



面向“十二五”高等学校精品规划教材◎电气信息类
高等教育课程改革项目研究成果

C语言程序设计 项目教程

主编 李震平 韩晓鸿

- **新**: 新思路、新领域、新技术、新变革
- **活**: 模块化、立体化、可扩展
- **精**: 精品、精心、精致

 **北京理工大学出版社**
BEIJING INSTITUTE OF TECHNOLOGY PRESS



清华大学出版社
TSINGHUA UNIVERSITY PRESS

C语言程序设计 项目教程

第2版

主 编 李 波 副 编 李 强 李 强 李 强

参 编 李 强 李 强 李 强

编 者 李 强 李 强

审 校 李 强 李 强

清华大学出版社

北京清华大学学研大厦A座

面向“十二五”高等学校精品规划教材·电气信息类
高等教育课程改革项目研究成果

C 语言程序设计项目教程

主 编	李震平	韩晓鸿			
副主编	李晓霞	魏红君	薛红梅		
编 委	程慧芳	赵 辉	李 燕	杨 丽	
	马丽艳	宁振刚	李 鹰		

 北京理工大学出版社

BEIJING INSTITUTE OF TECHNOLOGY PRESS

内 容 简 介

本书“以 CDIO 为理念,以项目驱动为主线”,通过一个典型项目“河北工程大学教师工资管理系统”讲解 C 语言所有知识点,展现了全新的教材结构。

本书共 10 章,前 9 章包含了软件开发基础、C 语言程序概述、程序设计三大结构、结构体、数组、函数、指针、文件操作和编译预处理,第 10 章对 C 语言的所有知识点进行了分类汇总,以体现知识点之间的联系,利于整体学习和把握 C 语言。本书除通过“河北工程大学教师工资管理系统”项目来讲解 C 语言知识点外,还提供了内容丰富的、趣味性较强的实例。同时配套出版的《C 语言程序设计实验实训教程》也具有相同的风格。

本书既可以作为非计算机专业本科学生的计算机 C 语言教材,也可以作为高等院校计算机专业本科和专科学生的基础教材,还可作为自学者和教师的参考资料。

版权专有 侵权必究

图书在版编目 (CIP) 数据

C 语言程序设计项目教程 / 李震平, 韩晓鸿主编. —北京: 北京理工大学出版社, 2011. 1

ISBN 978 - 7 - 5640 - 4130 - 4

I. ①C… II. ①李… ②韩… III. ①C 语言 - 程序设计 - 高等学校 - 教材 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2010) 第 260228 号

出版发行 / 北京理工大学出版社

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010)68914775 (办公室) 68944990 (批销中心) 68911084 (读者服务部)

网 址 / <http://www.bitpress.com.cn>

经 销 / 全国各地新华书店

印 刷 / 北京慧美印刷有限公司

开 本 / 787 毫米 × 1092 毫米 1/16

印 张 / 20.5

字 数 / 478 千字

版 次 / 2011 年 1 月第 1 版 2011 年 1 月第 1 次印刷

印 数 / 1 ~ 5000 册

责任校对 / 陈玉梅

定 价 / 35.00 元

责任印制 / 边心超

图书出现印装质量问题, 本社负责调换

前言

Preface

自 20 世纪 80 年代中叶 C 语言引入中国以来,已被许多高校列为程序设计课程的首选语言。尤其是随着我国经济的快速发展和科学技术的不断进步,工业化大力发展,自动化程度越来越高,社会对科技人员的计算机技术、技能要求也越来越高,C 语言既成为重要的计算机技术基础语言,更是工业自动化控制的重要开发语言,因此,学习和掌握 C 语言的重要性更加突出。近年来,“亚太大学生机器人大赛”“飞思卡尔杯全国大学生智能汽车竞赛”等面向在校大学生的各类电子竞技,从程序控制角度看,对 C 语言的驾驭能力是决定竞赛胜负的关键。

C 语言程序设计是一门实践性很强的课程,学习者必须在实践中逐步理解和掌握程序设计的思想和方法,而传统的 C 语言程序设计教材,往往采用“概念—解释—举例”的三部曲,这种方式曾经取得了一定成效,但也存在着不利于培养学生程序设计的实践能力和团队协作能力等不足,导致大学毕业生不能满足当今社会对工程人才的实际需要。

