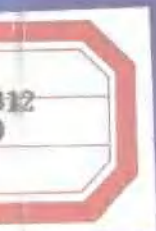


黑龙江省高等学校统编教材

FORTRAN90 程序设计



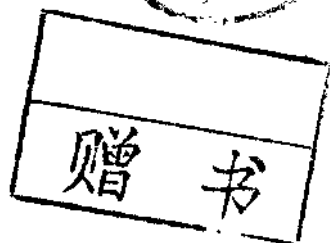
主编 马瑞民 衣治安



哈尔滨工程大学出版社

FORTRAN90 程序设计

主 编 马瑞民 衣治安
副主编 王 新 刘华莹
孙 雷 梁久桢
主 审 武常岭 陈仁华



00425537

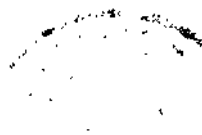
哈尔滨工程大学出版社

内 容 简 介

FORTRAN90 是 ISO(国际标准化组织)于 1991 年推出的最新标准,我国国家标准是 GB/T 3057-1996。除了保持 FORTRAN77 的全部优点之外,又增加了许多具有现代特性的功能,使它成为具有良好的结构性、鲜明的时代特性的程序设计语言。

本书按 FORTRAN90 组织各章节标题,重点介绍了三种控制结构、过程、数组、派生类型与指针等内容,其中包括了 FORTRAN90 新引入的各种概念与功能,如递归过程、派生类型、指针、动态数据结构、接口块、模块等。本书叙述详细、深入浅出、循序渐进,并配有大量的例题和习题。

本书适合于作为高等院校高级语言类课程的教材,也适用于作为成人教育的教材和工程技术人员自学参考书。此外,有与此书配套的课件及例题盘。



FORTRAN90 程序设计

FORTRAN90 CHENGXUSHEJI

主编 马瑞民 衣治安

责任编辑 朱春元

哈尔滨工程大学出版社出版发行

哈尔滨市南通街 145 号 哈工程大学 11 号楼

发行部电话(0451)2519328 邮编:150001

新华书店经销

东北农业大学印刷厂印刷

开本 787mm×1092mm 1/16 印张 17.5 字数 406 千字

1998 年 4 月第 1 版 1998 年 4 月第 1 次印刷

印数:1~4000 册

ISBN 7-81007-833-X

TP·71 定价:22.00 元

黑 龙 江 省
计算机基础教育教材编审委员会

主 任 董 浩

副主任 王义和 武常岭 王明志 张 桥

委 员 吕其诚 马瑞民 安齐国 张 伟

常健斌 于之硕 滕 虹 赵连江

陈守厂 耿于林 于水源 陈荣耀

周树杰 邓文新

序 言

面对 21 世纪信息化社会 and 科技飞速发展的挑战,对高等院校人才素质培养和知识结构提出了全新的挑战。计算机在经济和社会发展中起着越来越重要的作用,培养各行各业的高级专门人才时,计算机基础知识和应用能力不仅是本、专科学生必备的知识和能力,而且计算机已成为高等教育各学科的重要组成部分。因此,加强高等学校的计算机基础教育工作,使教学质量尽快上一个新台阶,是目前高等学校的重要任务之一。

黑龙江省教委十分重视计算机基础教学工作,为进一步推进此项工作的开展,制定了“黑龙江省普通高校非计算机专业计算机教学基本要求”,并在全省高校中开展了计算机基础知识和应用能力等级考试,引起了各高校对计算机基础工作的重视,取得较满意的效果。

为把省内各高校计算机基础教学提高到一个新水平,省教委把非计算机专业学生的计算机基础教学课程的教材建设,作为一项教学工作的重要基础建设来抓,成立了黑龙江省计算机基础教育教材编审委员会,组织了 20 余所高校中长期从事计算机基础教学的教师,按照“黑龙江省普通高校非计算机专业计算机教学基本要求”通知精神,并参考“计算机基础知识和应用能力等级考试大纲和样卷”一书,编写了这套“黑龙江省高等学校统编教材”。本套教材包括:“微机应用基础”、“BASIC 语言程序设计”、“FORTRAN 语言程序设计”、“PASCAL 语言程序设计”、“C 语言程序设计”、“计算机基础与 FoxBASE + 程序设计”、“FORTRAN90 程序设计”等教材。

