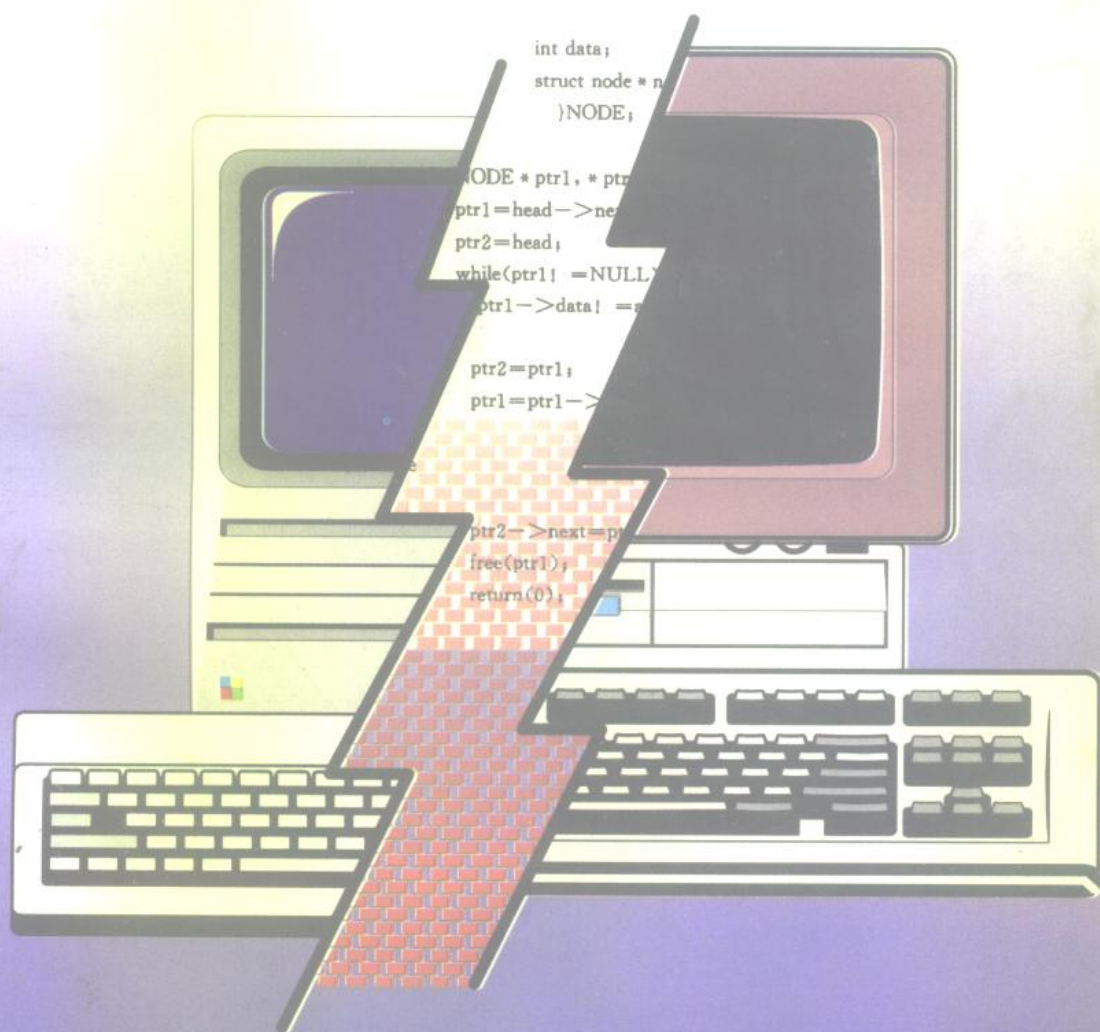


工程软件技术基础

 牛夏牧等 编著



哈尔滨工业大学出版社

TP311.5

N82

434192

工程软件技术基础

牛夏牧等 编著

哈尔滨工业大学出版社

JS/57/1/1

工程软件技术基础

Gongcheng Ruanjian Jishu Jichu

牛夏牧等 编著

*

黑龙江工业大学出版社出版发行

肇东粮食印刷厂印刷

*

787×1092毫米 1/16 印张 23.75 字数 582 千字

1997年8月第1版·1997年8月第1次印刷

印数:1—3 000册

ISBN 7-5603-1247-0/TP·102 定价:26.80元

前 言

目前,国内现有的软件技术基础教材有些内容已经陈旧,不能满足教学的要求,而且教材内容侧重理论性,工程应用性较差。本教材以“程序员级和高级程序员级软件考试大纲”的内容为主线,以工程实际应用为主导,以当前软件技术发展的方向、现状为依据,以及工程技术中所迫切需要的内容而编写的。它是掌握与开发各种系统软件、仪器软面板、用户交互平台软件、工程信息管理软件、网络工程开发软件以及各种实用工程软件等软件技术的基础内容。

