

中/等/职/业/技/术/学/校/计/算/机/专/业/系/列/教/材

数据结构 初步

李晓燕 等编



中国财政经济出版社

中等职业技术学校计算机专业系列教材

胡金柱 李邦畿 李晓燕 杨发明 主编

数据结构初步

李晓燕 吴宇红 李 斌 编

中国财政经济出版社

内 容 提 要

本书比较系统地介绍了数据结构的有关基本概念和基础知识。全书共十章。第一章绪论,介绍了“数据结构”所研究的对象和内容,因为本书中的算法的描述语言采用C语言,(一般教材均采用类pascal语言)故对C语言作了简单的回顾。第二章至第七章分别介绍了线性表、栈、队列、串、数组、树、二叉树和图等数据结构的逻辑结构和存贮结构,以及各种基本运算。第八章和第九章介绍了查找与分类。第十章主要介绍了文件的概念。各章均附有习题。

图书在版编目(CIP)数据

数据结构初步/李晓燕等编. -北京:中国财政经济出版社,1996
中等职业技术学校计算机专业系列教材
ISBN 7-5005-3156-7

I. 数… II. 李… III. 数据结构-专业学校-教材 IV. TP311.12

中国版本图书馆 CIP 数据核字(96)第 11345 号

中国财政经济出版社出版

社址:北京东城大佛寺东街 8 号 邮编:100010

湖北《新生报》印刷厂印刷 各地新华书店经销

787×1092 毫米 16 开 10.5 印张 268 800 字

1996 年 6 月第 1 版 1996 年 6 月武汉第 1 次印刷

印数:1-15 100 定价:12.00 元

ISBN 7-5005-3156-7/TP·0010

(图书如出现问题,本社负责调换)

中等职业技术学校计算机专业系列教材

编写及使用说明

为适应我国当前中等职业技术学校计算机专业教学的需要,我们组织编写了一套实用教材——中等职业技术学校计算机专业系列教材。本教材针对中专学生的层次水平(初中毕业生)及初学计算机的特点,以及计算机专业教学的需要,我们对整套书中的内容作了周密的安排,不仅体系合理,而且深入浅出、概念清晰、例题丰富、逻辑性强、文字流畅、通俗易懂。

全套书共分 11 册,它们分别是:

1. 微型计算机组成原理; 2. 微型计算机操作; 3. BASIC 语言程序设计; 4. C 语言程序设计; 5. 数据结构初步; 6. FOXBASE⁺关系数据库; 7. 关系数据库 FOXPRO; 8. 微型计算机系统的安装与维护; 9. 计算机网络基础; 10. 计算机软件开发技术; 11. Windows 基础教程。

每册教材的详细情况请见各册的前言。

本教材建议安排在 3—5 个学期讲授完,每学期可安排 2—3 门课程。教学顺序可按上述排列顺序进行,但应根据各学校学生具体情况和上机实习的条件酌情安排。每本书自成体系,可独立使用。这些课程实践性较强,上机实习必不可少。每门课程的讲授与实习学时数安排建议如下:

1. 《微型计算机组成原理》:可安排 70—90 学时,其中讲授 50—60 学时,汇编语言的上机可安排 20—30 学时。

2. 《微型计算机操作》:本课程以上机实习为主,课堂讲授为辅。总学时可安排 70—90 学时,讲授 30 学时左右,上机实习 40 学时以上。

3. 《BASIC 语言程序设计》:总学时 70—90 学时,课堂讲授占 2/3,上机实习占 1/3 左右。

4. 《C 语言程序设计》:总学时 70—90 学时,课堂讲授占 2/3,上机实习占 1/3 左右。

5. 《数据结构初步》:总学时 70—90 学时,讲授 50—70 学时,上机实习 20 学时左右。

6. 《FOXBASE⁺关系数据库》:总学时 70—90 学时,课堂讲授占 2/3,上机实习占 1/3。

7. 《关系数据库 FOXPRO》:总学时 70—90 学时左右,课堂讲授占 2/3,上机实习占 1/3。

8. 《微型计算机系统的安装与维护》:总学时 80—100 学时,课堂讲授占总学时的 3/4,实习可占总学时的 1/4(有条件的尽可能安排实习,没有条件的可演示)。

