

计算机等级考试应考丛书

二级考试

(第二版)

程序设计

胡礼和 主编

• 数据库语言程序设计

• 算法、数据结构

• 结构化程序设计

• 等级考试介绍

华中理工大学出版社

计算机等级考试应考丛书

二级考试

——程序设计

第二版

主编 胡礼和

编者 周行明 戚文正 许 勇

刘行淦 孟金平 汪秀芬

华中理工大学出版社

(鄂)新登字第 10 号

图书在版编目(CIP)数据

计算机等级考试应考丛书:二级考试——程序设计(第二版)/

胡礼和主编—武汉:华中理工大学出版社,1996.3

ISBN 7-5609-0980-9

I. 计…

II. 胡…

III. 电子计算机-技术等级标准-考试-自学参考资料

IV. TP3-42

计算机等级考试应考丛书

二级考试

——程序设计

(第二版)

主编 胡礼和

责任编辑 黄以铭 郑兆昭

华中理工大学出版社出版发行

(武昌喻家山 邮编:430074)

新华书店湖北发行所经销

湖北省安陆市印刷厂印刷

开本:787×1092毫米 1/16 印张:18.5 字数:450 000

1996年3月第2版 1996年3月第4次印刷

印数:18 001-28 000

ISBN 7-5609-0980-9/TP·120

定价:17.00元

(本书若有印装质量问题,请向承印厂调换)

内 容 提 要

本书按全国计算机等级考试二级考试大纲的要求编写,介绍了 True BASIC 语言模块化结构化程序设计、常用算法、基本数据结构、FoxBASE+ 数据库语言程序设计及有关的计算机基础知识、基本操作方法和上机实习、调试程序的方法。内容由浅入深,叙述深入浅出。可作各类学生、在职人员、待业人员参加二级考试的应考教材、程序设计课程教材,也可作广大计算机爱好者的自学读物。

再 版 前 言

(全国计算机等级考试及其配套教材介绍)

经国家教育委员会批准,国家教委考试中心已于1994年下半年起,在全国范围内举办计算机等级考试,用于测试应考人员的计算机应用知识和能力。

等级考试与计算机软件水平考试不同。软件水平考试是对计算机专业人员的考核,而等级考试主要是对广大非计算机专业人员的计算机应用能力定级。等级考试根据使用计算机的不同要求,划分若干等级分别考核,目前考试暂定四个等级,其中三级分为A、B两类,四级考试将与国际接轨。

所有在职职工、待业人员和各类学生都有权参加等级考试。应考者的年龄、职业、学历不限,均可到国家教委考试中心公布的就近考点报名。

考试采取全国统一命题,笔试与上机考试相结合的形式。先笔试,后上机考试。笔试时间为一级90分钟,二级、三级120分钟。上机考试时间为一级30分钟、二级60分钟,三、四级暂不规定。笔试、上机考试成绩均合格者,由国家教委考试中心颁发合格证书。笔证、上机考试成绩均优秀者,在合格证上加盖“优秀”字样。证书全国通用。具体考试办法以届时当地报纸刊登的通知为准。

随着计算机技术在我国各个领域的推广和普及,越来越多的人开始学习计算机知识,许多用人部门已将是否具有一定的计算机应用知识与能力作为考核和录用工作人员的标准之一。然而,目前还没有一项全国统一的、有权威的考核为聘人、用人提供依据。于是等级考试便应运而生了。即将举办的全国计算机等级考试面向社会,服务于正在形成和发展的劳动力市场,为人员择业、人才流动提供其计算机应用知识与能力之证明,以便用人部门录用和考核工作人员时有一个统一、客观、公正的标准。

编者从1984年开始从事计算机普及教育的试验与研究,深深体会到当今举办等级考试的重要意义,并预感到此项考试将推动计算机知识的普及,促进计算机技术的应用,有利社会主义经济建设,将对发展计算机文化作出贡献。早在1992年上海等地试行等级考试之时,编者即开始着手进行调查与研究,为了配合国家教委考试中心将此项考试推广到全国,特组织编写了《计算机等级考试应考丛书》。这套丛书是按照全国计算机等级考试大纲编写的,按等级分册。现已出版《一级考试》、《二级考试》、《三级考试(A)类》和《三级考试(B)类》等书。

《一级考试》的主要内容包括:

1. 计算机基础知识;
2. 微型计算机系统的基本组成;
3. 操作系统;
4. 汉字系统及汉字输入方法和字表处理软件;
5. 数据库应用系统;
6. 计算机安全使用知识。

《二级考试》的内容主要包括:

ASL89.3
AR48/02

1. 计算机的基础知识;
2. 程序设计语言;
3. 基本数据结构和常用算法、流程图的使用;
4. 数据库的基本概念、数据库管理系统的使用。

根据考试大纲的要求,在二级考试中,有关计算机的基础部分除了包括本丛书《一级考试》介绍的内容之外,还包括:

1. 数据类型(整型、实型、双精度型、字符型),数据表示形式和存储形式,定点数、浮点数及其表示形式;

2. 指令及指令系统,指令格式、指令分类及功能;

3. 软件的基本概念,程序、文档,软件的分类及其功能,系统软件、应用软件、支持软件。

对于二级考试的应考者,除了按《二级考试》教材学习之外,还需学习《一级考试》的内容,才能达到二级考试大纲的要求。

按考试大纲的要求,三级考试A类侧重于硬件,B类侧重于软件。

《三级考试(A类)》的内容主要包括:

1. 计算机应用基础知识;
2. 微型机硬件系统组成及工作原理;
3. 汇编语言;
4. 微型机接口技术;
5. 数据结构与算法;
6. 操作系统的基础知识;
7. 计算机在测控领域中的应用。

《三级考试(B类)》的内容主要包括:

1. 计算机应用基础知识;
2. 数据结构、算法以及程序设计的基本方法;
3. 操作系统及软件开发环境;
4. 软件工程方法;
5. 计算机在管理信息、数值计算、辅助设计等应用领域中的应用(仅选学一项)。

等级考试将以下列题型考核:

闭卷笔试的题型有选择题、填空题和程序题三种类型,其中程序题也以选择和填空形式出现。上机操作的题型有操作题、编程题和调试题三种类型,其中调试题按软件调试和硬件调试分别进行。

计算机等级考试的诞生和健康发展需要大家——不论是应考者还是计算机普及教育工作者的关心和支持,可以相信,通过一段时间的实践,她将日趋成熟,并将发挥越来越大的作用。

编者在十余年的计算机普及教育试验中,已撰写30余本普及教育教材和教学参考书,《计算机等级考试应考丛书》是在上述书籍编写和教学试验的基础上,根据全国计算机等级考试的要求编写的(对于考试大纲中未明确规定的内容以☆号作标记,供选学),并组织不同类型的教学单位在使用中加以修改。

为了既便于初学者从头学习,又便于已学者按考试大纲复习和补习,丛书在编排上采取了一定的措施。例如,根据需要将某些内容的例题相对集中,以综合应用的形式出现,供不同程度的读者选学。

由于考试现分为四个等级,每一等级均涉及不同领域内的不同层次的计算机知识和技能,为了方便应考者学习,丛书按考试的等级分册,并采取螺旋上升、循序渐进的方式安排学习内容。例如有关数据库的知识,在各级考试中均涉及,丛书在必要时以逐步加深的形式提纲挈领地复习前一册或前几册中与本册有关的内容,然后与本册新内容一并形成新的知识结构。

本修订本是由部分参与教学试验的教师和原书编者、根据教学中搜集的意见和两届全国计算机等级考试的具体情况修改编写的。欢迎更多的教学单位与我们一起继续进行试验,使之修订得更趋完善。来信可寄武汉市华中师大教科所胡礼和。

编者

修订于 1996 年 2 月

目 录

第一章 计算机的基础知识

第一节 计算机中的数据	(1)	第三节 软件	(6)
一、定点数与浮点数	(1)	一、软件的分类及其功能	(6)
二、数据类型	(4)	二、文档	(8)
第二节 指令及指令系统	(4)	第四节 计算机的主要技术指标	(8)
一、指令的概念	(4)	一、容量	(8)
二、指令格式	(4)	二、字长	(8)
三、指令长度	(5)	三、存取周期	(9)
四、指令系统	(5)	四、运算速度	(9)

第二章 True BASIC 语言初步知识

第一节 True BASIC 语言及其 优点	(10)	二、字符串常数	(13)
第二节 字符集与符号名	(10)	三、简单变量	(13)
一、字符集	(11)	第四节 语句和源程序	(14)
二、符号名(标识符)	(11)	一、语句的组成	(14)
第三节 数据	(12)	二、简单语句和结构语句	(14)
一、数值常数	(12)	三、可执行语句与非执行语句	(15)
二、字符串常数	(13)	四、源程序及其书写格式	(15)