本教材力图用较短的篇幅系统地介绍计算机软件技术基础知识,包括软件基本概念、数据结构、操作系统、计算机网络、数据库和软件工程等计算机软件技术基础内容,并剖析了常用的一些实际的系统工程软件。第一章,重点讨论软件的一些基本概念,以及软件的发展历史和现状。第二章讨论软件技术的重要基础知识——数据结构的内容,并用 C 语言给出了各种算法。第三章讨论 DOS 操作系统的内容,重点介绍 DOS 操作系统(V6.22)的系统结构、外壳(DOS-Shell)、内核(DOS-Kernel)和基本输入输出系统(BIOS)的结构与功能以及开发方法。同时介绍了 CD-ROM 的文件结构及 DOS 对 CD-ROM 的支持形式。第四章讨论 Windows 操作系统的内容,重点介绍 Windows 操作系统的组织结构、内存管理模式和操作环境等方面的知识及其开发方法。另外,对 32 位系统——Windows 95 也作了较详细的介绍。第五章讨论计算机网络系统,包括计算机网络的基本概念及其体系结构、Internet 广域互连网以及 Ethernet 局域网和 NOVELL NetWare 局域网的基本知识。第六章讨论数据库系统,介绍了 ISO 认定的 SQL 标准数据库语言以及 Foxpro 数据库管理系统。第七章,介绍软件工程的内容。

本书第一~第六章由牛夏牧编写;第七章由姜守达编写。哈尔滨工业大学计算机与电气工程学院孙圣和教授担任主审。

本书可作为非计算机专业本科生及研究生的教材,同时也可作为广大工程技术人员自学参考书。

由于水平所限,加上时间仓促,教材中的错误和疏漏在所难免,请读者批评指正。

作者于哈尔滨工业大学
1997 年 6 月

目 录

第一章 概论

- 1.1 计算机的组成 (1)
- 1.2 计算机软件 (2)
 - 一、计算机的各种语言 (2)
 - 二、计算机系统软件 (5)

第二章 数据结构

- 2.1 数据结构的概念 (9)
 - 一、数据结构的定义 (9)
 - 二、数据结构研究的两个问题 (10)
- 2.2 表结构 (10)
 - 一、简单线性表 (10)
 - 二、栈(特殊线性表) (13)
 - 三、队 (14)
 - 四、线性链表 (16)
- 2.3 树结构 (21)
 - 一、树的基本概念 (21)
 - 二、二叉树 (22)
- 2.4 检索 (26)
 - 一、线性检索 (27)
 - 二、对半检索 (27)
 - 三、分块检索 (28)
 - 四、散列检索 (29)
- 2.5 排序 (34)
 - 一、选择排序 (34)
 - 二、冒泡排序 (35)
 - 三、线性插入排序 (36)
 - 四、快速排序 (36)

第三章 操作系统(I)

- 3.1 绪论 (39)
 - 一、个人机的硬件资源 (39)
 - 二、PC 机的软件资源 (44)
- 3.2 DOS 的装入 (49)
 - 一、DOS 对存储器空间的分配 (49)
 - 二、DOS 的启动 (58)
- 3.3 DOS-Shell (62)
 - 一、DOS 内外部命令(以 MS-DOS6 为准) (63)
 - 二、磁盘的开工过程 (87)

三、DOS 的系统配置	(92)
四、DOS 两类程序(.COM 文件和 .EXE 文件)的结构与加载后映象	(104)
3.4 DOS-Kernel	(114)
一、引言	(114)
二、DOS 磁盘的组织机构	(114)
三、DOS 的文件管理机制	(121)
四、DOS 的系统功能调用	(131)
3.5 BIOS	(138)
一、只读存储器(ROM)的结构	(138)
二、BIOS 系统设计思想	(140)
三、硬盘的 ROM-BIOS 进入系统	(141)
四、BIOS 系统的结构与功能	(142)
五、BIOS 系统的 CMOS-RAM 参数的设置	(153)
3.6 CD-ROM 的组织结构	(159)
一、CD 光盘技术的发展	(159)
二、CD-ROM 的数据记录的物理格式	(161)
三、CD-ROM 数据记录的逻辑格式	(164)
四、DOS 操作系统对 CD-ROM 的支持	(168)
第四章 操作系统(II)	
4.1 绪论	(172)
4.2 WINDOWS 的组织结构	(173)
一、WINDOWS 的基本操作环境	(173)
二、WINDOWS 的内存管理	(175)
三、WINDOWS 的启动	(178)
四、WINDOWS 的组成部分	(180)
4.3 WINDOWS 系统的各种文件类型	(187)
一、WINDOWS 可执行文件的首部格式	(187)
二、图形文件(位图文件)格式	(194)
三、声波文件(.WAV)格式	(195)
四、WINDOWS 系统其他的文件格式	(196)
4.4 WINDOWS 的资源管理	(196)
4.5 WINDOWS 的外壳程序(I)——程序管理器	(197)
一、程序管理器的基本环境	(198)
二、程序组和程序项	(200)
4.6 WINDOWS 的外壳程序(II)——文件管理器	(202)
4.7 WINDOWS 的主要桌面工具	(204)
一、WINDOWS 的主要应用程序	(204)
二、WINDOWS 的信息交换机制——裁剪板、DDE、OLE	(206)
三、WINDOWS 的几项控制技术	(209)
四、WINDOWS 的汉化环境及开发环境	(214)
4.8 WINDOWS95 操作系统	(216)
一、概述	(216)
二、桌面	(216)