9. 《计算机网络基础》:总学时 70—90 学时左右,讲授占 3/4,实习占 1/4,没有实习条件的可安排参观、演示。

10. 《计算机软件开发技术》:总学时 70 学时左右,讲授 50 学时左右,寻找小课题按软件开发期进行实习,时数可酌情安排。

11. 《Windows 基础教程》:总学时 70—90 学时,讲授与上机各占一半。

另外:①书中凡打星号(*)的章节可作选讲内容,不作要求;②FOXBASE⁺与 FOXPRO 的许多命令相同,这两门课程可酌情选其中一种开设。对于学生来说,学会其中一种,另一种则很容易掌握(另一种可作为学生自学的教材);③五笔字型与自然码不要求都学,可选其中一种教学;④Windows 与网络有条件的学校应尽量开设,没有上机实习条件的学校可暂时不开。

前 言

数值计算是计算机应用的一个重要方面,而非数值数据的处理则是计算机应用更广泛、更重要的领域。《数据结构》研究的对象正是非数值数据的逻辑结构和存贮结构,以及能施加的各种基本运算。《数据结构》为非数值数据的研究提供了基础理论和方法。因此,《数据结构》是计算机科学中一个十分重要的专业基础课程。

我们编写的《数据结构初步》是一本面向中等专业学校学生的关于数据结构基础知识和基本理论的教材。在教材中,作者注意贯彻深入浅出、循序渐进的原则,并且,在描述算法时,特别注意图文并茂,以增加算法的易读性。

在其它的《数据结构》教材中,算法的描述语言一般采用一种类 Pascal 语言。但是,在《数据结构初步》这本教材中,算法的描述语言直接采用 C 语言,每一个算法就是一个可运行的 C 语言程序。而 C 语言的诸多特点则是众所周知的。

在描述算法的 C 语言程序中,结构、结构指针和结构数组等是经常用到的几个数据类型,读者务必熟悉这几个数据类型的定义和使用。list.h, linkl.h 和 string.h 是几个头文件,在这几个头文件中包含了几种结构类型定义和函数定义。

本教材共十章。第一章绪论,介绍了《数据结构》所研究的对象和内容,并对 C 语言作了简单的回顾。第二章至第七章,分别介绍了线性表、栈、队列、串、数组、树、二叉树和图等数据结构的逻辑结构和存贮结构,以及各种基本运算。第八章和第九章,介绍了查找与分类。第十章,主要介绍了文件的概念。

本书稿承蒙胡金柱副教授认真审阅,在出版过程中得到了有关单位和同志的热情支持和无私帮助,我们在此深表谢意。由于水平有限,不妥之处敬请各位批评指正。

作 者

1996 年 2 月 24 日

目 录

第一章 绪论	(1)
§ 1.1 数据结构研究的对象和内容	(1)
§ 1.2 基本概念	(1)
§ 1.3 算法的描述	(3)
1.3.1 简单程序举例	(3)
1.3.2 数据类型	(3)
1.3.3 变量	(5)
1.3.4 运算符	(5)
1.3.5 输入输出函数	(6)
1.3.6 语句	(6)
习题一	(8)
第二章 线性表	(9)
§ 2.1 线性表的逻辑结构	(9)
§ 2.2 线性表的物理结构——顺序存贮结构	(10)
2.2.1 顺序存贮方式	(10)
2.2.2 线性表在顺序存贮方式下的基本运算	(13)
§ 2.3 线性表的另一种物理结构——链式存贮结构	(17)
2.3.1 单链表	(18)
2.3.2 单链表的基本运算	(21)
2.3.3 多项式加法	(28)
§ 2.4 双向链表	(34)
§ 2.5 循环链表	(36)
习题二	(37)
第三章 栈与队列	(38)
§ 3.1 栈的定义和顺序存贮结构	(38)
3.1.1 栈的定义	(38)
3.1.2 栈的顺序存贮结构	(38)
§ 3.2 栈的基本运算	(39)
3.2.1 进栈运算	(39)
3.2.2 退栈运算	(39)
3.2.3 读栈顶元素运算	(40)
3.2.4 置空运算	(40)
3.2.5 测空运算	(40)
§ 3.3 算术表达式求值	(43)