我们相信,本套教材在全省高校推广使用过程中,通过广泛听取师生的意见,对教材内容的规范性、例题深度及适用性等进行不断地修正与完善,使此教材能够成为促进我省高校计算机基础教学工作的,并受师生欢迎的高质量的教材。

教材编审委员会

1998 年 1 月

前 言

当今的电子计算机都属于冯·诺依曼型计算机,也称存储程序式计算机,其特点之一是计算机的操作由程序自动控制,而各种程序的研制有赖于计算机语言的运用。FORTRAN语言作为最早出现的计算机高级语言,其标准化程度最高,应用范围较广,在全世界有着十分广泛的用户。

高等院校作为培养高级工程技术人才的主要基地,对大学生进行计算机文化教育是不可少的。1980年前后,我国工科院校曾把FORTRAN66作为计算机基础课程的主要高级语言之一,在80年代中后期又换代为以FORTRAN77为主。90年代FORTRAN90问世以后,人们很快发现它在保留原有的FORTRAN语言特点的基础上,吸取了其它高级语言的优点,增添了许多具有现代特性的语句及功能,融入了递归程序设计、数组直接运算、派生类型、指针、过程的关键字变元调用等技术,使FORTRAN语言的功能有了极大的增强。同时,它还把FORTRAN 77中分区书写格式、使用语句标号的有关语句以及数据块子程序等不符合现代特性的语句或项目列为过时特性,这就使FORTRAN语言具有了很强的活力和竞争力。另外,由于FORTRAN90增加了字符种别的概念,克服了在信息处理方面适应能力弱的缺点,它的语法表述使用BNF范式,其表述方法严谨、规范,体系性非常强。这些都使FORTRAN90成为工科高校计算机基础教育的首选高级语言之一。

本书是通过FORTRAN90语法及功能的叙述重点介绍程序设计方法的一本教科书,适合于理工科高校各专业学生使用,也可作为工程技术人员的培训教材。由于书中采用了由浅入深、循序渐进的讲述方式,并配有大量的例题和习题,也适合于作为自学参考书。

本书有11章,第1章主要介绍计算机常识,第2章介绍FORTRAN90基础知识,第3章介绍FORTRAN90的基本语句,第4章和第5章分别讲述选择结构和循环结构,第6章讲述程序单元和过程,第7章讲述数组,第8章讲述字符型数据,第9章讲述派生类型和指针等概念,第10章介绍文件系统,第11章介绍FORTRAN 77中已过时的特性。书后附有3个附录。

参加本教材编写工作的都是常年在计算机教学一线的教师。本书由马瑞民、衣治安任主编,由王新、刘华莹、孙雷、梁久祯任副主编,由武常岭、陈仁华主审。本教材的授课时间为48至80学时(含上机),教师可以根据授课对象和教学需要选讲部分内容,但作为一门完整的课程体系,不讲的内容应由学生自学完成。另外,还可根据需要增加一些课外上机训练的机会。

本书编写过程中一直得到王义和、武常岭、张桥和黑龙江省高等院校计算机基础教育研究会常务理事会的指导和大力支持,并得到哈尔滨工程大学出版社的大力帮助,大庆石油学院计算机科学系的老师们也对本书提出过许多意见,在此致以诚挚的谢意。

由于编者水平有限,难免有不足之处,恳请批评指正。

《FORTRAN90程序设计》编写组

1998年2月

目 录

第 1 章 程序设计基础知识	1
1.1 计算机简介	1
1.2 程序设计语言	5
1.3 算法的描述	7
1.4 计算机软件	11
习题 1	13
第 2 章 FORTRAN90 基础知识	14
2.1 FORTRAN 语言的发展	14
2.2 源程序及其构成	15
2.3 语言元素	18
2.4 程序单元概念	30
习题 2	33
第 3 章 基本语句	34
3.1 类型说明语句	34
3.2 算术表达式和赋值语句	37
3.3 输入与输出语句	45
3.4 输入与输出编辑符	50
习题 3	55
第 4 章 选择结构程序设计	57
4.1 关系表达式与逻辑表达式	57
4.2 IF 语句	61
4.3 IF 结构	62
4.4 CASE 结构	67
4.5 程序举例	70
习题 4	73
第 5 章 循环结构程序设计	75
5.1 引言	75
5.2 无循环变量的 DO 结构	76
5.3 带循环变量的 DO 结构	82
5.4 DO WHILE 结构	88
5.5 DO 结构嵌套	90
5.6 隐含 DO 循环	94
5.7 程序举例	95
习题 5	100