三、窗口	(222)
四、WINDOWS3.x 与 WINDOWS95 的常用工具比较	(223)
五、WINDOWS95 的引导配置	(226)
第五章 计算机网络	
5.1 计算机网络的概念	(232)
一、计算机网络	(232)
二、计算机网络的结构	(232)
三、ISO/OSI 参考模型	(237)
四、局部网络定义和分类	(239)
5.2 Ethernet 以太局部网络介绍	(239)
一、概述	(239)
二、Ethernet 网络的功能介绍	(245)
5.3 NOVELL 局部网络介绍	(248)
一、NOVELLNetWare 发展情况	(248)
二、NOVELLNet 的结构	(249)
三、NOVELLNet 的特点	(253)
四、NOVELL 网络的规划与安装	(254)
五、NetWare 的文件系统组织方式	(263)
六、NetWare 网络的管理方式	(268)
七、作为用户使用网络	(275)
八、WINDOWS95 下与 NetWare 网络互连	(281)
5.4 INTERNET 国际互连广域网	(284)
一、概述	(284)
二、INTERNET 网的体系结构	(285)
三、INTERNET 网的应用	(287)
四、接入 INTERNET 网	(290)
五、校园网	(291)
六、通过 WINDOWS95 上 INTERNET 网	(291)
第六章 数据库系统	
6.1 数据管理方法的发展及基本概念	(298)
一、数据管理技术的发展	(298)
二、信息模型——ER 模型	(299)
三、数据库系统结构	(300)
6.2 三种数据模型	(301)
一、层次模型	(302)
二、网状模型	(302)
三、关系模型	(303)
6.3 典型的数据库系统	(303)
一、关系模型的基本概念	(304)
二、关系数据语言——SQL 标准数据库语言	(306)
6.4 关系数据库设计理论	(309)
一、数据库设计过程	(309)
二、关系数据库设计理论	(310)

6.5 数据库保护	(312)
一、安全性	(312)
二、完整性	(313)
三、并发控制	(313)
四、恢复	(313)
6.6 微型机数据库管理系统 FOXPRO	(313)
一、软件的基本构成	(314)
二、FOXPRO 的数据类型、常量及变量	(319)
三、FOXPRO 的配置设定	(321)
四、FOXPRO 的工作环境	(323)
五、FOXPRO 数据库的基本格式	(323)
六、数据库的建立	(324)
七、信息查询	(328)
八、报表打印	(330)
九、计算统计	(332)
十、FOXPRO 的网络功能	(335)
十一、FOXPRO 的可视化功能	(337)
十二、一个图书信息管理系统例子	(344)
第七章 软件工程	
7.1 软件工程化问题	(348)
一、软件技术发展中的难题——软件危机	(348)
二、软件研制工程化	(349)
三、软件的生命期	(349)
7.2 软件的工程化过程	(350)
一、系统分析	(350)
二、实施计划报告	(351)
三、软件需求分析	(351)
四、软件设计	(353)
五、程序编制	(358)
六、软件测试	(360)
七、软件维护	(362)
7.3 面向对象的编程方法(OOP)	(364)
一、OOP 的概念	(364)
二、面向对象的软件设计方法学	(366)
三、面向对象的分析(OOA)和设计(OOD)	(367)
四、结论	(369)
参考文献	(371)

第一章 概 论

1.1 计算机的组成

自 1946 年世界上出现了第一台电子计算机以来,不到半个世纪的时间,计算机系统得到了飞速的发展,从单片微型机到超大型计算机无不得到广泛的应用。

就计算机的整体组成来讲,按现行的分法计算机可分为软件和硬件。

1. 计算机硬件

硬件通常由三个基本部分组成:

(1)中央处理器(CPU)。它是解释和执行指令的部件,是计算机的核心。

(2)主存贮器。它是用来存放当前要执行的指令和数据的存贮部件,它主要有两种结构,一是 RAM(随机存贮器),另一是 ROM(只读存贮器)。

(3)外部设备。外部设备包括:外存贮器(如磁盘、磁带、光盘等);输入/输出设备(如键盘、光笔、扫描器、CRT 等)。

2. 计算机软件

(1)计算机语言

开始,一台计算机中没有装入任何软件,我们称之为裸机,裸机只认识“0,1”代码。人们为了让计算机为人工作,必须按一定逻辑输入这些代码,通常称这些代码为机器语言。

为了方便地使用计算机,人们把这些代码用助记符的形式表现出来,这些助记符是英文名称的缩写,比较易记易编,这就是汇编语言,又称为低级语言。用汇编语言编写好程序之后,再利用特殊的程序(汇编软件)把它自动地翻译成机器语言,就可以让计算机按要求工作了。这样,机器语言是其最终的目的,故又称为目标程序(目标代码),而汇编语言编制的程序称为汇编源程序。