§ 3.4 栈的另一个重要应用——递归函数	(46)
3.4.1 递归函数	(46)
3.4.2 Hanoi 塔	(46)
§ 3.5 队列的定义	(48)
§ 3.6 队列的存贮结构	(48)
3.6.1 队列的顺序存贮结构	(49)
3.6.2 队列的链式存贮结构	(49)
3.6.3 链式队列基本运算	(49)
习题三	(52)
第四章 数组和稀疏矩阵	(54)
§ 4.1 数组的定义	(54)
§ 4.2 数组的存贮结构	(54)
§ 4.3 稀疏矩阵	(55)
4.3.1 三元组表	(55)
4.3.2 十字链表	(62)
习题四	(63)
第五章 串	(65)
§ 5.1 串的逻辑结构	(65)
§ 5.2 串的存贮结构	(66)
5.2.1 串的链表	(66)
5.2.2 串的顺序存贮结构	(67)
5.2.3 串的存贮映象	(67)
§ 5.3 串的基本运算	(68)
5.3.1 串的联接	(68)
5.3.2 串的赋值	(70)
5.3.3 测试两个串是否相等	(72)
5.3.4 求子串	(74)
5.3.5 求子串在主串中的位置	(76)
习题五	(79)
第六章 树	(80)
§ 6.1 树的定义	(80)
§ 6.2 二叉树	(81)
6.2.1 二叉树的定义	(81)
6.2.2 二叉树的基本性质	(81)
6.2.3 二叉树的基本运算和存贮结构	(82)
§ 6.3 遍历二叉树	(84)
6.3.1 前序遍历	(84)

6.3.2 中序遍历	(85)
6.3.3 后序遍历	(85)
§ 6.4 二叉树遍历算法的实现	(86)
§ 6.5 线索二叉树	(87)
6.5.1 线索二叉树	(87)
6.5.2 线索二叉树的遍历	(89)
6.5.3 线索二叉树的插入运算	(90)
§ 6.6 树、森林与二叉树的转换	(92)
6.6.1 树转换成二叉树	(92)
6.6.2 森林与二叉树的转换	(93)
§ 6.7 哈夫曼树及其编码	(95)
6.7.1 路径及路径长度	(95)
6.7.2 哈夫曼树	(95)
6.7.3 哈夫曼编码	(97)
习题六	(98)

第七章 图 (99)

§ 7.1 图的定义	(99)
7.1.1 定义	(99)
7.1.2 子图	(100)
7.1.3 连通图和图的连通分量	(100)
7.1.4 度、入度和出度	(100)
§ 7.2 图的存贮结构	(101)
7.2.1 邻接矩阵	(101)
7.2.2 邻接表	(101)
7.2.3 邻接多重表	(102)
§ 7.3 图的遍历	(103)
7.3.1 纵向优先搜索法	(103)
7.3.2 横向优先搜索法	(104)
§ 7.4 生成树	(104)
7.4.1 生成树	(104)
7.4.2 最小生成树	(105)
7.4.3 最小生成树的生成算法	(105)
习题七	(107)

第八章 查找 (108)

§ 8.1 顺序查找	(108)
§ 8.2 折半查找	(110)
§ 8.3 二叉排序树及其查找	(112)
8.3.1 二叉排序树	(112)