第 6 章 程序单元和过程	103
6.1 概述	103
6.2 函数子程序	104
6.3 子例子程序	108
6.4 虚实结合	109
6.5 子程序的嵌套调用	112
6.6 模块	116
6.7 递归过程	121
6.8 其它部分	126
6.9 程序举例	132
习题 6	137
第 7 章 数组	139
7.1 一维数组	139
7.2 多维数组	148
7.3 数组运算	153
7.4 数组作为过程变元	158
7.5 程序举例	166
习题 7	172
第 8 章 字符型数据	175
8.1 字符型数据基础	175
8.2 字符型数据的输入与输出	178
8.3 字符型数组	180
8.4 用于字符处理的内在函数	182
8.5 字符型数据作为过程的变元及函数值	184
8.6 程序举例	187
习题 8	190
第 9 章 派生类型和指针结构	192
9.1 数据结构与派生类型	192
9.2 指针	200
9.3 链表	206
9.4 二叉树	217
9.5 指针与数组	222
9.6 指针与过程	225
习题 9	229
第 10 章 数据文件	231
10.1 文件概述	231
10.2 对文件的基本操作	232
10.3 顺序文件的存取	236
10.4 直接文件的存取	239
10.5 INQUIRE 语句	241

10.6 无格式文件	243
习题 10	245
第 11 章 FORTRAN 77 的过时特性	246
11.1 过时的书写格式	246
11.2 过时的控制转移方式	247
11.3 数据说明的过时形式	249
11.4 程序中的过时功能	250
11.5 其它过时语句	252
附录 I ASCII 字符集	255
附录 II FORTRAN90 的语法描述	256
附录 III FORTRAN90 的内在过程	264
主要参考文献	269

第 1 章 程序设计基础知识

本书主要讨论用高级语言 FORTRAN90 编写程序的若干问题。由于程序运行的载体是电子计算机,这就需要先了解有关计算机组成等常识,同时也要了解用计算机解题的步骤以及计算机软硬件的简单知识,为以后使用 FORTRAN90 进行程序设计打下基础。

1.1 计算机简介

世界上第一台计算机于 1946 年诞生于美国。在半个世纪的发展过程中,计算机已经历了巨大的变化。按其电子器件的发展来划分,计算机已经换了四代:第一代电子管计算机,第二代晶体管计算机,第三代中小规模集成电路计算机,第四代大规模和超大规模集成电路计算机。近十几年来,人们一直致力于研究第五代计算机。在计算机刚刚诞生的二三十年内,其运算速度每 5~8 年就提高 10 倍,而体积和成本却降低为 1/10。计算机的发展非常迅速,特别是微型计算机问世以来,更加快了计算机的普及范围与速度,也导致了学习计算机浪潮的到来。

1.1.1 计算机的应用与特点

世界上第一台计算机(名叫 ENIAC)最初主要是出于战争的需要,用于计算火炮弹道,属于科学计算范畴。随着计算机硬件和软件的发展,计算机已从单一的科学计算领域,很快发展到国民经济的各个部门乃至家庭。可以说,世界上任何一个领域都已使用或可以使用计算机,计算机的应用范围是十分广阔的。但为了便于了解计算机的用途,可将计算机的应用领域概括成五个主要方面:科学计算、信息处理、实时控制、计算机辅助设计和人工智能。

1. 科学计算

科学计算又称数值计算,早期的计算机主要用于科学计算。例如工程设计、天气预报、地震预测、卫星轨道计算、炮弹弹道计算等等,这些都属于科学计算范畴。1948 年,美国原子能研究中有一项计划,要做 900 万道运算,需要 1500 名工程师计算一年。当时利用了一台初期的计算机,仅用了 150 小时就完成了。这大大地加快了科学研究的速度,使科学家从繁杂的数值计算中解放出来。

另外,由于计算机的出现,导致计算天文学、计算化学、计算生物学、计算医学等学科的产生与发展,使许多古老的学科焕发了青春。

2. 信息处理

信息处理又称数据处理。由于计算机的普及,管理工作也实现了电脑化。人们将用于管理的庞大的程序和数据存入计算机中,管理人员通过终端就可以进行查询、统计、打表等各项管理工作,计算机还可以对数据进行分析 and 汇总,供管理人员分析决策时使用。