采用汇编语言编制的程序,仍要记住机器指令的助记符,且所编的程序只针对某一类机器(因为它与机器代码密切相关)。为解决这些问题,出现高级语言程序,它与机器硬件完全脱离,只关心算法。通过特殊的程序(编译软件)把源程序翻译成目标程序,以让机器工作。由于这种语言设计的程序比较接近于人们的习惯,编出的源程序与具体机器的指令无关,可适用于任何机器(对不同的机型可用不同的编译软件进行翻译),因此,人们把这种程序设计语言称为高级语言。

(2)系统软件

系统软件是为帮助用户编写与调试应用程序而提供的系统程序集合。它可以用低级语言编写,也可以用高级语言编写。

系统软件通常包括:操作系统、监控程序、诊断软件、调试程序、汇编软件、编译和解释软件、连接定位程序、仿真软件、数据库管理系统以及网络管理系统等等。

(3)应用程序

应用程序是用户自己运用某种计算机语言(低级语言或高级语言)编制出的源程序。换

2. 高级语言

高级语言又称为程序设计语言,它的产生与发展是新科学的智力成就之一。它的成功主要在于提高了程序人员的工作效率和可移植性。从50年代后期FORTRAN语言问世以来,30多年中语言的发展是相当迅速的。目前世界上使用的各种程序设计语言不少于数百种,从使用的角度来看,一类是以FORTRAN、COBOL为代表的强制性语言,即通过赋值逐个改变变量的状态以达到其基本目的,所以强制性语言共同的依赖性在于变量和赋值;另一类是以LISP语言(表处理语言)为代表的函数型语言,它是通过函数递归或组合来实现其基本目的的。在过去的40多年里,强制性语言的吸引力是公认的,而应用性语言直到后阶段才引起人们的重视,但现在已充分显示出它的成功之处。

FORTRAN语言和COBOL语言是两种比较成熟的语言,在科学计算中(FORTRAN)和商业数据处理中(COBOL)曾起着压倒优势的地位。用于结构设计的PASCAL语言以及它衍生出的有关语言,在大专院校中曾一直处于主导地位,并广泛地用于系统程序设计,目前,以Borland公司为代表推出的新一代PASCAL语言——Delphi语言,使得PASCAL语言达到了一个崭新的阶段。PL/1语言是一种多用途语言,它综合了FORTRAN,ALGOL和COBOL语言的特性,具有“公共汽车”之称;Simula语言是仿真语言;LISP和SNDBOL语言更多地用于人工智能;APL作为一种函数型语言,也一直在吸引着它的热心追随者们用它来解决各种类型的计算问题。前面所述高级语言是一些面向问题或面向应用的语言,利用这类语言在计算机上解题时,无需了解计算机的内部逻辑,只需选择适当的数据结构和正确的算法,就可编制出解题程序。

70年代初,以关系模型为基础的数据库管理系统的出现,以SQL(结构化查询语言)为代表的一类高级查询和操纵语言相继问世。用这类语言查询和操纵数据库,无需详细指明如何做,只要给出需要做什么或要求是什么即可以。这类语言称为非过程化语言,它的突出特点是能够较方便地为非程序人员所直接使用,加速程序设计的进程。目前有关数据库管理系统,在微机上走了从DBASE到FOXBASE乃至到FOXPRO2.5/2.6以及VisualFoxPro3.0等这样一个过程。由于SQL是一种日趋流行的标准关系型数据库管理系统语言,FOXPRO则尽量向SQL方向靠拢,例如,FOXPRO的相关举例检索(RQBE)的功能就是通过SQL的Select命令处理的,其数据检索方便灵活,不过目前的FOXPRO还不全面地支持SQL。

由美国国防部研究室资助的Ada语言提供了很好的程序模块化、可移植性、可扩充和抽象性,很多国家的军方均选用为军用语言。1979年由苏黎世联邦理工学院信息研究所的N. Wirth(PASCAL语言的创始人)指导下的一个小组,设计了一种通用系统程序设计语言,称为Modula,提供了灵活而高效的实现并发程序设计的设施。Ada和Modula是描述并发特性的两个著名语言。

作为单片微型机而言,由于直接与应用紧密相联,同时又与硬件关系密切,所以,一直处在一种汇编语言进行程序设计的状态之中。近几年来产生的一种单片机高级语言PL/M语言,已经彻底地改变了这种状态,使得编程和调试变得极为方便,而且可移植性和通用性也成为可能。

3. C 语言

C 语言是具有类似于汇编语言的程序设计能力的最有影响的一种通用语言。C 语言是 1972 年由美国的 Dennis Ritchie 设计发明的,并首次在 UNIX 操作系统的 DECPDP-11 计算机上使用。它由早期的编程语言 BCPL(Basic Combind Programming Language)发展演变而来的。在 1970 年,AT&T 的 Bell 实验室 Ken Thompson 根据 BCPL 语言设计出较先进的并取名为 B 的语言,最后导致了 C 语言的问世。