基于对培养工程人才的迫切需要,从 2000 年起,麻省理工学院和瑞典皇家工学院等 4 所大学经过 4 年的探索研究,创立了 CDIO 工程教育理念,并成立了以 CDIO 命名的国际合作组织。CDIO 代表构思 (Conceive)、设计 (Design)、实现 (Implement) 和运作 (Operate),它以产品研发到产品运行的生命周期为载体,让学生以主动的、实践的、课程之间有机联系的方式学习工程。目前国内已经成立了“中国 CDIO 工程教育模式研究与实践”课题组,主要研究国际工程教育改革情况和 CDIO 工程教育模式的理念及做法。

本教材依据 CDIO 工程教育模式的理念,结合作者多年 C 语言的教学经验和软件开发实践,对 C 语言知识点的编排进行了精心的策划和组织,直接从实践应用入手,通过一个完整的“河北工程大学教师工资管理系统”项目的实现,讲解了绝大部分的 C 语言知识点: C 语言程序结构、数据类型、变量作用域、结构化程序三大结构、结构体、数组、函数、指针、文件操作和预处理等。因此,本书有着鲜明的工程项目特色,更有利于学生在实践中领会所学知识,解决实际问题。

(1) 实用性强。通过对“工资管理项目”的学习与开发,将分散的知识点进行有机联系,真正达到学以致用目的,学习者可以通过借鉴课程中项目的经验,最终用于开发其他实用的小软件。附录一给出了开发一个软件项目的整体参考流程:文档资料、结构框图和流程图,及采用结构体组织工资记录的“工资管理系统”源程序,方便学习者学习和借鉴;附录二给出了采用双向链表组织工资记录的“工资管理系统”的源程序;附录四给出了 C 语言常用标准库函数,方便查阅;第十章给出了 C 语言全面的知识点汇总,更加适合初学者和有一定基

础的读者。

(2) 趣味性强。所有案例都精心选择与设计,除一些经典算法外,在趣味性上做了大量的工作,一改过去学习计算机语言的生硬与枯燥,能有效提高读者的学习兴趣,激发读者的求知欲望。

(3) 符合认识规律。根据知识点的特点,或采用“实践—理论—实践”,或采用“理论—实践—理论—实践”的编排模式,以利于读者学习和掌握。

(4) 强调“阅读、编写、调试”三大能力的培养。这是确定能否真正掌握 C 语言的根基,本书许多案例的分析过程,均渗透了这 3 种能力的培养。

本书所有案例的调试环境均为 MicroSoft Visual C++ 6.0,以适应现实的需要和全国计算机 C 语言二级考试环境,并且所有案例均调试通过。

《C 语言程序设计项目教程》由李震平主编,第一、二章由杨丽编写,第三章由李晓霞编写,第四章由赵辉编写,第五章由魏红君编写,第六章由韩晓鸿编写,第七章由程慧芳编写,第八章、第十章、附录二由李震平编写,第九章由薛红梅编写,附录一由李燕编写,附录三由马丽艳编写,附录四由宁振刚编写。全书由李震平负责主审,韩晓红、魏红君、李晓霞和薛红梅负责统稿。同时,与本书配套的《C 语言程序设计实验实训教程》亦同时出版。

在本书的撰写过程中,得到了河北工程大学张永强教授和贺洪江教授、天津大学计算机科学与技术学院李鹰的大力支持和指导,在此表示衷心感谢。

由于作者水平有限,书中难免有疏漏或处理不当之处,恳请读者批评指正。

李震平

目录

Contents

第一章 软件开发基础——工资管理系统之项目管理	1
任务一 学习目标	1
任务二 了解软件工程	1
任务三 了解数据结构	5
任务四 了解算法设计	7
任务五 本章要点汇总	11
任务六 实力检验	12
第二章 C 语言程序设计概述	14
任务一 了解程序与程序设计	14
任务二 了解 C 语言	19
任务三 了解 Visual C++ 6.0 集成开发环境	23
任务四 创建、编辑源程序	24
任务五 编译、连接、运行和调试	26
任务六 程序的保存、打开、关闭与退出	31
任务七 本章要点汇总	32
任务八 实力检验	33
第三章 结构化程序的三大结构——工资管理系统之主菜单设计	36
任务一 学习目标	36
任务二 掌握 C 语言的词法	37
任务三 理解 C 语言的数据类型	38
任务四 理解变量与常量	39
任务五 掌握运算符、表达式的使用	44
任务六 掌握输入/输出数据的方法	52
任务七 顺序结构程序设计	64
任务八 选择结构程序设计	65
任务九 循环结构程序设计	74