第三章 顺序结构的程序设计

第一节 函数、运算符和表达式	(16)	一、上机实习应具备的条件	(25)
一、函数	(16)	*二、上机实习前的准备	(26)
二、算术运算符及运算顺序	(17)	三、上机操作的步骤	(27)
三、表达式	(18)	第五节 键盘输入语句(INPUT 语句)	(30)
第二节 赋值语句(LET 语句)	(19)	一、键盘输入语句的格式和功能	(30)
一、赋值语句的格式	(19)	二、键盘输入语句的使用	(31)
二、变量的赋值与引用	(19)	三、行输入语句(LINE INPUT 语句)	(32)
三、赋值语句的使用	(19)	四、综合应用	(32)
第三节 显示输出语句(PRINT 语句)	(21)	第六节 读数、置数语句(READ/ DATA 语句)	(34)
一、显示输出语句的格式和功能	(21)	一、读数、置数语句的格式和功能	(34)
二、屏幕显示的规定和区域划分	(21)	二、读数、置数语句的使用	(34)
三、显示格式	(22)	第七节 上机操作方法(2)	(35)
四、显示输出语句的使用	(23)	一、常用编辑键及其功能	(35)
五、用 tab 定位	(23)	二、功能键及其功能	(36)
六、综合应用	(24)		
第四节 上机操作方法(1)	(25)		

三、程序的编辑	(36)	第十节 自选格式输出语句(PRINT USING 语句)	(43)
四、汉字输入输出	(37)	一、自选格式输出语句的格式和功能	(43)
五、应用举例	(38)	二、格式字符串的意义	(44)
第八节 恢复数据区语句(RESTORE 语句)	(40)	三、自选格式输出语句的使用	(45)
一、恢复数据区语句的格式和功能	(40)	第十一节 打印输出语句(OPEN #n, PRINT #n 语句)	(46)
二、恢复数据区语句的使用	(41)	一、打印输出语句的格式和功能	(46)
三、三种提供数据语句的比较	(41)	二、打印输出语句的使用	(46)
第九节 单键输入语句(GET KEY 语句)	(42)	第十二节 上机操作方法(3)	(46)
一、单键输入语句的格式和功能	(42)	一、True BASIC 系统命令	(46)
二、单键输入语句的使用	(42)	二、应用举例	(49)

第四章 选择程序设计

第一节 关系表达式和逻辑表达式	(50)	三、结构化程序的三种基本结构	(53)
一、关系运算符	(50)	四、结构化程序设计准则	(54)
二、关系表达式	(50)	五、结构化程序设计方法	(55)
三、逻辑运算符	(50)	六、结构化流程图(N-S 图)	(55)
四、逻辑表达式	(51)	第三节 条件语句(IF 语句)	(56)
五、逻辑值及有关规定	(51)	一、IF-THEN 结构	(56)
第二节 流程图和结构化程序设计	(52)	二、IF-THEN-ELSE 结构	(57)
一、流程图	(52)	三、IF-THEN-ELSE IF 结构	(59)
二、程序设计的基本步骤	(53)	四、多分支选择语句(CASE 语句)	(61)

第五章 循环程序设计

第一节 计数型循环	(66)	二、“直至”型循环	(77)
一、FOR/NEXT 语句	(66)	三、条件型循环的各种格式	(77)
二、FOR 循环的出口语句(EXIT FOR 语句)	(68)	四、条件循环的出口语句(EXIT DO 语句)	(79)
三、多重循环	(69)	第三节 无限循环	(81)
四、综合应用	(72)	一、无限循环结构语句	(81)
第二节 条件型循环	(75)	二、带 EXIT 的无限循环	(82)
一、“当”型循环	(76)	三、综合应用	(83)

第六章 数组

第一节 单下标变量和一维数组	(88)	二、数组定义语句的格式	(90)
一、数组的引入和定义	(88)	三、数据表的处理	(91)
二、数组定义语句的用途	(89)	四、应用举例	(91)
三、应用举例	(89)	第三节 多维数组及其有关概念	(94)
第二节 双下标变量和二维数组	(90)	一、数组的存储	(94)
一、二维数组的引入和定义	(90)	二、数组的定义	(95)