C 语言把高级语言的基本结构和语句与低级语言的实用性结合起来。C 语言可以象汇编语言一样对位、字节和地址进行操作,而这三者是计算机最基本的工作单元。C 语言同时又是结构式语言,其显著的特点是代码及数据的分隔化,即程序的各个部分除了必要的信息交流外彼此独立。C 语言是以函数形式提供的,这些函数可被用户方便地调用,并具有多种循环、条件语句控制程序流向,从而使程序完全结构化。

C 语言主要以 Microsoft 公司和 Borland 公司两家的开发版本最有成就,Microsoft C 语言版本已发展到 C/C++7.0、VC2.0、VC4.0...;BorlandC 的版本已发展到 BC++3.1、BC++4.0、BC++4.5、BC++5.0。目前由 C++产生的一种面向对象编程方法的技术(OOP 技术)使得整个软件的编制方法出现了一次革命性的转折,因此,人们把 90 年代的程序语言设计称为对象语言时代。

4. 描述性语言

描述性语言的典型代表就是 Prolog 语言。这是随着第五代计算机发展而发展起来的语言。人们常称该语言为人工智能程序设计语言。随着计算机应用的不断发展,把人工智能程序与传统的程序融合在一起的需求稳步增长,对更友好的用户界面及智能化程序的需求也不断增长,因此人们对开发 Prolog 应用程序的要求与日俱增。

所谓的描述性语言,这意味着只要给出必要的事实与规则,Prolog 就能使用演绎推理解决问题,这与 BASIC、PASCAL 和 C 等传统的过程性程序设计语言不同。在过程性语言中,程序必须精确地向计算机提供解决问题的每条指令,即在计算机解题前,程序员就必须知道如何解决此问题。而 Prolog 的程序员则不然,他们只需提供对要解决的问题求解规则的一个描述。基于这种描述,Prolog 就能自动确定找到问题解的方法。Prolog 的重要特长也同样适用于任何其它应用领域:通过让程序员描述对象及过程间的逻辑关系,复杂的问题就变得易于解决,所产生的程序也更易于维护。由于 Prolog 的这种描述性特性,一些常见的错误(诸如循环次数太多或太少)就再也不会出现。此外,Prolog 还鼓励程序员在开始编程时对问题进行结构清晰的描述,这样,在经过一段时间的实践之后,Prolog 就不仅能被用作实现工具,还能被当作一种描述工具。

由于 Prolog 的逻辑方法非常自然,程序设计新手及熟练程序员都能在极短的时间内开发出功能很强的应用程序,而且还可以极其方便、迅速地开发诸如知识库系统、专家系统、自然语言接口、智能化信息管理系统等应用系统软件。

5. Java 语言

Java 语言的出现可谓是形式发展的需要。SUN 公司为了开发一种消费电子产品使用的先进软件,最初决定选用流行的面向对象语言 C++。在开发过程中,SUN 公司的工程师们发现 C++存在许多问题。随着工作的深入进行,他们觉得必须研制一种新的面向对象的语言,才能彻底解决 C++固有的缺欠,于是 Java 语言应运而生。SUN 公司推出的语言非常适合于企业网络和 Internet 环境,现在已成为世界上最受欢迎、最有影响的编程语言。Java 语

育有许多值得称到的优点,如简单、面向对象、分布式、解释性、可靠、安全、结构中立性、可移植性、高性能、多线性、动态等。随着 Java 的应用日益广泛,Java 的实际意义也越来越明确。SUN 公司与目前世界上占有 Internet WWW 浏览器市场份额最大的 Netscape 公司合作开发了旨在把 Java 作为一种适合于 Internet 的全新计算机技术的基础的新一代 Java 语言——一种开放的、跨平台的对象描述语言 JavaScript。其可以工作于 Intel 微处理器,同时又不需要 Microsoft DOS、Windows 操作系统。如这项计划获得成功,计算机厂就可以利用标准的 PC 器件来生产极便宜的“Internet Web 终端”(又称为网络计算机,NC 机),这是因为这种机器只需很少的存储芯片和其它的高价器件,绝大多数计算任务都可由 Internet 来完成。这实际上对目前的 Microsoft—Intel 体系是一个非常大的冲击。