任务十 知识扩展	83
任务十一 精选案例	86
任务十二 应用实例——工资管理系统的主菜单设计	93
任务十三 本章要点汇总	94
任务十四 实力检验	95
第四章 结构体数据类型——工资管理系统之数据结构设计	101
任务一 学习目标	101
任务二 声明结构体类型	101
任务三 职工工资数据结构设计与结构体变量定义	103
任务四 输入职工工资	107
任务五 计算职工工资	111
任务六 输出职工工资	112
任务七 知识扩展	112
任务八 精选案例	116
任务九 本章要点汇总	120
任务十 实力检验	120
第五章 数组——工资管理系统之数据管理	122
任务一 学习目标	122
任务二 认识数组	123
任务三 查询工资	128
任务四 排序工资	129
任务五 添加/删除工资	133
任务六 数组实践——结构体数组	136
任务七 知识扩展之一：二维数组与多维数组	138
任务八 知识扩展之二：字符数组与字符串	143
任务九 精选案例	153
任务十 未来的任务	162
任务十一 本章要点汇总	163
任务十二 实力检验	163
第六章 函数——工资管理系统之模块功能的实现	165
任务一 学习目标	165
任务二 了解函数的作用	165
任务三 库函数的调用	167
任务四 函数的定义	168
任务五 函数的类型和返回值	170
任务六 函数调用及参数传递	173

任务七	函数的嵌套调用、递归调用	177
任务八	函数的作用域	180
任务九	知识扩展——变量的作用域与存储方式	181
任务十	精选案例	186
任务十一	本章要点汇总	198
任务十二	实力检验	199
第七章 指针——工资管理系统之灵活数据处理		201
任务一	学习目标	201
任务二	了解指针的概念	201
任务三	了解指针变量的定义与初始化	203
任务四	掌握指针的常规应用（指针运算）	205
任务五	了解指针与数组	207
任务六	了解指向结构体变量的指针	217
任务七	了解指针与函数	218
任务八	知识扩展	221
任务九	精选案例	232
任务十	本章要点汇总	237
任务十一	实力检验	238
第八章 文件操作——工资管理系统之数据永久记录		241
任务一	学习目标	241
任务二	了解文件	241
任务三	了解文件的存储形式	241
任务四	了解对文件的操作	242
任务五	打开文件——fopen()	242
任务六	将职工工资数据写入文件——fwrite()函数的应用	244
任务七	从文件中读取职工工资数据——fread()函数的应用	245
任务八	关闭文件——fclose()函数的应用	246
任务九	更多的读写文件函数	246
任务十	了解文件数据的定位	250
任务十一	知识扩展	253
任务十二	精选案例	253
任务十三	本章要点汇总	260
任务十四	实力检验	261
第九章 编译预处理——工资管理系统之预处理应用		265
任务一	学习目标	265
任务二	文件包含	265

任务三	宏定义	266
任务四	条件编译	271
任务五	知识扩展	273
任务六	精选案例	274
任务七	本章要点汇总	276
任务八	实力检验	276
第十章 C 语言程序设计之技术与知识点分类汇总		280
一、	C 语言程序设计之结构	280
二、	C 语言程序设计之语句	281
三、	C 语言程序设计之数据类型	281
四、	C 语言运算符之优先级与结合性	282
五、	C 语言程序设计之函数	282
六、	C 语言程序设计之指针	283
七、	C 语言程序设计之文件操作	283
八、	C 语言程序设计之编译与预处理	284
附录一	基于软件工程的“工资管理系统”软件开发附源程序	286
附录二	“河北工程大学工资管理系统”链表结构程序源代码	304
附录三	ASCII 字符编码一览表	315
附录四	Turbo C 部分常用库函数	316
参考文献		320