三、数组元素的引用	(96)	一、数组的显示输出语句(MAT PRINT 语句)	(101)
第四节 数组的赋值和运算	(97)	二、数组的键盘输入语句(MAT INPUT 语句)	(102)
一、用常数、变量或表达式赋值的 MAT 语句	(97)	三、字符串数组的键盘输入语句(MAT LINE INPUT 语句)	(103)
二、用数组赋值的 MAT 语句	(98)	四、数组的读数、置数语句(MAT READ/ DATA 语句)	(104)
三、数组加减的 MAT 语句	(98)	五、综合应用	(104)
四、数组乘常数或变量的 MAT 语句	(99)		
五、数组乘数组的 MAT 语句	(99)		
六、有关数组的函数	(100)		
第五节 数组的输入输出语句	(101)		

第七章 字符串

第一节 字符串的概念	(109)	第四节 字符串函数	(114)
一、字符串常量和字符串变量	(109)	一、字符串长度函数	(114)
二、字符串数组	(109)	二、子字符串查询函数	(115)
三、使用字符串的注意事项	(109)	三、字符大小写转换函数	(115)
第二节 提供字符数据的语句	(110)	四、数字字符串与数值数据的转换函数	(116)
一、LET 语句	(110)	五、字符与 ASCII 码转换函数	(116)
二、INPUT 语句	(110)	六、字符串重复函数	(117)
三、READ/DATA 语句	(112)	七、删除前导空格函数	(117)
第三节 字符串的运算	(112)	八、删除后续空格函数	(117)
一、字符串的合并	(112)	九、删除首尾空格函数	(117)
二、取子字符串	(112)	十、系统内部字符串函数综合介绍	(117)
三、字符串比较	(113)		

第八章 子程序和自定义函数

第一节 子程序概述	(120)	第三节 函数的定义与调用	(122)
一、子程序的种类	(120)	一、自定义函数的格式	(123)
二、子程序的特性	(120)	二、自定义函数的定义说明	(123)
三、子程序的执行	(120)	三、自定义函数的调用	(124)
四、引入子程序的优点	(120)	四、应用举例	(124)
第二节 子程序的定义与调用	(121)	第四节 辅程序	(125)
一、子程序的定义格式	(121)	一、辅程序的概念	(125)
二、子程序的调用	(121)	二、子程序与自定义函数的比较	(125)
三、应用举例	(121)	三、模块化程序设计简介	(126)

第九章 文 件

第一节 文件的基本概念	(127)	五、文件指针	(128)
一、文件的定义	(127)	六、文件访问模式	(128)
二、文件的作用	(127)	七、文件建立模式	(129)
三、文件的种类	(127)	八、文件的组织模式	(129)
四、通道	(128)	九、文件的长度	(129)
		十、文件的通用语句	(129)

十一、文件命名	(130)	六、顺序文件读出语句	(134)
第二节 程序文件的操作	(130)	七、文件结束测试语句	(134)
一、文件存入命令	(130)	八、删除文件内容语句	(134)
二、文件读取命令	(130)	九、顺序文件的操作步骤和组织数据的 方法	(134)
三、同名文件存储命令	(131)	十、应用举例	(134)
四、删除文件命令	(131)	第四节 随机文件	(136)
五、文件名列显示命令	(131)	一、随机文件操作步骤	(136)
六、清除内存命令	(131)	二、打开文件语句	(136)
七、文件拼接命令	(131)	三、属性设置语句	(136)
第三节 顺序文件及其操作	(132)	四、属性查询语句	(137)
一、打开文件语句	(132)	五、随机文件写入语句	(137)
二、属性设置语句	(132)	六、随机文件读出语句	(138)
三、属性查询语句	(133)	七、应用举例	(138)
四、顺序文件写入语句	(133)	八、多个数据项随机文件的处理	(139)
五、文件关闭语句	(134)		