8.3.2 二叉排序树的查找	(112)
§ 8.4 哈希查找	(113)
8.4.1 哈希函数	(113)
8.4.2 冲突处理	(115)
习题八	(118)
第九章 分类	(119)
§ 9.1 概述	(119)
§ 9.2 冒泡分类	(120)
§ 9.3 简单选择分类	(123)
§ 9.4 线性插入分类	(125)
§ 9.5 折半插入分类	(128)
§ 9.6 希尔分类	(130)
§ 9.7 快速分类	(133)
§ 9.8 堆分类	(136)
9.8.1 堆和完全二叉树	(136)
9.8.2 堆分类	(137)
9.8.3 堆分类算法	(139)
§ 9.9 归并分类	(141)
9.9.1 分类数据序列的归并	(141)
9.9.2 多个分类数据序列的归并	(141)
9.9.3 归并分类	(142)
§ 9.10 基数分类	(144)
9.10.1 多关键字分类	(144)
9.10.2 基数分类	(144)
§ 9.11 外部分类	(150)
9.11.1 磁带文件的归并分类	(151)
9.11.2 磁盘文件的归并分类	(152)
习题九	(153)
第十章 文件	(154)
§ 10.1 外存贮器	(154)
10.1.1 磁带	(154)
10.1.2 磁盘	(154)
10.1.3 硬盘	(155)
§ 10.2 文件的特性及基本概念	(155)
§ 10.3 顺序文件	(157)
§ 10.4 索引文件	(158)
§ 10.5 随机存取文件	(158)
习题十	(160)
参考文献	(160)

第一章 绪 论

数据结构是计算机科学中一门十分重要的课程。它研究的对象是什么,研究的内容是什么,这是读者在学习数据结构这门课程时首先会提出的问题。本章将就此问题逐一展开讨论。

§ 1.1 数据结构研究的对象和内容

随着计算机科学技术的不断发展,其应用领域更加广泛和深入。早期计算机主要用于纯数值的科学计算,而现代计算机则更多用于非数值信息处理。我们把计算机加工处理的对象,数值的与非数值的信息,统称之为数据。字符、图像、声音等一类数据,称为非数值数据。假如我们打算把一所图书馆的所有图书和资料都交给计算机来管理,使读者能更方便、更迅速地检索、查阅和复制文献与资料,这类问题便属非数值信息处理。

众所周知,计算机对信息(数据)的加工处理是通过计算机程序来实现的,也就是说数据是计算机程序加工处理的对象。程序要提高对数据加工处理的效率和速度,除了要选择高效率的算法之外,还须选择合理的数据结构。

程序加工处理的数据,正是数据结构这门课程所研究的对象,而数据间的关系即结构以及能对它施加的运算,正是数据结构这门课程所研究的内容。

例 1.1 设计一个程序,其任务是编制一个电话号码簿,以方便用户查找。

为了实现这一查找算法,首先必须选择合适的数据结构。在电话号码簿中,我们令一个人名与一个电话号码相对应,即一个人名与他的电话号码构成一个偶对,作为一个数据存入计算机中。如:

(张三,8776524),(李四,7832715),…,(王五,6451903)。

显然,电话号码簿中这些数据在计算机中的排列方式直接影响查找效率。如果这些数据在计算机中排列毫无规律,那么查找就十分不方便。假若这些数据在计算机中按一定规律排列,如按字典顺序排列人名,那么查找起来就比较方便了。

由此可见,数据的组织即数据间的结构,直接影响对数据操作(算法)的效率。

电话号码簿中数据间的这种顺序关系,我们称之为线性关系,电话号码簿称为线性表。除了线性表之外,还存在有其它更复杂的数据结构。

例 1.2 一个中学各年级和班级的设置,如图 1-1 所示。

如果在计算机中存贮这个组织设置,且表达学校与年级之间所存在的关系,那么这些数据之间肯定不是一种线性关系,而是一种比线性关系更复杂的一种关系,我们把这样一类数据结构称为树结构。