第一章 软件开发基础——工资 管理系统之项目管理

任务一 学习目标

C 语言 (Combined Language) 是最通用和流行的计算机程序设计语言之一, 从诞生到现在, 历经了近 40 年的辉煌历程, 仍然以其紧凑的代码、高效的运行效率、强大的功能, 以及可以直接操作计算机硬件等特点, 深受广大编程爱好者和软件开发人员的欢迎。

C 语言是 Dennis M. Ritchie 于 1972 年在 BCPL (Basic Combined Programming Language) 的基础上研制而成的, 并用 C 语言写成了第一个在 DEC PDP-11 计算机上实现的 UNIX 操作系统。1990 年美国国家标准学会 (ANSI) 颁布了 ANSI C 标准。一年后, 该标准被定为国际标准, 一个新的“无歧义、与具体平台无关的 C 语言定义”产生。随后出现了遵循该标准的各种 C 和 C++ 集成开发环境。在学习 C 语言的过程中, 读者将会体会到 C 语言的魅力以及编程带来的无穷乐趣。

C 语言的应用领域十分广泛, 如人工智能、单片机控制、计算机系统软件等的底层开发, 工业控制、智能仪表、嵌入式系统、硬件驱动程序开发等。

近年来, 在每年的“亚太大学生机器人大赛”“飞思卡尔杯全国大学生智能汽车竞赛”及某些国际或国内大型企业举办或赞助举办的面向在校大学生的各种电子竞技中, 许多参赛院校都是使用 C 语言编制其控制程序的。有人风趣地形容: “学好 C 语言, 走遍天下都不怕。”另外, C 语言的掌握可以为学习其他语言 (如 C++、Java 等面向对象的程序设计语言) 奠定基础。

在学习 C 语言之前, 需要先了解一些程序设计与开发的相关知识, 这些是程序设计的基础。本章的后续 3 项任务中分别介绍了软件工程、数据结构和算法设计的一些基本概念和思想。如果读者需要深入理解和掌握这些理论知识, 可以查阅相关书籍。

任务二 了解软件工程

1. 软件的概念及特点

(1) 软件的定义

软件是程序、数据及其相关文档的完整集合。也可以说, 软件由两部分组成: 一是程序和数据; 二是文档。文档是与软件的程序设计、开发、维护和使用有关的图文资料。

(2) 软件的特点

软件具有以下特点。

① 软件是一种逻辑产品, 它与物质产品有很大的区别, 它以程序和文档的形式出现, 保



存在计算机的存储器中（磁盘或光盘），通过计算机的运行才能体现它的功能和作用。

② 软件产品的生产主要是研制。其成本主要体现在软件的开发和研制上，软件开发研制完成后，通过复制可以生产大量相同的软件产品。

③ 软件产品不会用坏，不存在磨损、消耗的问题，但是软件在使用过程中，往往需要根据用户需求的变化或软硬件环境的变化对软件进行修改，这种修改称为软件维护。

④ 软件产品的生产主要是脑力劳动。

⑤ 软件的费用是不断增加的，软件成本相当昂贵。

2. 软件工程相关知识

1) 软件工程

20 世纪 60 年代中后期，软件数量越来越多，规模越来越大，而软件的生产基本上是各自为战，缺乏科学规范的系统规划与测试、评估标准，致使大批耗费巨资研制出来的软件系统，因含有错误而无法使用，造成了巨大损失。由于问题软件不断出现，给人的感觉是软件越来越不可靠。这一切极大地震撼了计算机界，历史上称之为“软件危机”。一些软件开发人员认识到：大型程序的编制不同于小程序的编写，它应该是一项新的技术，应该像管理工程一样管理软件研制的全过程。于是，1968 年北大西洋公约组织在联邦德国的一次学术会议上，首次提出了“软件工程”的概念，从而形成一门新兴的学科——软件工程。

软件工程是一门用工程化方法，指导计算机软件开发和维护的学科，它采用工程的概念、原理、技术和方法来开发、维护以及管理软件。软件工程是目前计算机学科中一个年轻并且充满活力的研究领域。

2) 软件工程的 3 要素

软件工程包括 3 个重要要素，即方法、工具和过程。

软件工程的方法是完成软件工程项目的手段。它包括许多方面，例如，项目计划与估算、软件需求分析、系统总体结构设计、数据结构设计、算法设计、编写代码、软件测试及维护等。