第十章 图 形

第一节 图形模式和开设窗口	(142)	第五节 BOX 语句	(150)
一、显示器工作方式	(142)	一、画矩形框语句	(150)
二、图形模式	(142)	二、画矩形面语句	(150)
三、设置图形模式语句	(142)	三、画椭圆(或圆)语句	(152)
四、开设窗口语句	(143)	四、FLOOD 语句	(152)
第二节 图形坐标	(143)	五、动画	(153)
一、图形坐标设置语句	(144)	六、图形中的字符显示语句	(154)
二、询问图形坐标语句	(145)	第六节 上机操作方法(4)	(155)
第三节 基本画图语句	(145)	一、在解释执行过程中查错	(155)
一、画点语句	(145)	二、在编译执行过程中查错	(156)
二、画线语句	(146)	三、设置断点调试程序	(156)
三、画面语句	(147)	四、使用跟踪命令调试程序	(157)
第四节 图形颜色	(148)	五、使用暂停语句检查变量的值	(157)
一、前景颜色的设置与询问	(148)	六、出错处理函数及其功能	(158)
二、背景颜色的设置与询问	(149)	七、出错处理结构	(158)
三、清屏语句	(149)		

第十一章 程序设计的常用算法

第一节 排序与检索	(165)	三、递归	(181)
一、排序	(165)	★第三节 搜索与回溯	(185)
二、检索	(174)	一、搜索与回溯算法在 True BASIC 语言中的 实现	(185)
第二节 迭代、递推与递归	(179)	二、搜索与回溯算法的应用	(188)
一、迭代	(179)		
二、递推	(180)		

第十二章 基本数据结构

第一节 数据结构概述	(193)	三、队列的基本操作	(197)
一、数据结构的定义	(193)	四、应用举例	(197)
二、数据结构研究的对象	(193)	第四节 堆栈	(199)
三、数据结构的作用	(193)	一、堆栈的定义	(199)
第二节 线性表	(193)	二、堆栈的表示方法	(199)
一、线性表的定义	(193)	三、堆栈的基本操作	(199)
二、线性表的表示方法	(194)	四、应用举例	(199)
三、线性表的基本操作	(194)	第五节 链表	(201)
四、应用举例	(194)	一、链表的定义	(201)
第三节 队列	(196)	二、链表的表示方法	(201)
一、队列的定义	(196)	三、链表的基本操作	(202)
二、队列的表示方法	(196)		

第十三章 FoxBASE+数据库语言和程序设计

第一节 数据管理的基本知识	(203)	五、程序的循环结构	(235)
一、数据处理步骤	(203)	六、程序的过程调用	(238)
二、数据库与数据库系统	(203)	第五节 FoxBASE+中的数组及菜单	(244)
三、数据管理技术的发展	(204)	一、FoxBASE+中的数组定义	(244)
四、数据库系统的特点	(204)	二、FoxBASE+中的光条式菜单	(246)
五、数据的组成层次	(205)	三、FoxBASE+中的上拉式菜单	(247)
六、数据模型	(205)	四、FoxBASE+中的下拉式菜单	(248)
七、FoxBASE+简介	(205)	第六节 用CCED对FoxBASE+数据进行报表输出	(251)
第二节 汉字FoxBASE+数据库系统	(206)	一、输出报表中的三个文件	(252)
一、FoxBASE+的优缺点	(206)	二、调用FoxBASE+数据产生多层报表	(255)
二、FoxBASE+中的命令	(206)	三、多维报表	(256)
三、汉字FoxBASE+中的文件	(208)	第七节 FoxBASE+应用程序的编写	(258)
四、汉字FoxBASE+中的变量	(209)	一、主控模块(RSGL.PRG)	(260)
五、FoxBASE+中的运算符和表达式	(210)	二、录入模块(RS10.PRG)	(261)
六、汉字FoxBASE+中的函数	(210)	三、删除模块(RS21.PRG)	(264)
七、FoxBASE+的运行环境、启动与退出	(214)	四、插入模块(RS22.PRG)	(265)
第三节 数据库的建立与操作	(216)	五、查询模块(RS30.PRG)	(266)
一、数据库的建立	(216)	六、打印模块(RS40.PRG)	(267)
二、数据库的基本操作	(217)	七、退出模块(RS50.PRG)	(267)
三、数据库文件的操作	(222)	附录一 True BASIC 常用语句表	(269)
第四节 汉字FoxBASE+编程	(229)	附录二 True BASIC 出错信息表	(274)
一、命令文件的建立、修改与运行	(230)	附录三 FoxBASE+出错信息表	(279)
二、交互式命令	(230)		
三、程序的顺序结构	(232)		
四、程序的选择结构	(232)		