§ 1.2 基本概念

本节将集中介绍一些基本概念。这些基本概念在后面章节中将会经常出现。

一、数据

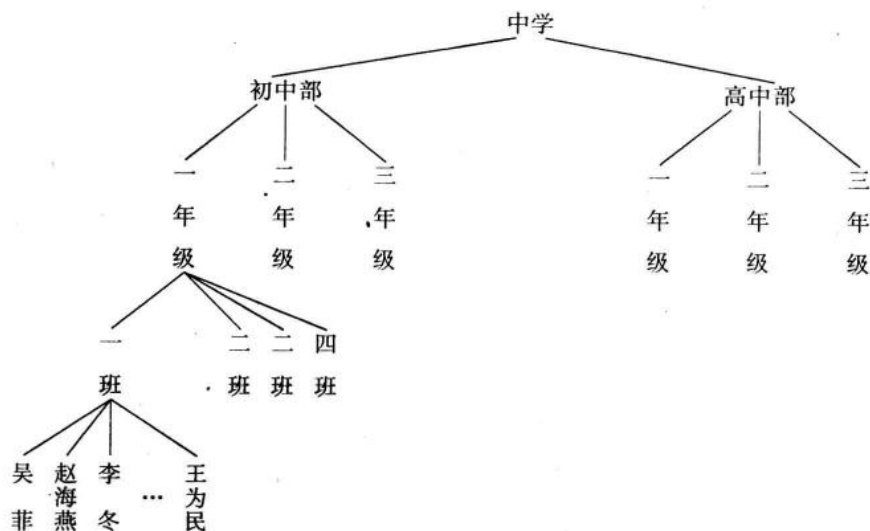


图 1-1

这是一个含义比较广泛的概念。凡是能输入到计算机中被计算机程序加工处理的符号，统称之为数据。例如，数、字符串，还有图形、图像、声音等，都是属于数据这个范畴。总之，数据是计算机程序加工处理的对象的总称。

二、数据元素

我们往往把计算机程序加工处理的一个整体称为数据元素，也就是说数据元素是数据的基本单位。例如，一个数，一个字符串，一个学生的有关信息（学号、姓名、性别、年龄、籍贯、住址等），都可作为数据的基本单位，即都可作为数据元素。从这几个例子可知，一个数据元素可以只包含单项，而有的数据元素可能包含多个项。我们把构成数据元素的各组成部分称为数据项。数据项是数据的具有独立意义的最小单位，也就是说，一个数据元素可能只包含一个数据项，也可能包含多个数据项。数据元素这个概念，在不同的、具体的数据结构中可能有各种称呼。如，在文件中，数据的基本单位称为记录；在树、图数据结构中，数据的基本单位又称为结点或顶点。

三、数据对象

具有相同性质的数据元素之集合，称为数据对象。例如，实数型数据对象是所有实数构成的实数集。

四、数据结构

数据结构是用来描述数据对象中数据元素之间相互关系和存贮方式的一种抽象化表示形式。例如，线性表、栈、队列、二叉树、图等等，这是一些基本的数据结构，在后面的章节中我们将作详细的讲述。

数据结构是指数据元素之间的逻辑结构和物理结构。数据之间的物理结构是指数据元素在计算机中的存贮方式。数据元素之间的逻辑结构是指数据元素之间所存在的一种抽象化的相互关系，它不涉及数据元素在计算机中的存贮形式。

当我们定义一种数据结构时，我们是规定它的逻辑结构。例如，我们定义线性表为 $n(\geq 0)$ 个有限数据元素构成的一个序列，记为

$$(a_1, a_2, \dots, a_n)$$

在这个定义中,我们并没有谈到这些数据元素在存储器中的存储形式,只是规定了这些数据元素之间的一种先后顺序关系。

在后面章节中,我们讨论一个数据结构,不仅要给出它的逻辑结构定义,而且要介绍它的存储结构以及能施加于该数据结构的基本运算。当然,这些基本运算对于具体某一种数据结构来讲应该是封闭的,也就是说,对某一种数据结构施加某一种运算之后,得到的结果应仍然是该数据结构。例如,在一个线性表中插入一个新数据元素,得到的仍然是一个线性表。