6. 可视性编程语言

Windows 操作系统出现之后,编制 Windows 应用程序则是很复杂的事情,它要求程序员要具有较高的编程水平,为了克服这一问题,Microsoft 公司有远见地推出了编程“可视化”的概念。所谓“可视化”是依照面向对象编程方法(OOP 技术),编程人员只要按照系统提示的各种工具,利用鼠标即可完成繁重的 Windows 应用程序的界面(对象)设计工程。往往一种可视性编程语言,是以某种高级语言的语法为其规则的。最有代表性的可视性编程语言是 Microsoft 公司的 Visual Basic,其语法是沿用 Basic 的语法规则,而其编程方式则是按可视化编程方式。继 Visual Basic 之后,Microsoft 公司由先后推出了 Visual C++ 和 Visual Fox-Pro 等可视性编程系统。目前,可视性编程语言中最为理想的可视性编程系统是 Borland 公司推出的 Delphi,它不仅克服了以前的各种可视性编程系统中的许多缺点,而且 Delphi 的出现使得可视性编程系统进入了一个相当成熟的阶段。另外,在传统的 C 语言编程过程中,编译系统为了给程序员提供足够的灵活性,对于程序中的一些小的错误是忽略不计的,这对于一些中小程序的编制而言是有很大的优越性的,但是对于编制极其庞大的 Windows 应用程序来说,这些小的错误往往的灾难性的。而 Delphi 所采用语言是结构非常严谨的 Pascal 语言,这就克服了上述问题。

二、计算机系统软件

1. 操作系统

操作系统是最基本的系统软件,是硬件机器的第一级扩充。它是这样一个程序集合:它的主要功能是使用户可以充分地利用系统的资源,同时又提供各种友善的方式来帮助用户方便地使用资源。有关操作系统的进一步的内容请参阅本书第三章与第四章。

2. 网络操作系统

计算机网络是适应计算机分布式应用的需要,也是计算机技术和通信技术高度发展、密切结合的产物。其特点是综合利用现今各种重要的信息技术的研究成果,把分布在广泛区域中的众多信息系统有机地连接在一起,构成一个规模更大、功能更强、可靠性更高和用户界面友好的综合信息处理系统。自从 NOVELL 公司推出 NetWare 系统软件以及 Microsoft 公司推出 Windows NT 系统软件之后,网络上的管理才有了操作系统这样一个概念,因此,NetWare 与 Windows NT 通常称为网络操作系统。有关这方面的内容读者参阅第五章。

3. 数据库管理系统软件

数据库技术是管理数据的一种最新方法。数据管理是指对数据的组织、编目、定位、存贮、检索和维护等,它是数据处理的中心问题。数据库技术在 60 年代开始出现,主要适应管

理信息系统对数据管理要求,30 多年来,在理论上、实现技术上都得到很大发展,开发出许多数据库管理系统软件,使得数据库技术得到广泛的应用。现在数据库技术不仅应用于事务处理,而且进一步应用于工程设计,甚至在专家系统、人工智能方面也得到应用。有关数据库方面的具体内容,请参阅本书第六章。

4. 编译软件

普通的计算机不懂得高级语言,也不懂得它自己的汇编语言。因此,计算机无法直接执行用高级语言(或汇编语言)编写的程序,必须经过语言处理软件的分析,然后将其翻译成机器语言程序,供计算机执行。语言处理软件的工作方式大致上分为编译和解释两类。相应的语言处理软件分别称为编译软件和解释软件。

编译软件是指这样一种软件,它把由高级语言编写的程序(称为源程序)翻译成与之等价的低级语言程序(称之目标代码或结果程序)。如果把某汇编语言程序翻译成等价的机器语言程序,这样的语言处理软件称为汇编软件。

(1) 汇编基本原理

汇编语言的程序和对应的机器语言程序有相同之处,又有不同之处。相同之处是,程序主体部分几乎是一一对应的;不同部分是数码换成了符号,地址换成了名字,另外还增加了关于工作单元和常数单元的成分。这些不同之处正是汇编语言的优点,使得汇编语言的程序和机器语言程序比较起来,好写、好读、好改。

汇编语言也是依赖于机器的,因此称它为面向机器的语言。使用时必须了解机器的某些细节,如累加器、每条指令的执行速度、内存容量等等,但正由于它依赖于机器,所以就可以与机器语言程序一样可结合机器特点编出高质量的程序,即程序又短,执行速度又快。所以直至今日,汇编语言仍起着重要作用,在一些国外计算机公司中仍用汇编语言编写系统软件,来保证高质量的软件功效。

汇编软件的主要功能是把用汇编语言编写的源程序加工成机器语言写的目标程序。其过程可以分成三步:

- 用汇编语言编制出源程序;
- 将源程序输入到计算机内,由汇编软件把它加工成计算机能够执行的目标程序;
- 执行目标程序,得到计算结果。

下图示出了上一过程。

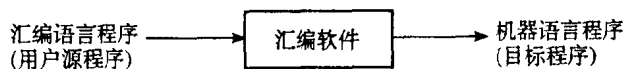


图 1.2 汇编过程

源程序是由一串符号化的指令组成,国内采用标准的 ASCII 码,当汇编软件加工源程序时,总是从头到尾,一个符号接着一个符号的阅读,称为扫描源程序,从头到尾扫描一次源程序为扫描一遍,一般汇编软件是两遍扫描源程序的。第一遍把源程序中所有出现的名字进行造表,确定每个名字将占用的内存位置。第二遍扫描时,按造出的表,把每条符号化指令代换成数码形式的机器指令。