人类已经进入了信息时代,其标志之一就是用计算机存储和处理各种信息,运行在计算机上的各种管理软件已在发挥着巨大的作用。例如企业管理系统、情报检索系统、学生成绩

管理系统等等。由于计算机网络系统的发展也十分迅速,这些信息处理软件也可在网络环境中运行,使办公系统实现自动化。

3. 实时控制

实时控制又称过程控制。计算机在机械加工、自动生产线的控制、整个生产过程乃至整个工厂的控制方面有广泛的应用,在化工、石油、造纸、钢铁等企业已广泛地使用了计算机。现代交通控制、高射炮瞄准、飞行器飞行等也都需要使用计算机控制技术。

在生产流程控制中,可将各个环节的测量、调整装置与计算机相连,计算机可以根据事先编好的程序对生产过程进行自动控制。这不但可以提高控制的精度,也可以节省人力。例如某年产 500 万吨钢的钢厂需要职工 15000 人,利用计算机自动化生产后仅需 4000 人。

4. 计算机辅助设计(CAD)

用计算机辅助人们进行设计工作,如设计飞机、房屋、汽车、桥梁等,这可以提高设计速度和节省人力。

以飞机设计为例,人工设计飞机的速度慢,成功率也较低。因为仅仅依靠设计者的经验和头脑很难对多个设计方案进行比较,加上飞机零部件极多,哪一方面出现问题都可能导致设计的失败。利用 CAD 技术,输入一个原始方案后,计算机首先计算出它的性能,然后调整各个设计参数,优化设计方案。经过这样的多次反复过程,选出最佳方案。这显然比人工设计效率高得多。

随着 CAD 技术的发展与应用范围的扩大,又派生出许多新技术分支,如 CAM(计算机辅助制造)、CAI(计算机辅助教学)等。

5. 人工智能

主要研究用计算机来“模仿”人的智能,让计算机具有“推理”和“学习”的能力。例如,人机博弈系统、专家系统、机器人等。

计算机之所以有如上诸多的用途,是因为它具有以下特点:

1. 运算速度快。ENIAC 每秒运算的次数为 5000 次,而巨型机的运算次数已达每秒几十亿次以上。普通计算机也达每秒几百万次。这种运算速度是传统的计算工具无法相比的。

2. 精确度高。一般计算机能提供十几位以上的有效数字。

3. 具有“记忆”和逻辑判断能力。它能把计算步骤、原始数据、运算结果存储在计算机内,它还能进行逻辑判断,并根据判断结果自动决定以后执行哪个命令。

4. 执行步骤由程序控制。计算机内部操作能按预先编好的程序步骤执行,不需要人工干预。

综上所述,可以认为电子计算机是一个以高速进行操作、具有内部存储能力、由程序控制操作过程的自动电子装置。电子计算机简称计算机,俗称电脑。

1.1.2 计算机的硬件组成

现在的计算机均属于冯·诺依曼型计算机,它具有两个主要特点。一是采用“存储程序”的工作原理,即在计算机算题之前,预先编好程序存入计算机存储器,计算机自动按这些程序工作。二是采用二进制编码,指令和数据均以二进制形式存储,这种编码方式能简化计算机的结构。

计算机系统由硬件系统和软件系统两部分组成。

计算机硬件是指其机械部分和电子部分,由运算器、控制器、存储器、输入设备和输出设备五大部分组成(见图 1-1,带箭头的实线表示数据流,带箭头的虚线表示控制命令,虚线框中间的是主机),其中输入设备和输出设备合称输入输出设备,因此也可以说计算机由四部分组成。

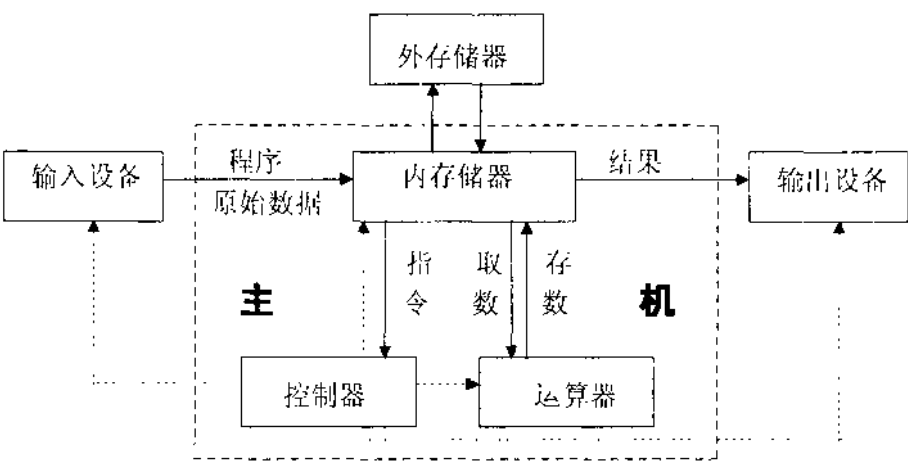
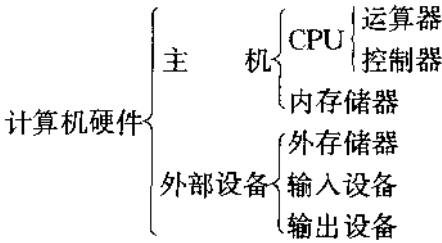