软件工程的工具是人类在开发软件的活动中智力和体力的扩展和延伸，它提供了自动或半自动的软件支撑环境。

软件工程的过程贯穿于软件开发的各个环节，支持软件开发各个环节的控制和管理。

3) 软件生命周期

任何有生命的动物、植物和人，都有一个生存周期，没有生命的事物或实体，例如 PC（个人计算机）、家具、房子、汽车等，它们也有一个生存周期，这个生存周期就是使用寿命。软件和其他事物相似，有它的产生、发展和消亡的过程，软件生命周期是指从开发软件概念的提出起，直到该软件的使用、失去使用价值而被废弃的整个过程。

在软件生命周期中，每个阶段都有确定的任务，并产生一定规格的文档传递给下一阶段，下一阶段在前一阶段所提供文档的基础上继续开展工作。软件生命周期包括以下 6 个阶段。

(1) 可行性研究与计划制订

在此阶段，软件开发方与需求方需要共同讨论以确定软件的开发目标，开发方还要进行软件项目的可行性研究。可行性研究的目的是研究这个软件项目是否值得去开发，以及其中的问题能否解决。可行性研究的内容包括 4 个方面。

① 经济可行性：进行成本/效益分析，从经济角度判断系统开发是否“合算”。

② 技术可行性：进行技术风险评估，从开发者的技术实力、工作经验、问题的复杂程度等方面出发，判断在时间、费用等限制条件下软件项目能否实现。

③ 社会可行性：研究在系统开发过程中可能涉及的人力资源、各种合同、知识产权纠纷、责任以及各种与法律相抵触的问题。

④ 方案的选择：提出并评价系统的各种开发方案，从中选出一种最适宜的方案。

(2) 需求分析

软件需求是指用户对目标系统在功能、行为、性能等方面的期望。需求分析是发现、求精、建模和生产规格说明的过程，软件开发人员需要在对应用问题及环境理解和分析的基础上，为问题涉及的信息、功能及行为建立模型。需求分析阶段研究的对象是用户对软件项目的要求。在此阶段，一方面，必须全面理解用户的各项要求，但并不一定全盘接受所有的要求；另一方面，要准确地表达出所接受的用户要求，只有经过确切描述的软件需求才能成为软件设计的基础。

需求分析阶段的工作可概括为 4 个方面：需求获取、需求分析、编写需求规格说明书和需求评审。

(3) 软件设计

软件设计是将需求分析阶段得出的逻辑模型，设计成具体的软件方案。软件设计主要包括设计软件的总体结构、数据结构和软件具体模块的算法，解决“系统怎么做”的问题。软件设计阶段结束之前，也要进行有关评审，评审通过后才能进入编码阶段。软件设计阶段也可以根据具体软件的规模、类型等决定是否细分成概要设计和详细设计两个阶段。

本书通过软件项目实例“河北工程大学职工工资管理系统”，实现对学校职工的有关个人信息和工资信息进行存储和管理。

① 软件的总体结构设计：工资管理系统的软件结构如图 1-1 所示。

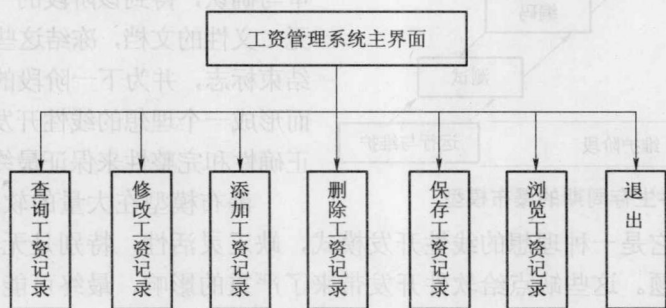


图 1-1 工资管理系统的总体结构

② 数据结构设计：采用链表处理工资信息，工资信息包括职工工号、职工姓名、岗位工资、薪级工资、职务津贴、绩效工资、应发工资、个人所得税和实发工资。

③ 具体模块的实现算法设计：查询职工工资记录、修改职工工资记录、添加职工工资记录、删除职工工资记录。

每个模块的算法的设计为下一个“软件实现”阶段做准备，例如，查询工资记录模块，该模块的算法设计思路是以“工号”为查询对象，在全体职工范围内进行查找，通过比较“工号”找到相应的工资记录并输出；如果找不到，则显示“职工工号有误或不存在的”提示信息