此外汇编软件还具有一些附加的功能。

查错:当用户写的汇编语言源程序不符合汇编语言所要求的书写格式、不符合语法要求时,汇编软件指出源程序的某个位置出了什么性质的错误。

修改:汇编软件提供修改源程序的简便办法,用户把修改的要求提供给汇编软件,由汇编软件实现对源程序的自动修改。

打印:在汇编过程中,当发现错误时,将错误信息打印出来,必要时还可打印出名字表及目标程序,还可打印出修改后源程序的文本等。

实用的汇编语言中还可以包括一些用户编程时很需要的指令,这些指令并无对应的机器指令,称为伪指令。

(2)编译基本原理

编译软件的工作过程是把高级语言写的源程序译成目标程序。采用的方法是先分析词法和语法,然后进行综合,包括代码优化、存贮分配和代码生成三部分。词法分析程序是编译软件中的最基础的程序。要完成这么复杂的任务,编译软件要对源程序进行多次扫描。每扫描一遍完成一项或几项任务。不同语言的编译软件采用的具体做法不同。词法分析、语法分析、代码优化、存贮分配和代码生成五部分都生成各种表格。每扫描一遍造出或修改各种表格,交下一遍使用,完成各遍之间的交换工作,每遍扫描就把源程序或内部表示形式从头到尾查看一遍,同时完成编译过程中的一部分工作。如图 1.3 所示。

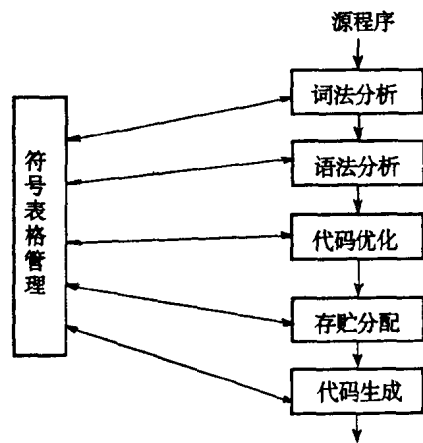


图 1.3 编译基本过程

词法分析过程中造出各种表格,登记了词法分析扫描中所获得的有关信息。在以后加工源程序的每个阶段中,还要不断地在这些表格中增加一些信息,或对原有的信息作些修改、增补。在生成机器指令的阶段,就要在表格中增设各变量(或标号等)所分配的内存地址,最后编译软件就根据这些表格生成机器指令,即目标指令。在这些表格中最为重要的,有代表性的是名字特性表和数表。每当读到一个新的名字时,就在名字特性表中增添一项,登记名字和有关特性的信息。名字特性表主要是生成目标程序用的,词法分析程序还把源程序中的数加工成机器内部表示形式的数并送到数表里。词法分析程序加工后,需要进行换码。所谓换码是将源程序中的字符换成代码。

语法分析程序的重要功能之一是找出源程序的语法错误,有时还能发现一些语义错误。语法分析的另一重要功能是通过语法分析弄清了句子的结构,以便将其翻译成目标程序。实际上,语法分析程序一方面将各种表格进一步改造或写一些从语法分析得到的信息;另一方面,经过语法分析后就可以编出接近于目标程序的过渡形式。这种过渡形式近似于汇编语言那样的格式,所有程序中的名字还只能是代码。

代码优化是编译技术中重要的问题之一。它为保证目标程序更加简洁、更加合理,为生成高质量的目标程序打下基础。代码优化的内容很多,例如合并常数,即如果一个运算的操作对象都是已知的常数,在编译时就把它算出来,将结果存入数表,目标程序运行时就简单了。又如删去多余运算,即如果一个运算和它前面的某一运算完全相同,而且参加运算的操作数没有改变,这时,后一个运算就完全可以去掉,因为它是多余的,再例如将表达式外提,即如果在循环内部有一组运算,在每次循环时,它的计算结果永远不变,而每次循环工作中都重复计算一遍,那么可以把循环内部保持不变的运算提到循环外面去做。这样执行时间大大节省。优化做得好的编辑软件,其目标程序的长度和执行时间,可以接近于手编的程序。

存贮分配是为了在代码生成前先确定程序、变量以及常数在内存中安放的地址。把分配的地址填在表中,在代码生成时用它生成目标程序。

代码生成程序是编辑软件的最后一道工序。它从头到尾依次加工由语法分析后得到的程序,生成对应的机器语言写成的指令,或类似于汇编语言形式的半成品指令,同时不断地查看各种表格,确定运算对象和结果存放的存贮地址。目标程序生成后与各种表格存放到外存贮器中,等待装配运行。

在一个大的程序中,其中有些程序块是程序库中的标准程序和标准子程序,也有些程序是用其它语言编写的,这些程序块需要装配在一起组成一个可运行的目标程序,才能进行计算。因此,首先要求编译软件能对每个程序块进行独立编译。每个程序都不能单独运行,因为程序中所涉及的地址会相互重迭,必须把各程序中所涉及的地址经过修改,重新确定其地址,使它们之间既不重迭、也很紧凑,既保证运行的正确性,又不加长程序,完成此项工作的程序称为装配程序。就是把几个目标程序装配成一个大的目标程序。

