据项的概念
- ② 掌握数据结构的四种逻辑结构的特点
- ③ 掌握数据结构的四种物理结构的特点
- ④ 掌握算法的五大特性
- ⑤ 理解算法的描述方法
- ⑥ 掌握算法的时间复杂度分析方法

1.2 本 章 要 点

本章主要讲解了两大问题: 一是数据结构的定义和相关概念; 二是算法的定义和算法分析。需要掌握的重点内容包括:

(1) 数据结构的定义: 数据结构是一门研究非数值应用问题中数据的物理结构、逻辑结构和对数据的操作即计算机算法的学科。

(2) 数据指所有能输入到计算机中并被计算机程序处理的符号的总称; 数据元素是构成数据的基本单位, 一个数据通常由若干个数据元素组成; 数据项是构成数据元素的基本单位, 一个数据元素通常由若干个数据项组成。

(3) 数据的逻辑结构是指数据之间的关联方式, 包括两大类: 线性结构和非线性结构, 并且非线性结构又分为树形结构和图形结构, 通常还将集合这种结构增加进来。因此也可以说, 数据的逻辑结构包括: 集合结构、线性结构、树形结构、图形结构四种类型。

(4) 数据的物理结构,是指数据在计算机中的存储方式(也称为存储映像),主要包括:顺序结构、链式结构、索引结构和散列结构四种。

(5) 算法是解决某一类问题的特定方法,算法可以用计算机语言、自然语言、框图、类语言等多种形式来描述。一段计算机程序可以用来表示一个算法,但是算法并不一定是用计算机程序描述的。

(6) 任何一个算法都必须满足五个特性,即正确性、确定性、有限性、输入和输出。其中,正确性是进行算法分析的前提。一个算法可以在程序中初始化一些变量,所以可以没有输入,但是一定要有输出。

(7) 当解决一个问题有多种算法时,可以通过算法分析进行优化选择。通常,算法分析包括时间复杂度分析和空间复杂度分析两种。时间复杂度并不是具体的算法运行时间,而是算法的运行时间的数量级度量,通常有 $O(1)$ 、 $O(n)$ 、 $O(n\log_2 n)$ 、 $O(n^2)$ 、 $O(n^3)$ 几种。空间复杂度是算法运行时需要的附加空间的数量级度量,常见的有 $O(1)$ 、 $O(\log_2 n)$ 、 $O(n)$ 几种。

(8) 算法的时间复杂度并不是算法具体的执行步数,而是基本操作次数相对于问题规模 n 的增长率。在计算时间复杂度时,需要忽略所有的系数、常数和低次项。

1.3 典型例题

例1 数据结构是一门研究非数值计算的程序设计问题中计算机的(①)以及它们之间的(②)和运算的学科。

- ① A. 数据元素 B. 计算方法 C. 逻辑存储 D. 数据映像
② A. 结构 B. 关系 C. 运算 D. 算法

解答:

数据结构主要研究数据元素(并非数据项)及其关系和施加在数据上的运算实现三个方面,所以本题的答案为:① A ② B。

例2 在数据结构中,从逻辑上可以把数据结构分为()。

- A. 动态结构和静态结构 B. 紧凑结构和非紧凑结构
C. 线性结构和非线性结构 D. 内部结构和外部结构

解答:

逻辑结构反映数据之间的逻辑关系,线性结构表示数据元素之间为一对一的关系,非线性结构表示数据元素之间为一对多或者多对多的关系。本题答案为 C。

例3 在数据结构中,与所使用的计算机无关的是()。

- A. 逻辑结构 B. 存储结构
C. 逻辑和存储结构 D. 物理结构

解答:

物理结构即为存储结构,它是数据在计算机中的物理映像,所以和具体的计算机有关。而逻辑结构描述的是数据元素之间的关系,与使用的计算机无关。本题答案为 A。

例4 算法分析的目的是(①),算法分析的两个主要的方面是(②)。

- ① A. 找出数据结构的合理性 B. 研究算法中输入和输出的关系
C. 分析算法的效率以求改进 D. 分析算法的易懂性和文档

- ② A. 空间复杂度和时间复杂度 B. 正确性和简明性
C. 可读性和稳定性 D. 数据复杂性和程序复杂性

解答:

算法分析的目的是为了找出高效的算法, 算法分析中最重要的是算法的效率分析, 包括时间复杂度分析和空间复杂度分析。本题答案为: ① C ② A。

例 5 计算机算法指的是 (①), 它必须具有输入、输出和 (②) 等五方面特性。

- ① A. 计算方法 B. 排序方法
C. 解决问题的有限运算序列 D. 调度方法
② A. 可行性、可移植性、可扩充性 B. 可行性、确定性、有穷性
C. 确定性、有穷性、稳定性 D. 易读性、稳定性、安全性