息，并返回到主菜单。具体算法可以采用流程图或自然语言进行描述。

(4) 软件实现

软件实现即编写代码。此阶段程序员按照软件设计说明书的要求，为每个模块编写程序代码。编写的程序应该结构合理、清晰易读，并且与设计说明书要求一致。

(5) 软件测试

软件测试是保证软件质量的重要手段，目的是尽可能多地发现程序中的错误。软件测试分为模块测试、组装测试和确认测试。在软件测试阶段要做好软件测试记录，写出软件测试与评价报告。

(6) 运行与维护

运行与维护是对软件运行提供支持，软件系统经过测试后仍可能存在潜在的错误或缺陷，当软件投入运行后，这些潜在的问题可能会显现出来。用户的需求和系统的运行环境也可能发生改变，需要对软件进行修改，成功维护后的软件成为新的软件版本，进行软件维护也要详细做好维护修改记录。

4) 软件开发模型

软件开发模型是软件开发全部过程、活动和任务的结构框架，它能直观表达软件开发全过程，明确规定要完成的主要活动、任务和开发策略。常用的软件开发模型有：瀑布模型、快速原型模型、螺旋模型、基于构件的开发模型以及基于知识的模型。

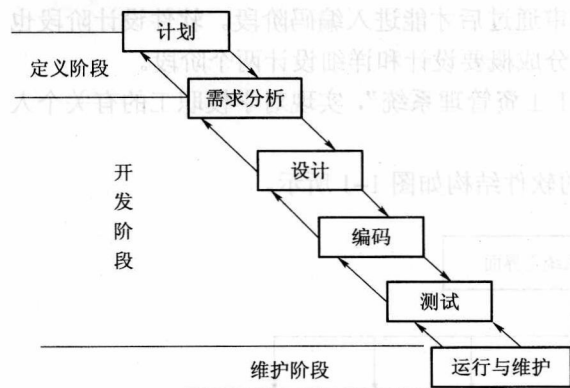


图 1-2 软件生存周期的瀑布模型

① 瀑布模型严格按照生存周期各个阶段的目标、任务、文档和要求来进行软件开发，如图 1-2 所示。强调了每一阶段的严格性，尤其是开发前期的良好需求说明。开发人员试图在每一阶段活动结束后，通过严格的阶段性复审与确认，得到该阶段的一致、完整、准确和无二义性的文档，冻结这些文档作为该阶段的结束标志，并为下一阶段的活动奠定基础，从而形成一个理想的线性开发序列，以每一步的正确性和完整性来保证最终系统的质量。

瀑布模型在大量的软件开发实践中逐渐暴露出它的缺点。它是一种理想的线性开发模式，缺乏灵活性，特别是无法解决软件需求不明确或不准确的问题。这些缺点给软件开发带来了严重的影响，最终可能导致开发的软件不是用户真正需要的软件，并且这一点在开发过程完成后才能发现。

② 原型模型即快速模型。原型模型的第一步是快速建立一个能反映用户主要需求的原型系统，让用户在计算机上使用它，通过实践来了解目标系统的概貌。通常，用户试用原型系统之后会提出许多修改意见，开发人员按照用户的意见快速地修改原型系统，然后再次请用户试用。一旦用户认为这个原型系统确实能满足他们的需求，开发人员便可根据这份规格说明文档设计软件，此软件可以满足用户的真实需求。