图 1.1 计算机硬件的基本结构

存储器又分成内存器(或称主存储器)和外存储器(或称辅存储器)。运算器和控制器合称中央处理机(简称 CPU),CPU 与内存器又合称主机。外存储器、输入设备、输出设备合称外部设备。因而也可把计算机看作是由主机和外部设备组成：



下边简单介绍计算机的五个部件。

1. 运算器

它主要进行算术运算和逻辑运算。运算器的主要技术指标是机器字长和运算速度。机器字越长,能表示的数的范围也就越大,有效数字的位数就越多。运算速度可用每秒中加法指令运算次数、加法和乘法等指令平均运算次数、主频等多种方法表示。

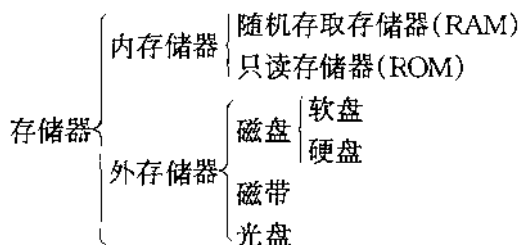
2. 控制器

控制器是分析和执行指令的部件,它是统一指挥和控制计算机各个部件按时序协调操作的中心枢纽。

3. 存储器

它用于存放各种信息。例如从输入设备送入的指令程序和原始数据,从运算器中得到的中间结果等。

存储器有多种分类方法,常见的分类方法如下：



内存储器在主机之内,它与 CPU 之间有信息传递的直接通道,这样内存储器可以随时与 CPU 交换数据。外存储器不能直接与 CPU 交换数据,因为它们之间没有信息传递的直接通道。CPU 若想使用外存储器中的数据,必须利用内存储器作为中间媒介。CPU 与内存储器交换速度快,但内存容量较小;CPU 与外存储器交换速度慢,但外存容量很大。因而只有最常用的信息才放在内存中,其它信息可先放在外存上,需要时必须先换到内存中。

信息存储的最小单位是二进制位,8 个二进制位为 1 字节(Byte,用 B 表示),常用的存储容量单位还有千字节(KB)、兆字节(MB)和千兆字节(GB)等。其中 $1\text{KB}=2^{10}\text{B}=1024\text{B}$, $1\text{MB}=2^{10}\text{KB}=1024\text{KB}$, $1\text{GB}=2^{10}\text{MB}=1024\text{MB}$ 。

4. 输入设备

把信息(程序和数据)送入计算机的部件。常用的输入设备有键盘、鼠标、数字化仪、光笔、摄像机等。

5. 输出设备

将计算机中的信息输出的部件。常用的输出设备有显示器、打印机、绘图仪等。

外存储器属于输入输出型部件,它既可以将信息送入计算机,也可以从计算机中取出信息。常用的外存有磁盘机、磁带机等。

1.1.3 微型计算机

如果将计算机按照价格和运行速度等分型,可分成巨型机、大型机、中型机、小型机和微型机(简称微机)。由于微机价格便宜,又能满足企事业单位的一般需求,其普及面最广。

微型机是由微处理器、存储器、输入/输出接口和系统总线等组成的计算机。由于微机内有許多输入/输出接口,用户可根据需要配置相应的外部设备,再配上适当的软件。就构成了相应的微型计算机系统。

微型机可分成单片机、单板机和个人计算机等几类。目前国内流行的微机是 IBM-PC 系列机,它由 IBM 公司推出。实际上绝大多数用户使用的都是 IBM-PC 系列的兼容机。IBM-PC 机也是由五种部件构成,但根据它们的组装情况,该机硬件可分成主机箱、键盘和显示器三部分。