解答:

由算法的定义和特性可知本题的答案为: ① C ② B。

例 6 在决定选取何种存储结构时, 一般不考虑 ()。

- A. 各结点的值如何 B. 结点个数的多少
C. 对数据有哪些运算 D. 所用编程语言实现这种结构是否合理

解答:

存储结构是数据的逻辑结构在计算机存储器中的实现 (或称为映像)。存储结构有顺序、链式、索引和散列四种, 它们都可以存储结点的值, 主要区别在于各结点之间的关系不同。因此, 在决定选取何种存储结构时, 一般不考虑各结点的值如何。本题答案为 A。

例 7 以下说法中正确的是 ()。

- A. 数据元素是数据的最小单位
B. 数据项是数据的基本单位
C. 数据结构是带有结构的数据项的集合
D. 一些表面上很不同的数据可以采用相同的逻辑结构

解答:

数据元素是数据的基本单位, 而数据项是数据的最小单位, 数据结构是带结构的数据元素的集合。本题答案为 D, 比如家谱和行政区划都是树形结构。

例 8 算法可以用不同的语言描述, 如果用 C 语言或者 Pascal 语言等高级语言来描述, 则算法实际上就是程序了。这种说法正确吗?

解答:

这种说法是错误的, 因为算法不等同于程序。程序可以作为算法的一种描述手段, 但是不能直接将算法叫做程序。

例 9 数据在计算机内有链式存储和顺序存储两种方式, 在存储空间使用的灵活性上, 链式存储比顺序存储要 ()。

- A. 低 B. 高 C. 相同 D. 不好说

解答:

顺序存储方式要求所有数据存储在在一块连续的空间上, 所以要求存储器中必须有足够大的

连续空间；链式存储不要求数据的存储地址连续，因此可以使用分散的存储空间，只要空闲空间的总量足够即可。所以在灵活性上要比顺序存储高。本题答案为 B。

例 10 数据结构的基本任务是 ()。

- A. 逻辑结构和存储结构的设计 B. 数据结构的运算实现
C. 数据结构的评价与选择 D. 数据结构的设计与实现

解答：

数据结构主要研究数据的物理结构、逻辑结构以及算法的实现，所以本题的答案为 D。

例 11 下面程序段的时间复杂度是 ()。

```
for(i=0;i<n;i++)
    for(j=0;j<m;j++)
        A[i][j]=0;
```

解答：

算法中的基本操作是 $A[i][j]=0$ ，整个算法包括了两层循环，外层循环的执行次数为 n 次，内层循环的执行次数为 m 次，所以算法的时间复杂度为 $O(mn)$ 。

例 12 下面程序段的时间复杂度是 ()。

```
i=s=0;
while(s<n)
{
    i++;          //i=i+1
    s+=i;         //s=s+i
}
```

解答：

本算法中循环体的循环次数主要由变量 s 来控制，假设循环执行的次数为 $T(n)$ ，仔细分析程序可以发现 s 第 1 次增加 1，第 2 次增加 2，依此类推， $s=1+2+3+4+\cdots+T(n) < n$ 。即 $(1+T(n))*T(n)/2 < n$ （等差数列求和公式），显然 $T(n) < \sqrt{2n}$ ，其时间复杂度为 $O(\sqrt{n})$ 。

例 13 下面程序段的时间复杂度是 ()。

```
i=1;
while(i<=n)
    i=i*3;
```

解答：

本算法中循环体的循环次数主要由变量 i 来控制，假设循环执行的次数为 $T(n)$ ，仔细分析程序可以发现 i 第 1 次为 3（即 3^1 ），第 2 次为 9（即 3^2 ），第 3 次为 27（即 3^3 ）依此类推，第 $T(n)$ 次循环时 $i=3^{T(n)}$ 。显然 $T(n)=\log_3 n$ ，其时间复杂度为 $O(\log_3 n)$ 。

例 14 下列程序段的时间复杂度为 ()。

```
product=1;
for(i=n;i>0;i--)
    for(j=i;j<=n;j++)
        product*=j;
```

解答：

解题方法一：本算法中包括两层循环，外层循环变量为 i ，内层循环变量为 j ，其中 j 的循环范围会受到 i 取值的影响。仔细分析程序可以发现：当 i 为 n 时，内层循环执行 1 次；当 i

为 $n-1$ 时，内层循环执行 2 次，依此类推，循环体总共执行 $1+2+3+\cdots+n$ 次，共计执行 $(1+n)n/2$ 次。所以时间复杂度为 $O(n^2)$ 。