此外,编译软件还提供修改源程序的功能和较完善的调试措施,帮助用户检查程序的错误。

解释软件与编译软件不同之处在于接受源程序时,并不是一下子把它全部翻译出来执行,而是边翻译边执行,执行后再取下一个语句。这种工作方式非常适合于人通过终端设备与计算机会话。但它存在一个严重的缺陷——执行速度慢,即如果源程序中出现循环,重复地执行一组语句,那么解释程序也重复地解释执行多遍。图 1.4 示出了解释软件的四种处理过程。

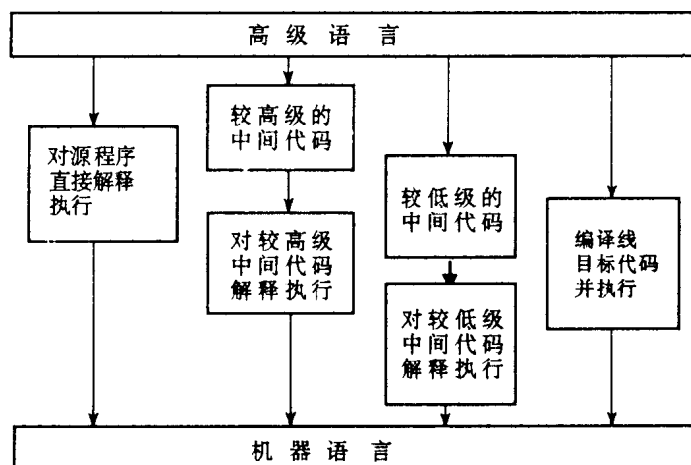


图 1.4 解释软件的四种处理过程

本书用一章的篇幅介绍了数据结构方面的知识。数据结构是软件的基础,在计算机科学中是一门综合性的专业基础课。汇编程序、编译软件、操作系统、数据库系统、网络操作系统、人工智能等均要用到数据结构的概念和算法。数据结构是包括分析研究计算机加工的数据对象的特性,选择合适的数据结构和存贮结构以及相应的算法,并且包括各种算法在时间空间上的分析技巧。通过数据结构的学习可以加强进行复杂程序设计的训练。

另外,本书分别用了相当篇幅讨论了操作系统、计算机网络和数据库系统以及软件工程等重要系统软件的内容,对飞速发展的计算机科学而言,这些无疑是十分重要的。

第二章 数据结构

2.1 数据结构的概念

数据结构是计算机科学的重要基础,近年来已发展成一门专门学科。数据结构的内容深及到计算机系统的硬件和软件。数据结构的知识对设计和实现计算机系统软件,如操作系统、编译程序、数据库管理系统等,是十分重要的。它对编写应用软件的人来说也是必不可少的。关于数据结构的专门著述很多,在这里只介绍部分常用数据结构的基本知识。

一、数据结构的定义

人对计算机的操作过程大致可以用图 2.1 表示。

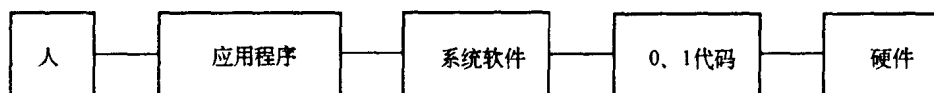


图 2.1 人对计算机的操作过程

一方面,人必须运用某种计算机语言把要做的工作编写成应用程序,通过系统软件的支持实现对计算机的操作。这程序实际上就是一系列的字符、数字的有机的集合。另一方面,计算机对这些字符、数字进行处理变成 0、1 代码以完成任务。可见,人和计算机所面对的就是这些字符和数字。例如,要想让计算机实现 $X=A * B+C/D$ 的算式,可用下述 FORTRAN 语言实现:

```
float A,B,C,D,X
A=1.0
B=2.0
C=3.0
D=5.2
X=A * B+C/D
write(*,*)X
stop
end
```

这里,A,B,C,D,X,write 等等为字符,2.0,5.2,4.0,1.0 等等为数字。

<定义>数据就是一些可输入到计算机中去的描述客观事物的符号(字符和数字)。每一个数据又可称为数据元素。

再看上述程序,人必须把各种数据按一定的逻辑进行有机地组合构成程序,并利用某个编辑软件(如 ENDLIN.EXE,NE.EXE 等)输入到计算机中,再利用 FORTRAN 语言的编译软件(如 FOR1.EXE 和 FOR2.EXE)对上程序进行编译,检查是否有错误(同时提示用户进行重新修改),并把之变成机器可执行的 0、1 目标代码,然后利用连接程序(如 LINK.EXE)进行连接供机器运行。这些都需要对数据进行运算(如插入,删除,修改,检索和排序等)。同时,计算机还必须把上述程序按一定的方式存贮和处理。