其他模型的开发过程可查阅软件工程方面的书籍。

3. 软件工程目标

软件工程是一门工程性学科，目的是成功地建造一个大型软件系统，同时达到这样几个

目标：较低的成本、较好的性能、完善的功能、较好的移植性、较低的维护费用、及时的交付以及较高的可靠性等。

任务三 了解数据结构

1. 数据、数据元素和数据项的概念

数据是对客观事物的符号表示，在计算机科学中是指能输入到计算机中并被计算机存储和加工的符号的总称。

数据元素是数据的基本单位，可由若干个数据项组成，在程序中通常作为一个整体进行考虑和处理。

数据项是数据不可分割的最小单位。

数据对象是性质相同的数据元素的集合，是数据的一个子集，如工资管理项目中的职工即数据对象。

2. 数据结构的概念及分类

数据结构是指相互之间存在一种或多种特定关系的数据元素的集合，是计算机存储和组织数据的方式。数据结构主要研究和讨论数据的逻辑结构、存储结构和数据的运算。

数据结构分为逻辑结构和存储结构两大类，具体分类如图 1-3 所示。

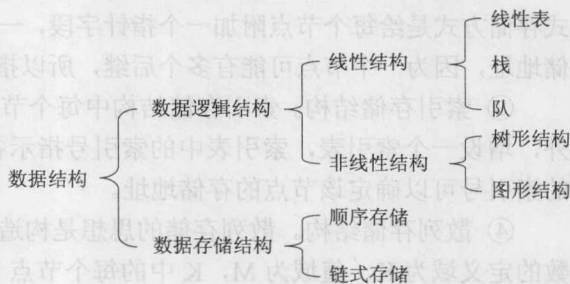


图 1-3 数据结构分类

(1) 数据的逻辑结构

数据的逻辑结构是对数据元素之间逻辑关系的描述，是数据的组织形式。数据元素之间有 4 种基本逻辑结构：集合、线性结构、树形结构和图状结构。

- ① 集合。集合中任意两个数据元素间没有逻辑关系，组织形式松散，如图 1-4 所示。
- ② 线性结构。数据元素间构成一种顺序的线性关系，如图 1-5 所示。



图 1-4 集合



图 1-5 线性结构

③ 树形结构。树形结构具有分支、层次特性，数据元素间形成一种一对多的关系，如图 1-6 所示。

④ 图状结构。也叫网状结构，是最复杂的一种数据结构，结构中数据元素之间存在多对多的关系，如图 1-7 所示。

(2) 数据的存储结构

数据的逻辑结构是独立于计算机的，与数据在计算机中的存储无关。但是，要使用计算机对数据进行处理，就必须将数据存储于计算机中，数据结构在计算机中的表示称为数据的存储结构，也称为物理结构。那么，数据在存储器中是如何存放的呢？试想，如果《英汉词

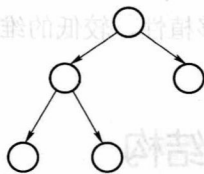


图 1-6 树形结构

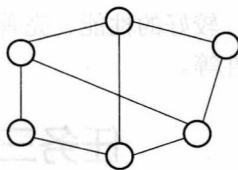


图 1-7 图状结构

典》中的单词没有任何规律地随意罗列，那么，用这样的词典查询将犹如大海捞针。同样，如果数据在计算机中无规律地存储，则处理时也会非常困难。

数据的存储结构有 4 种基本的存储映像：顺序存储结构、链式存储结构、索引存储结构和散列存储结构。

① 顺序存储结构。顺序存储通常用于存储具有线性结构的数据。将逻辑上相邻的节点连续地存放在存储器中，使得逻辑上相邻的节点在物理位置上也一定是相邻的。

② 链式存储结构。节点在存储器中随意存放，节点之间的物理关系与逻辑关系无关。链式存储方式是给每个节点附加一个指针字段，一个节点的指针所指的是该节点的后继节点的存储地址，因为一个节点可能有多个后继，所以指针字段可以是一个指针，也可以是多个指针。

③ 索引存储结构。索引存储结构中每个节点仅含一个数据元素，所有节点连续存放。此外，增设一个索引表，索引表中的索引号指示各存储节点的存储位置，换句话说，根据节点的索引号可以确定该节点的存储地址。

④ 散列存储结构。散列存储的思想是构造一个从集合 K 到存储区域 M 的函数 h ，该函数的定义域为 K ，值域为 M ， K 中的每个节点 k_i 在计算机中的存储地址由 $h(k_i)$ 确定。

总之，在程序设计中，数据结构的选择是一个非常重要的环节。许多大型系统的开发经验表明，系统实现的难易程度和系统的质量在很大程度上依赖于是否选择了最优的数据结构。一旦确定了数据结构，算法就容易得到。当然也可以根据特定算法来选择数据结构与之适应。不论哪种情况，选择合适的数据结构是非常重要的，它可以带来更高的运行效率和存储效率。