第一章 计算机的基础知识

按全国计算机等级考试一级考试大纲的要求,在《一级考试》一书中介绍了计算机的部分基础知识。按二级考试大纲的要求,二级考试应考者除了需要掌握《一级考试》中介绍的基础知识之外,还应补充学习如下知识。

第一节 计算机中的数据

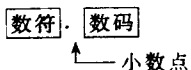
一、定点数与浮点数

在计算机中采用的数制是二进制。为了正确地表示一个数值,除应准确地给出数字外,还要清楚地标出小数点的位置,给出正、负号。

计算机中表示带小数点的数有两种方法,一种是定点表示法,小数点在数中的位置固定不变;一种是浮点表示法,小数点在数中的位置浮动可变。

1. 定点表示法

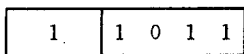
如果在数的表示中,小数点的位置均是固定的,这种表示法称为定点表示法。采用定点表示法的计算机称为定点计算机,简称定点机。原则上说来,在定点表示中,小数点位置可固定在任一位。但为了方便起见,一般都把小数点固定在数的最高位之前,使计算机用纯小数进行运算。因此,定点数的形式为



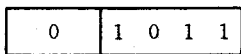
这种定点机要求所有参加运算的数的绝对值都小于1,而且在计算机计算过程中也不应出现大于或等于1的情况。因此,在使用定点机时,必须把数据按一定的比例缩小后才能送入计算机。这一工作,称为选择比例因子。计算所得的结果,按相应比例增大之后才是实际的结果。如果在计算过程中出现大于或等于1的情况,称为溢出或超载。凡出现溢出时,计算机将自动停机或中断(中断是指计算机暂停原来的工作而转去做更紧急的事情),以便对紧急的情况或出现的错误进行处理。

在计算机中,为了区分数的正负,每个数均用前面的一位作为符号位,通常用“0”表示正号;用“1”表示负号。

例如, -0.1011 与 $+0.1011$ 这两个数分别表示为



负数符号 尾数



正数符号 尾数

其中,数的数值部分称为尾数。具有 n 位尾数的二进制数在定点机中表示为



符号位

- n 位尾数

n 位尾数的定点机所能表示的最大二进制数为

$$0.\underbrace{11 \dots 11}_{n \uparrow 1}$$

$$n \uparrow 1$$

即 $1-2^{-n}$ 。其绝对值比它大的数, 计算机无法表示。

n 位尾数的定点机所能表示的最小正数为

$$\underbrace{0.00 \dots 01}_{n \uparrow 0}$$

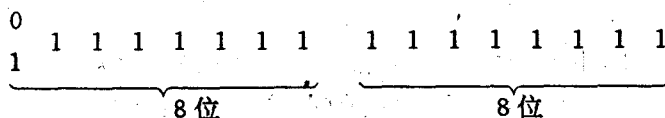
 $n \uparrow 0$

即 2^{-n} 。其绝对值比它小的数, 计算机只能表示成零。这种零称为机器零。定点机的数 N 的绝对值 $|N|$ 以 2^{-n} 为单位由零递增。

由此可见, n 位尾数的定点机所能表示的数 N 的范围是

$$|N| \leq 1 - 2^{-n}$$

实际上计算机中并不能表示出小数点来,必须人为地作出规定,因此定点表示法也可以用来表示整数(小数点在有效数字的后面),它能表示的数值数据的范围取决于计算机中一个数的长度(即位数)。例如,若计算机中一个数值数据的长度为16位,则用它来表达定点整数时,所能表示的最大二进制整数(包括符号位在内)是:



8 位

8 位

即它所能表示的整数 N 的范围是

$$|N| \leq 2^{15} - 1$$

2. 浮点表示法

如果在数的表示中,小数点的位置是可以变动的,这种表示法称为浮点表示法。采用浮点表示法的计算机称为浮点计算机,简称浮点机。

一个十进制数 86.4 可表示为

$$86.4 = 10^2 \times 0.864$$

类似的,对于二进制数 10.11 可表示为

$$10.11 = 2^{10} \times 0.1011$$

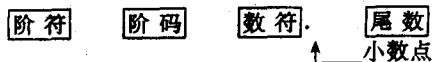
这里指数“10”为二进制数,即十进制数 2;纯小数“0.1011”也为二进制数,即十进制数 $\frac{11}{16}$ 。