基本运算主要包括以下几种:

1. 建立一个数据结构;
2. 从已有的数据结构中删除一个数据元素;
3. 向一个数据结构插入一个新数据元素;
4. 访问数据结构的数据元素;
5. 对一个数据结构进行修改;
6. 对一个数据结构进行排序;
7. 对一个数据结构进行查找。

§ 1.3 算法的描述

算法是执行特定计算的有穷过程。算法中的每条指令都须是确定的、无二义性的。我们在讨论各种数据结构的基本运算时,都将给出相应的算法,而这些算法我们将用 C 程序设计语言来描述。为了便于读者编写和阅读算法,我们在此对 C 程序设计语言作一简单回顾。

1.3.1 简单程序举例

```
main( )
{
    char  a[10],c;
    int   t;
    t=0;
    while((c=getchar( ))!= '\n')
        a[++t]=c;
    while(t!=0)
        putchar(a[t--]);
}
```

执行该程序,输入一行字符:

abcdefgh_✓

显示:

hgfedcba

1.3.2 数据类型

每个变量在使用之前均须定义其数据类型。C 语言有以下几种数据类型:整型、浮点型、字

符型、指针型、无值型、结构和联合。

一、整型

整型数据加上不同的修饰符,有下列几种类型:

signed short int	有符号短整型
signed long int	有符号长整型
unsigned short int	无符号短整型
unsigned long int	无符号长整型

二、浮点型

float	单精度浮点型
double	双精度浮点型

三、字符型

char	有符号字符型
unsigned char	无符号字符型

有符号字符型数取值范围为 $-128\sim 127$,无符号字符型数取值范围则为 $0\sim 255$ 。

一个字符串可用一个字符型数组表示。一个字符串常数一般用双引号括起来表示。一个字符常数可直接用单引号括起来表示,或者用该字符的 ASCII 码值来表示。

四、指针型

指针是指向变量的地址。根据所指向的变量类型的不同,指针型可以是整型指针、浮点型指针、字符型指针、结构指针和联合指针等。

int *	整型指针
float *	浮点型指针
char *	字符型指针

五、无值型

void

其用途有两个:一是明确表示一个函数不返回任何值;二是产生一个同类型的指针。

六、结构

结构的定义格式:

```
struct 结构名
{
    类型 变量名;
    类型 变量名;
    :
} 结构变量;
```

结构成员的表示方式:

结构变量·成员名

在结构的定义格式中,结构变量可以是数组,此时则定义一个结构数组。如果在结构的定义格式中,在结构变量之前加一个星号“*”,则定义一个结构指针。

七、联合

```
union 联合名
{
```



```

    类型  成员名;
    类型  成员名;
    :
} 联合变量名;
联合变量也可以定义成数组或指针。

```

1.3.3 变量

一、变量说明

所有变量在使用之前都必须加以说明。

二、变量种类

1. 局部变量

系函数内部说明的变量。

2. 全程变量

系所有函数之外说明的变量。

三、变量存储类型

```

auto          自动变量
static        静态变量
extern        外部变量
register      寄存器变量

```

四、数组

系具有相同数据类型的变量集合。

数组可以是一维的,也可以是多维的。

一维数组说明格式:

类型 数组名[长度];

多维数组说明格式:

类型 数组名[第 n 维长度][第 n-1 维长度]...[第 1 维长度];

1.3.4 运算符

一、算术运算符

+	(正,加)	%	(取模)
-	(负,减)	--	(减 1)
*	(乘)	++	(加 1)
/	(除)		

二、关系运算符

>	(大于)	<=	(小于或等于)
>=	(大于或等于)	==	(等于)
<	(小于)	!=	(不等于)

三、逻辑运算符

&&(逻辑与),||(逻辑或),!(逻辑非)。

四、位运算符

&	(位逻辑与)		(位逻辑或)
^	(位逻辑异或)	~	(位逻辑反)
>>	(右移)	<<	(左移)

五、条件运算符“?”