(3) 线性结构与非线性结构

根据数据结构中各元素之间前驱后继关系的不同，一般将数据结构分为线性结构和非线性结构两大类。

如果在一个数据结构中，数据元素之间为一对一的线性关系，第一个元素没有直接前驱，最后一个元素没有直接后继，其余元素都有且只有唯一的一个直接前驱和直接后继，则称这种数据结构为线性结构。如果一个数据结构不是线性结构，则称之为非线性结构。

(4) 常用的几种数据结构

① 线性表。线性表中的数据元素是一一对应的关系，即除了第一个和最后一个数据元素之外，其他数据元素都是首尾相接的。它是最基本、最简单的一种数据结构，便于实现和操作，因此，线性表在实际应用中被广泛采用。

线性表的存储结构有两种：顺序存储结构和链式存储结构。

如果线性表采用顺序存储的方式存储则称之为顺序表。顺序表是将表中的节点依次存放在计算机内存中一组地址连续的存储单元中。由于一维数组采用顺序存储结构，因此，可以用数组来实现线性表的顺序存储结构。线性链表是具有链式存储结构的线性表。

线性表可以进行插入、删除、排序、查找、计数等操作。

② 栈。栈是只能在一端进行插入和删除操作的特殊线性表。它按照“后进先出”的原则存储数据，先进入的数据被压入栈底，最后进入的数据在栈顶，需要读数据的时候从栈顶开始弹出数据（最后进入的数据第一个被读出来）。

③ 队列。队列是只允许在表的前端（front）进行删除操作、在表的后端（rear）进行插入操作的线性表。进行插入操作的端称为队尾，进行删除操作的端称为队首。队列是“先进先出”的线性表。队列中没有元素时，称为空队列。

④ 链表。链表是一种物理存储单元上非连续、非顺序的存储结构，数据元素的逻辑顺序是通过链表中的指针链接次序实现的。链表由一系列节点（链表中每一个元素称为节点）组成，节点可以在运行时动态生成。每个节点包括两部分：一部分是存储数据元素的数据域；另一部分是存储下一个节点地址的指针域。

⑤ 树。树是包含不少于一个节点的有穷集合。例如，有 n ($n > 0$) 个节点的集合 K ，且在 K 中定义了一个关系 N ， N 满足的条件有：有且仅有一个节点 k_0 ，它对于关系 N 来说没有前驱，称 k_0 为树的根节点。除 k_0 外， K 中的每个节点对于关系 N 来说，有且仅有一个前驱，可以有 m 个后继 ($m \geq 0$)。树是一种简单的非线性结构，所有元素之间具有明显的层次特性。

⑥ 图。由节点的有穷集合 V 和边的集合 E 组成。其中，为了与树形结构区别开，在图结构中常常将节点称为顶点。若有两个顶点 v_i, v_j 之间存在一条边，则用偶对 (v_i, v_j) 来表示无向边，用 $\langle v_i, v_j \rangle$ 来表示从 v_i 到 v_j 的有向边。

任务四 了解算法设计

1. 什么是算法？

算法是解决问题所采取的步骤和方法。人们的生产活动和日常生活离不开算法，很多时候都在不知不觉中使用算法。例如，购物的“算法”是：确定需购买的商品；确定选购的商场；到达商场；购买商品；返回。

计算机算法分为两大类：数值运算算法和非数值运算算法。数值运算算法的目的是求解数值，例如，求函数的定积分或方程的解等。非数值运算算法的范围很广，最常见的是事务管理领域，例如，成绩管理、工资管理、图书检索等。目前，计算机在非数值运算方面的应用远远超过了数值运算方面。

下面通过一个简单例子来说明如何设计一个算法。

【案例 1-1】计算 $1+2+3+\dots+100$ 。

【分析】

顺序算法

第一步：计算 1 加 2 得 3；

第二步：将 3 与 3 相加得 6；

第三步：再将 6 与 4 相加得 10；

……

最终算法描述中需要写 99 个步骤，显然此方法十分烦琐，不是最佳算法。

改进算法