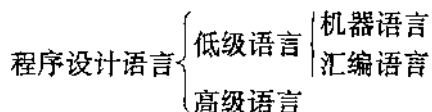
主机箱中除 CPU 和存储器以外,还有配合 CPU 工作的协处理器,在箱中的系统板上还有输入/输出控制部分、总线及扩展槽等。键盘是终端的输入设备,它是由标准的英文打字机键盘演变而来的。显示器有单色和彩色显示器之分,目前一般都使用分辨率较高的彩色显示器。

IBM-PC 机自 80 年代初问世以来发展极快,硬件和软件都在不断地发展中。现在的高档 586 微机无论其运算速度、内外存容量及所携带的外设等都具有相当的规模,软件配备也在日新月异,其适用性很强,在生产实践中使用很广。

1.2 程序设计语言

计算机系统中如果只有硬件还不能被一般用户所使用,必须配有相应的计算机软件。对于科技工作者来说,最常用的软件就是程序设计语言系统。程序设计语言的主要用途是用来编写程序。当人们希望计算机为自己做一些事情时,必须把要完成的这些工作逐项分解,用程序的形式告诉计算机每一步应该怎样去做。因而人们使用程序设计语言的目的是为了产生程序,即产生计算机能识别的操作步骤。

程序设计语言可按如下方式分类:



在上一节中我们说过计算机的控制器是分析和执行指令的部件,而指令就是计算机所能做的操作的最小单位,例如一次加法运算或一次数据传递都可以用一条指令完成。在设计和制造硬件时,就已经决定了该机能识别和执行哪些指令。一台计算机所能识别的一组指令的集合称为该机的指令系统。显然,指令系统中的指令越多,其处理能力越强,用起来越方便。但指令系统太大会使硬件线路复杂化,从而导致计算机价格昂贵。

由于计算机中的信息全用二进制形式表示,其指令和数据也应是二进制形式。早期的计算机仅能识别这种由一条条二进制指令所编写的程序。而这种完全用 0 和 1 二进制代码来书写指令,且这些代码不须翻译就可直接被机器所接受的语言就称机器语言。机器语言书写的程序占用内存空间小,执行速度快,能充分利用硬件资源且可直接被计算机识别。但由于它必须使用绝对地址,程序出错时极难修改,易读性和通用性极差,很难被一般人员所接受。

为了弥补机器语言的弱点,人们用一些比较直观、易记、易写的助记符代替其中的二进制序列,如用 ADD 表示加法指令操作符,用 MOV 表示传递指令操作符,这样机器指令中的操作符、操作数及地址等均用助记符来代替,从而形成了汇编语言。例如 IBM-PC 机 8088 汇编中的一条加法指令可写为:

ADD FIRST, 200H

它表示把存储器中名字为 FIRST 的那个单元之中的数据取出,加上十六进制数 200H(转换成二进制数为 100000000)之后,再存入 FIRST 单元之中。这里的 FIRST 是一个助记符,表示内存中某一个存储单元,可以由用户在使用它之前定义。ADD 也是一个助记符,表示两个数相加的操作符。

汇编语言采用助记符以后,程序的易读性及易修改性比机器语言有了很大的改善,但增加了一件工作:必须将用汇编语言编写的源程序翻译成用机器语言写的二进制代码才能被计算机执行。做这件翻译工作的是汇编程序(见图 1-2)。

无论是机器语言还是汇编语言,它们都具有一个特点:面向机器。所谓面向机器是指其语言与具体的机器有关,换一种机型就不可使用,即其缺乏通用性。例如,我们用 8088 汇编语言编写的程序只能在 IBM-PC 系列机上使用,放到 VAX 机或其它型号的机器上均不能使用。我们把这种与机器有关的语言称作低级语言。机器语言和汇编语言都是低级语言。

由于低级语言与具体的机器关系密切,使用它的人必须了解机器许多内部结构,如寄存器有哪些,它们的名称及功能等,这就使得对计算机内部结构了解不多的人无法有效地使用低级语言。为了解决这一问题,使那些对计算机内部构造并不熟悉的绝大多数人都能编写

计算机程序,人们又发明了高级语言。

高级语言是面向问题计算过程的程序设计语言,它非常接近于人们的习惯,易于人们理解。高级语言源程序不再以指令为组成单位,一般它都是由语句序列组成,而高级语言中的一条语句往往要对应多条指令。