解题方法二：一般情况下，如果基本操作（本例中为 `product *= j`）不会对循环变量产生影响，则几层循环体，算法的时间复杂度就是 n 的几次方。本题中含有两层循环体，则时间复杂度为 $O(n^2)$ 。这种方法适合于快速估算时间复杂度，具体解题时最好按照方法一处理。

第2章 线性表

在许多领域中都有线性结构的概念，在计算机科学中，线性结构也被称为线性表。线性表是最基本最常用的一种线性结构。线性结构的特点是在数据元素的非空集中，存在唯一的一个首元素；存在唯一的一个末元素；除首元素外，每个元素均只有一个直接前驱；除末元素外，每个元素均只有一个直接后继。

2.1 学习目标

学习本章内容，需要做到以下几点：

- ① 掌握线性表的两种存储方式，即链式存储方式和顺序存储方式
- ② 掌握顺序表的插入、查找、删除操作程序及效率分析
- ③ 掌握单链表的插入、查找、删除操作程序及效率分析
- ④ 掌握双向链表、循环链表的概念
- ⑤ 掌握双向链表的插入、删除操作程序
- ⑥ 理解线性表两种存储方式的异同点及每种存储方式的适用范围

2.2 本章要点

本章介绍了线性表的逻辑结构及它的两种存储结构：顺序表和链表。对于链表，又介绍了单链表、双向链表和循环链表三种常见的链表。

(1) 通俗地讲，线性表是由同一类型的数据元素构成的一种线性结构，数据元素之间是一种线性关系，数据元素“一个接一个地排列”，通常记为： $(a_1, a_2, \dots, a_{i-1}, a_i, a_{i+1}, \dots, a_n)$ 。其中， n 为表长， $n=0$ 时称为空表。表中相邻元素之间存在着顺序关系。将 a_{i-1} 称为 a_i 的直接前驱， a_{i+1} 称为 a_i 的直接后继。对于 a_i ，当 $i=2, 3, \dots, n$ 时，有且仅有一个直接前驱 a_{i-1} ；当 $i=1, 2, \dots, n-1$ 时，有且仅有一个直接后继 a_{i+1} ；而 a_1 是表中第一个元素，它没有前驱； a_n 是最后一个元素，它没有后继。

(2) 线性表通常具有初始化(INIT_LIST)、求表长(LENGTH_LIST)、取表元(GET_LIST)、按值查找(LOCATE_LIST)、插入元素(INSERT_LIST)、删除元素(DELETE_LIST)等操作。

(3) 线性表的顺序存储是指在内存中用地址连续的一块存储空间顺序存放线性表的各元素，用这种存储形式存储的线性表称为顺序表。顺序表用物理上的相邻关系来记录数据元素之

间的逻辑相邻关系。设 a_1 的存储地址为 $\text{Loc}(a_1)$ ，每个数据元素占 d 个存储地址，则第 i 个数据元素的地址为：

$$\text{Loc}(a_i) = \text{Loc}(a_1) + (i-1) * d \quad 1 \leq i \leq n$$

这就是说，只要知道顺序表首地址和每个数据元素所占地址单元的个数，就可求出第 i 个数据元素的地址。因此，顺序表具有按数据元素的序号随机存取的特点。

(4) 顺序表的插入是指在表的第 i 个位置上插入一个值为 x 的新元素，插入后使原表长 n 变为 $n+1$ 。完成这一运算需要：① 将 $a_i \sim a_n$ 顺序向下移动，为新元素让出位置；② 将 x 置入空出的第 i 个位置；③ 修改 last 指针（相当于修改表长），使之增加 1。算法要注意以下问题：① 由于顺序表的空间大小是预先分配的，所以在向顺序表做插入时要先检查表空间是否已满，在表满的情况下不能再做插入，否则产生溢出错误。② 要检验插入位置的有效性，这里 i 的有效范围是： $1 \leq i \leq n+1$ ，其中 n 为原表长。③ 注意数据的移动方向。在顺序表上做插入操作平均需移动表中一半的数据元素，时间复杂度为 $O(n)$ 。

(5) 顺序表的删除操作需要注意：① 删除第 i 个元素时要检查删除位置的有效性， i 的取值为 $1 \leq i \leq n$ ，否则第 i 个元素不存在。② 当表空时不能做删除。③ 删除 a_i 之后，该数据已不存在，如果需要，先取出 a_i ，再做删除。算法的时间复杂度为 $O(n)$ 。

(6) 按值查找是指在线性表中查找与给定值 x 相等的数据元素。从第一个元素 a_1 起依次和 x 比较，直到找到一个与 x 相等的数据元素，则返回它在顺序表中的存储下标或序号（二者数值相差 1）；若查遍整个表都没有找到与 x 相等的元素，则返回 0。本算法的主要运算是比较。显然，比较的次数与 x 在表中的位置有关，也与表长有关，平均比较次数为 $(n+1)/2$ ，时间复杂度为 $O(n)$ 。