一般地说,一个二进制数 N 可表示为

$$N = 2^J \cdot S$$

其中J是一个二进制整数,称为数N的阶码(简称阶);S是一个二进制纯小数,称为数N的尾数或数码。

因此,浮点数分为阶码和尾数两个部分,均有各自的符号位,分别称为“阶符”和“数符”,其形式为



小数点

阶码和尾数的正负号也分别用“0”和“1”表示。

例如,二进制数 0.0111 的浮点形式为

$$2^{-1} \times 0.1110$$

这里二进制数“-1”是阶码;二进制数“0.1110”是尾数。在浮点机中这个数表示为

1	0 1	0	1 1 1 0
阶符	阶码	数符	尾数

又如二进制数-111.1000 的浮点形式为

$$2^{11} \times (-0.1111)$$

这里二进制数“11”为阶码;二进制数“-0.1111”为尾数。在浮点机中这个数表示为

0	1 1	1	1 1 1 1
---	-----	---	---------

这里阶符占 1 位,阶码本身占 2 位;数符占 1 位,尾数本身占 4 位。

一般地说,具有 m 位阶码 J 和 n 位尾数 S 的浮点数 $N = 2^J \cdot S$, 在浮点机中表示为

J_1	J	S_1	S
阶符	阶(m 位)	数符	尾数(n 位)

此时,浮点机所能表示的数 N 的范围为

$$|N| \leq 2^{2^m-1} \times (1-2^{-n})$$

对给定的二进制数来说,其浮点形式不是唯一的。例如对二进制数 0.0111 来说,由于

$$0.0111 = 0.0111 \times 2^0 = 0.1110 \times 2^{-1}$$

因此可以写出二进制数 0.0111 的两种浮点形式。由这个例子可以看到,当尾数的小数点位置改变时,只要阶码也相应地改变,则可以保证整个数值不变。

为了保证数的精确度,在浮点机中一般采用规格化的形式。

如果二进制数的尾数 S 的最左一位数字为“1”,则称此数为规格化形式的数。

如果二进制数的尾数 S 的最左一位数字为“0”,则称此数为非规格化形式的数。

例如,对二进制数 0.0111 来说, 0.0111×2^0 为非规格化形式的数,而 0.1110×2^{-1} 为规格化形式的数。

利用增加或减少阶码数值的方法,使尾数部分最左一位数是 1 叫做数的规格化。这样的表示方法能提高数的有效数位。

例如,非规格化形式的数 $+0.001101 \times 2^0$,规格化后应写为 $+0.110100 \times 2^{-10}$,这就是规格化形式的数。计算机实现数的规格化时,只要将尾数左移两位,同时阶码减 2 即可。

阶码的大小能够确切地代表小数点的位置。所以,也可以说,小数点的位置是随数的阶码的大小而“浮动”的。阶码的正、负号只决定小数点的位置是左移还是右移;尾数的符号决定该数的正、负值。

就一台计算机而言,字节的位数是固定的,若采用定点表示法,则所能表示的数值范围较小,可是有效数字的位数较多;若采用浮点表示法则所能表示的数值范围较大,但有效数字的位数较少,运算速度较慢,机器比定点机复杂,所用器件较多。

一台计算机究竟采用定点表示法还是浮点表示法,由客观实际对计算机的具体要求来决定。目前,较高档次的机器一般都具备定点与浮点两种表示法,由使用者自由选择。较低档次的机器通常只有定点表示法。

数的定点表示方法和浮点表示方法都是取其一位机器码表示数值的符号,其他位机器码表示其二进制数的绝对值,这种方式称为原码编码方式。在实际使用时还有补码和反码等其他几种编码方式。采用补码和反码可以将减法化为加法来做,而乘法和除法又可通过累加和累减来实现,所以,只要有一个加法器就可以实现加、减、乘、除四则运算。

二、数据类型

多数程序设计语言中都有整数型、实数型、逻辑型、字符型这四种数据类型。

(1)整数型数据的取值范围取决于给定机器所能表示的最小最大整数值,可以进行算术运算。

(2)实数型数据的取值决定于给定机器的浮点数的表示范围,可以进行算术运算。

(3)逻辑型数据只有 TRUE(真)和 FALSE(假)两个逻辑值,它们可以进行 AND,OR 和 NOT 三种逻辑运算。

(4)字符型数据取值来源于系统提供的 ASCII 码字符集。

练习 1-1

1. 什么叫做数的定点表示法和浮点表示法,各有什么优缺点?
2. 二进制数 -0.0101 在定点机和浮点机中分别如何表示?
3. 数据可分为哪几种类型?