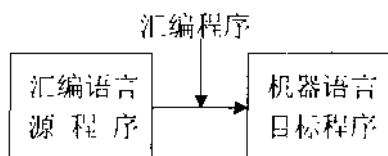


图 1-2 汇编程序工作图

高级语言的主要特点是其独立于具体的计算机,使用方便,通用性强,易于记忆和修改,用户学习高级语言时并不需要学习计算机的内部结构。高级语言的出现,大大地加快了计算机的普及速度,也增加了它的应用范围。

用高级语言编写的源程序必须经过编译(图 1-3)或解释(图 1-4)之后才能被计算机执行。编译程序是把源程序全部编译成用机器语言书写的程序。这种方法的优点是程序执行速度快,一次编译可多次执行。其缺点是源程序修改后必须重新编译,而编译要占据相应的机时。解释程序是逐句逐段地翻译源程序,一边翻译,一边执行。其优点是源程序的修改和扩充较方便,但解释方法工作速度慢,程序被重复执行时也要重复解释。

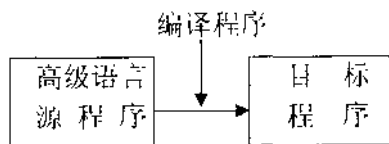


图 1-3 编译程序工作图

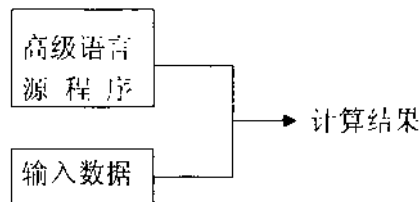


图 1-4 解释程序工作图

国内外使用的高级语言达数百种之多,比较流行的几种如下:

FORTRAN——世界上出现最早的高级语言,主要适用于科学计算。

BASIC——易学易用的语言,适用于初学者。最近几年出现的新版 BASIC 语言可用于科学计算、数据处理及绘图等,其功能大大地增强了。

PASCAL——最早出现的结构化语言,由于其结构严谨、数据类型丰富而最适用于计算机专业教学。

C——近些年最受欢迎的语言之一,由于用它设计出著名的 UNIX 操作系统而闻名。它主要适用于系统软件设计。近些年由于 Borland 等公司将它配上集成环境和引人面向对象的技术而扩大了应用范围。

LISP——是函数式语言,其程序和数据的表示形式是等价的,用递归程序设计作为基础,主要用于人工智能等领域。

PROLOG——以逻辑程序设计为基础,属于描述性语言,主要用于人工智能及专家系统等领域。

其它流行的高级语言还有 COBOL、ADA、ALGOL、PL/1 等。每一种高级语言都在不断地发展着。如 FORTRAN 语言在国内曾流行的版本就有 FORTRAN66、FORTRAN77 等。本书主要介绍 FORTRAN 语言的最新标准——FORTRAN90 的使用方法。

人们使用程序设计语言的主要目的是为了编写出反映人们特定意图的程序,但不能把

程序设计单纯地理解成编程序。实际上,编程序只是程序设计若干步骤中的一步,我们为了编写出合乎要求的程序,要经过分析问题、建立数学模型、设计算法和画出流程图、编制程序、上机调试、维护等多个阶段。

分析问题是对所求问题进行分析,弄清什么是已知条件,要解决什么问题,最后得到什么样的结果等。为了解答未知问题,我们需要对它进行数学表示,从而找出相应的数学模型。由于程序设计是用计算机解决实际问题,因此我们需要在已知数学模型的基础上,提出相应的计算机化的算法。这种算法可以用自然语言表示,也可以用流程图类的图示描述。完成了以上各步以后,才能根据算法编写出相应的程序,经过上机调试之后得到正确的程序投入使用。程序使用中还需要做相应的维护工作。

在初学程序设计语言阶段,由于例题及习题都比较简单,分析问题和建立数学模型等步骤都非常简单,甚至是一目了然的。人们往往不注重这些步骤,也不习惯用流程图描述算法之后再编程,实际上这样做是得不偿失的。随着要解决的问题的复杂化,如果不进行相应的算法描述,编程时会非常困难。因此,在程序设计中,不能忽视程序设计的各个中间步骤。