(7) 顺序存储的优点：

① 方法简单，各种高级语言中都有数组，容易实现。

② 不用为表示结点间的逻辑关系而增加额外的存储开销，而是通过元素的物理位置相邻来表示逻辑位置的相邻关系。

③ 顺序表具有按元素序号随机访问的特点。

(8) 顺序结构的缺点：

① 在顺序表中做插入、删除操作时，平均移动大约表中一半的元素，因此对 n 较大的顺序表效率低，插入、删除、按值查找的时间复杂度均为 $O(n)$ 。

② 需要预先分配足够大的连续存储空间，如果空间预分配过大，可能会导致顺序表后部大量闲置；预分配过小，又容易造成溢出。

(9) 线性表链式存储结构不要求逻辑上相邻的两个数据元素物理上也相邻，因此它不需要用地址连续的存储单元来实现，它通过“链（指针）”来表示数据元素之间的逻辑关系，对线性表的插入、删除不需要移动大量数据元素。在链式结构中，既需要存储元素的值，又需要额外的空间存储元素之间的关系，因此，存储密度一般要小于线性表的顺序存储结构。

(10) 单链表中每个结点中只有一个指向后继的指针，因此单链表中元素的后继可以通过指针直接找到，但是查找一个元素的前驱需要从表头开始遍历整个链表。

(11) 单链表的建立分为头插入法和尾插入法两种，其中头插入法建立的链表，其结点顺序和插入元素的顺序正好相反；尾插入法建立的链表，其结点顺序与插入元素的顺序相同。

(12) 由于链表中逻辑相邻的两个元素物理位置上不一定相邻，所以不能通过某种公式简单地算出某个元素所在的位置。无论是按照序号查找某一个元素，还是按值查找某一个元素，都需要从表头开始遍历整个链表。

(13) 循环链表: 在单链表的基础之上, 将链表头指针置入最后一个结点的指针域, 使得链表头尾结点相连, 就构成了单循环链表。循环链表可以很方便地从表尾结点直接找到表头结点。为了提高效率, 一般在程序中使用尾指针来表示循环链表。

(14) 双向链表: 在单链表的基础之上, 每个结点再加一个指向前驱的指针域, 用这种结点组成的链表称为双向链表。双向链表的优势在于求一个结点的前驱结点和后继结点非常容易, 时间复杂度为 $O(1)$ 。双向链表具有对称性。

(15) 线性表的链式存储结构插入、删除元素效率高, 空间不需要预先分配, 插入元素不需要考虑空间溢出的情况。但是链式结构查找元素的效率低, 因此链式结构一般用于插入、删除元素频繁的程序中。顺序结构的优缺点和链式结构相反, 查找效率高, 但是插入、删除效率低, 一般用于查询较多, 但元素变化较少的程序中。实际应用中选用哪种存储结构要根据具体问题的要求来选择。

2.3 典型例题

例 1 链表不具备的特点是 ()。

- A. 可以随机访问任一结点
- B. 插入删除不需要移动元素
- C. 不必事先估计存储空间
- D. 所需空间与其长度成正比

解答:

顺序表可以随机访问其任一结点, 而链表必须从第一个数据结点出发逐个查找每个结点, 所以本题答案为 A。

例 2 不带头结点的单链表 head 为空的判定条件为 (), 带头结点的单链表 head 为空的判定条件为 ()。

- A. head==NULL
- B. head->next==NULL
- C. head->next==head
- D. head!=NULL

解答:

在不带头结点的单链表中, head 指向第 1 个元素结点, head==NULL 表示该单链表为空。在带头结点的单链表 head 中, head 指向头结点, 由头结点的 next 域指向第 1 个元素结点, head->next==NULL 表示该单链表为空。本题的答案为 A、B。

例 3 在线性表的下列存储结构中, 读取元素花费时间最少的是 ()。

- A. 单链表
- B. 双链表
- C. 循环链表
- D. 顺序表

解答:

顺序表可以根据下标直接读取对应的元素, 所以读取元素的时间花费最少, 答案为 D。

例 4 对于只在表的首、尾两端进行插入操作的线性表, 宜采用的存储结构为 ()。

- A. 顺序表
- B. 用头指针表示的单循环链表
- C. 用尾指针表示的单循环链表
- D. 单链表

解答:

顺序表在表尾插入元素只需要 $O(1)$ 的时间复杂度, 而在表头插入元素时, 需要将所有的元素向后移动, 时间复杂度为 $O(n)$ 。