第二节 指令及指令系统

一、指令的概念

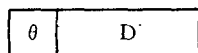
指令是指指挥计算机工作的指示和命令。人们用指令表达自己的意图,下达给控制器,控制器靠指令指挥整个计算机工作。一系列的指令按规定集合,即组成程序。计算机的整个工作过程实质上是执行程序的过程。

二、指令格式

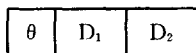
不同的机器可能有不同的指令内容和格式。一条指令通常包括两个方面的内容:一是指出机器操作的要求,二是给出操作数在存储器中的地址。

在计算机中,指令是用一组数码表示的,这组数字形式的代码称为指令码。指令码由操作码和地址码两个基本部分组成。

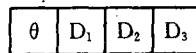
指令的操作码部分规定计算机应执行的操作性质;地址码部分提供一个或几个操作数的地址。只提供一个操作数地址的指令叫单地址指令或单操作数指令;提供两个操作数地址的指令叫双地址指令或双操作数指令。还有三地址指令及其他形式的指令。它们的基本格式如下:



单地址指令格式



双地址指令格式



三地址指令格式

其中:0 为操作码。

D, D_1, D_2, D_3 为地址码。

操作码一般位于指令的前部,并由一定位数的二进制数码组成。每一种操作(如算术的加、减、乘、除运算和逻辑加、逻辑乘以及比较、传送等)都要用不同的二进制代码表示。因此,操作码占用的位数和计算机所能执行的指令数目有密切的关系。如果操作码占用 6 位,则 6 位二进制数可有 64 种不同的组合,它最多能表示 64 种操作。编写程序时,操作码一般不用二进制而是用八进制或十六进制书写。对于同一种操作,不同类型的计算机所用的操作码是各不相同的。

在计算机中使用了能存放大量信息的存储器和寄存器。为了识别各个存储单元,便于存取信息,给这些存储单元进行了编号,这个编号就是各个存储单元的地址。用代码表示的地址叫地址码。指令中的地址码部分指出的地址如果是参加运算的操作数的地址,那么这个地址叫操作数地址;地址码指出的地址如果是运算结果的地址,那么这个地址就叫结果地址。

三、指令长度

代表一条指令的二进制位数叫指令长度。

在计算机中,用 8 位二进制代码表示一个字节,如果用两个字节来表示一条指令,则这条指令的长度就为 16 位。

随着计算机内存容量的不断扩大,指令长度也随之而增加。图 1-2-1 给出了不同指令长度的形式。

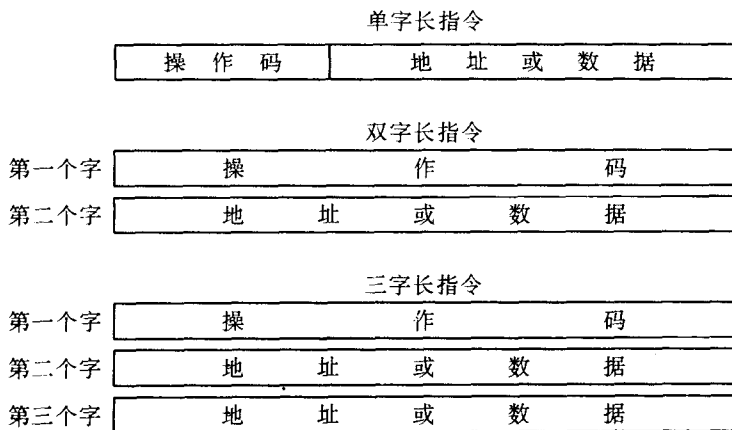


图 1-2-1 不同指令长度的形式

四、指令系统

一台计算机所能执行的各种不同类型指令的总和称为该机器的指令系统,又叫指令集。不同机器的指令系统所包含的指令种类和数目是不同的。以它们的功能来划分,大致有以下几类:

1. 算术运算指令

常用的有定点或浮点加、减、乘、除等指令。

2. 逻辑运算指令

常用的有逻辑加法、逻辑乘法、模 2 加法(异或函数)等指令。