随着要解决的问题的增大,软件编程量也会增大,其产生的代码会大幅度地增长。这样,程序中出现错误的可能性也增大。在实际的软件开发中,大程序中的错误很难避免,往往会出现软件开发进度比预计拖延,需要的经费不断上涨等问题,从而导致软件危机。为了解决这种危机,人们提出了开发软件的软件工程法。

软件工程是指导计算机软件开发和维护的工程化方法。它采用工程的概念、原理、技术和方法来开发和维护软件,能把经过时间考验而证明正确的管理技术和当前能够得到的最好的技术方法结合起来。软件工程把软件生存周期分成软件定义、软件开发和软件维护三个时期,而软件定义又分成问题定义、可行性研究和需求分析三个阶段,软件开发又分成一般设计、详细设计、编码和单元测试、综合测试四个阶段。软件工程法是目前开发大型软件的有效方法。

1.3 算法的描述

用计算机解决实际问题的主要工作之一是算法的描述。我们从算法的基本概念出发,引出算法的图示描述法。

1.3.1 算法概述

人们每做一件事情都会有具体的工作步骤与方法,这种具体的工作步骤或解题方法就是算法。也可以说,算法就是为解决某一特定问题而采用的具体工作步骤或方法。由于我们讨论的算法最终要反映成计算机程序,因此这里所说的算法实际上是指计算机算法,即算法中的每步都应能被计算机处理,此外还要充分运用计算机本身的特性。

[例 1-1] 设计 $S = \sum_{i=1}^4 a_i$ 的算法。

算法一:

在内存中设一累加用的存储单元 S。

第 1 步:将存储单元 S 清零。

- 第2步:输入第1个数 a_1 。
- 第3步:把 a_1 加到存储单元 S 中。
- 第4步:输入第2个数 a_2 。
- 第5步:把 a_2 加到存储单元 S 中。
- 第6步:输入第3个数 a_3 。
- 第7步:把 a_3 加到存储单元 S 中。
- 第8步:输入第4个数 a_4 。
- 第9步:把 a_4 加到存储单元 S 中。
- 第10步:把存储单元 S 中的结果输出。

这种算法是正确的,但它并未充分利用计算机解题方法的特点。如果按算法一方式,4个数相加需要10个步骤完成,以后每增加一个加数要增加两个步骤。由此推论:10个数相加需要22步,100个数相加需要202步。当加数增多时,程序长度会不断膨胀。1.1节中曾指出计算机的特点之一是其具有逻辑判断能力,我们可以利用这一特性将算法改写如下:

算法二:

在内存中设一累加用的存储单元 S 和一计数用的单元 I 。

第1步:将累加单元 S 清零。

第2步:将计数单元 I 清零。

第3步:输入一个数 A 。

第4步:把 A 加到累加单元 S 中。

第5步:计数单元 I 的值增加1,即 $I+1 \Rightarrow I$ 。

第6步:若 $I < 4$ 则转去执行第3步,否则,继续执行第7步。

第7步:输出存储单元 S 中的结果。

算法二就是计算机程序设计时常用的算法。假设要做100个数相加时,只需把第6步中的“ $I < 4$ ”改成“ $I < 100$ ”,算法步骤并不增加。即100个数相加的算法步骤与4个数相加的算法步骤是相同的(但其执行次数并不相同)。算法二是一种比较通用的算法,其第6步使用了计算机中具有“逻辑判断后转移”的功能,使其输入数据 A 并把 A 加到 S 中这几步骤重复使用,从而简化了算法的步骤,使算法的描述更具有实用性。

除了算法中各步应能被计算机准确执行这一特点之外,算法中往往还有若干个输入量和若干个输出量,这些输入量就是程序的原始或中间输入数据;这些输出量就是程序的计算结果,它表示问题得到解答结果或问题未得到解答。算法另一个重要特性就是算法的有穷性,即一个算法应在有穷步之内结束。假如把上述算法二中第6步改成:转去执行第3步。这样算法每到第6步就回到第3步,第7步无法执行到,算法将无法终止。这种算法是不符合要求的。

1.3.2 算法图示表示法

以上介绍了用自然语言文字表示算法的方法。实际上算法的最终表示是程序设计语言表示法,即用程序设计语言书写的程序来表示算法,这是我们要做的主要工作。由于程序设计语言是一种形式化语言,比较抽象难懂,其规定又过细,在算法设计中直接编出程序尚有相当的困难。用自然语言表示算法十分适合人们的习惯,但书写起来极为不方便,算法表