“?”为三目运算符,其形式为。

〈表达式 1〉? 〈表达式 2〉:〈表达式 3〉

进行条件运算时,先计算表达式 1 的值,如果值为非 0,则计算表达式 2 的值,且将该值作为整个表达式的值;否则计算表达式 3 的值,且将它作为整个表达式的值。

六、赋值运算符“=”

赋值表达式的形式为:

$$v=e$$

允许多重赋值,按照从右到左的顺序结合。还可以有以下形式:

$$v+=e$$

等价于

$$v=v+e$$

七、单目运算符“&”与“*”

“&”运算符返回操作数的地址。“*”运算符是对“&”运算符的一个补充,它返回位于这个地址内的变量值。

八、“,”运算符

“,”运算符用于将多个表达式串在一起,从左到右逐个计算表达式的值,最右边表达式的值才是整个表达式的值。

1.3.5 输入输出函数

一、读字符函数

getchar()

二、写字符函数

putchar(ch)

三、格式输入函数

scanf(格式参数,地址参数 1,地址参数 2,...)

四、格式输出函数

printf(格式参数,值参数 1,值参数 2,...)

1.3.6 语句

一、条件语句

if(表达式)

语句;

else if(表达式)

语句;

⋮

else

语句;

这种结构是从上到下逐个对表达式进行判断,如果某表达式的值为非 0,则执行与该表达式相对应的语句,否则执行最后一个 else 的语句。该结构中的“else if(表达式)”与“else”语句均为可选项。

二、while 语句

while (表达式)

语句

while 语句表示当表达式的值为非 0 时,便重复执行循环体,直到表达式的值为 0 才终止循环。

三、for 语句

for (表达式 1;表达式 2;表达式 3)

语句

表达式 1 用于给循环变量赋初值;表达式 2 是一个关系表达式,决定循环何时终止;表达式 3 规定每循环一次之后循环控制变量的增量。

四、do—while 语句

do

语句

while (表达式)

该循环与 while 循环不同之处在于它先执行循环体,然后再判断表达式的值,若为非 0 则继续循环,否则终止循环。

五、开关语句

switch(e)

{

case e₁:语句 1;

case e₂:语句 2;

⋮

case e_n:语句 n;

default:语句;

}

六、break 语句

break 语句通常用于循环语句和开关语句之中。当 break 用于开关语句 switch 中时,可使控制跳出 switch;如果没有 break 语句,则将成为一个死循环而无法退出。当 break 用于循环语句中时,可使循环终止。

七、continue 语句

continue 语句只用于循环语句中,它使控制跳过循环体中剩余的语句而强行执行下一次循环。

八、goto 语句

goto 语句是一种无条件转移语句,其格式为:

goto 标号;

其中标号是一个有效的标识符,这个标识符加上一个“:”一起出现在函数内,该标识符和 goto

语句须同处于一个函数中。goto 语句一般不用,因为它可能使程序层次不清,不易读,但在多层嵌套中退出时,用 goto 语句较为合理。

习 题 一

1. 数据结构研究的对象和内容是什么?
2. 解释下列名词:数据、数据元素、数据对象、数据结构和存贮结构。
3. 二阶斐波那契序列定义如下:

$$f_0 = 0, \quad f_1 = 1, \quad f_n = f_{n-1} + f_{n-2} \quad (n \geq 2)$$

试编写一个 C 语言程序,计算二阶斐波那契序列的前 n 项。

4. 试编写一个 C 语言程序,按学号顺序输出一个班级所有学生的全部信息,一个学生的信息包括学号、姓名、性别、年龄以及政治、语文、数学、外语、物理等各科成绩。
5. 试问:C 语言中一元运算 $v++$ 与 $++v$ 有何区别, $v--$ 与 $--v$ 又有何区别?
6. 在 C 语言程序中,当使用一个已定义的结构指针型的结点时,须注意什么?
7. 试编写一个 C 语言程序,从 a、b、c 三个数中挑选其中最大的一个数。