



普通高等教育“十一五”国家级规划教材 计算机系列教材

C程序设计

吴德成 主编
邓长春 齐俊英
何东武 贤继红 编著
王家海 主审

清华大学出版社





普通高等教育“十一五”国家级规划教材 计算机系列教材

吴德成 主编
邓长春 齐俊英
何东武 贤继红 编著
王家海 主审

C程序设计

清华大学出版社
北京

11500

内 容 简 介

本书参照普通高等教育“C 语言程序设计”课程教学大纲的基本要求编写,将培养应用创新能力的目标融会、贯穿于教材之中,以“够用、实用”为原则,精简传统 C 语言教材的内容知识,通过简明扼要、通俗易懂地介绍 C 语言的基本概念、基本语法和编程方法,旨在使学生了解程序设计语言的基本知识,掌握程序设计的基本方法与思路,进而较为深入地理解程序的内涵,并基本具备编程解决实际问题的能力,也为学习计算机后继课程打下良好基础。

本书不仅可以作为高等院校学生初学计算机高级语言程序设计的教材,以及计算机培训班学员 C 语言的考前培训教材,也适合广大软件开发人员和自学人员参考阅读。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

C 程序设计/吴德成主编;邓长春等编著. — 北京:清华大学出版社,2011.3

ISBN 978-7-302-24771-5

I. ①C… II. ①吴… ②邓… III. ① C 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 012487 号

责任编辑:梁 颖

责任校对:焦丽丽

责任印制:李红英

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62795954, jsjje@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185×260 印 张:14 字 数:311 千字

版 次:2011 年 3 月第 1 版 印 次:2011 年 12 月第 2 次印刷

印 数:4001~7500

定 价:25.00 元

产品编号:041053 01

C 语言是一种计算机程序设计语言。它兼有高级语言和汇编语言的特点,既可以作为系统设计语言,编写系统应用程序;又可以作为应用程序设计语言,编写不依赖计算机硬件的应用程序。因此,应用范围极为广泛。

本书参照普通高等教育 C 语言程序设计课程教学大纲的基本要求编写,将培养应用创新能力的目标融会、贯穿于教材之中,以“够用、实用”为原则,精简传统 C 语言的教材内容知识。简明扼要、通俗易懂地介绍 C 语言的基本概念、基本语法和编程方法,使学生能在短时间内快速掌握程序设计的方法。选用的实例既有科学性和易学性的演示类例题,又有趣味性和实用性的设计类例题。例题由浅入深、循序渐进,相关章节保持连续性、前后衔接、逐步扩展。由于计算机语言是实践性非常强的课程,在各章中均配有实训内容以加强上机实践这一必须重视的关键环节。本书旨在使学生了解程序设计语言的基本知识,掌握程序设计的基本方法与思路,进而较为深入地理解程序的内涵,并基本具备编程解决实际问题的能力,也为学习计算机后继课程打下良好基础。

本书共分 12 章。第 1 章重点介绍 C 语言的特点、程序的结构特点以及在 Microsoft Visual C++ 6.0 环境下运行 C 语言程序的操作方法。第 2 章介绍 C 语言编程必须掌握的一些基础知识,包括最基本的数据类型、对数据进行运算的运算符和数据的输入输出。第 3、4 章讲述 C 语言程序的两种基本程序结构,即选择结构和循环结构的特点及程序的设计方法。第 5、6、9 章分别讲述 C 语言中的高级数据类型(数组、指针、结构体等),培养读者对数据的组织能力,为进一步表示复杂的数据结构打下基础。第 7、8 章中所介绍的函数和编译预处理有助于培养模块化程序设计及解决大型应用问题的能力。第 10 章位运算的讲述有助于进一步加深对数据在计算机中存储方式的理解。第 11 章介绍了解决大量数据输入输出问题的方法。第 12 章介绍了面向对象程序设计的基础知识。

本教材由吴德成主编,并负责第 4、12 章及附录的编写,其余章节中第 1、2、6 章由邓长春编写;第 7、11 章由齐俊英;第 3、5 章由何东武编写,第 8、9、10 章由贤继红编写,全书由王家海主审。

全书概念清晰、结构合理、内容严谨、讲解透彻、重点突出、示例典型、实用性强,既考虑到初学者的特点,又能满足软件设计人员的工作需要,不仅可以作为高等院校本、专科学生初学计算机高级语言程序设计的教材以及计算机培训班学员 C 语言的考前培训教材,也适合广大软件开发人员和自学人员参考阅读。

由于作者水平有限,书中难免有错误和疏漏之处,恳请广大读者批评指正。

编 者

目 录

CONTENTS

第 1 章 C 语言概述	1
1.1 C 语言的特点	1
1.2 简单的程序例子	2
1.3 运行 C 程序的方法	4
1.3.1 运行 C 程序的一般过程	4
1.3.2 Microsoft Visual C++ 6.0 环境简介	4
习题	6
第 2 章 数据的类型与基本操作	7
2.1 基本数据类型与常数的表示	7
2.1.1 基本数据类型	7
2.1.2 常数表示	8
2.2 变量与内存	9
2.2.1 标识符	9
2.2.2 变量的定义与初始化	10
2.3 运算符与表达式	13
2.3.1 算术运算符与算术表达式	14
2.3.2 赋值运算符与赋值表达式	15
2.3.3 数据类型转换	16
2.3.4 常用数学函数	17
2.3.5 几个特殊的运算符	19
2.4 输入输出函数	20
2.4.1 printf 输出函数	20
2.4.2 scanf 输入函数	22
2.4.3 字符输入输出函数	26
习题	28
第 3 章 选择结构程序	29
3.1 选择结构程序的构成与特点	29

3.2 关系运算与逻辑运算.....	31
3.2.1 关系运算与关系表达式	31
3.2.2 逻辑运算符与逻辑表达式	32
3.3 if 语句.....	32
3.3.1 if~else~形式	32
3.3.2 if~形式	34
3.3.3 if~else if~形式	35
3.3.4 if 的嵌套	36
3.3.5 条件表达式	38
3.4 switch 语句和 break 语句.....	39
习题	41
第 4 章 循环结构程序设计	43
4.1 用 goto 语句实现循环	43
4.2 while 语句	45
4.3 do~while 语句	47
4.4 for 语句	48
4.5 多重循环.....	50
4.6 break 和 continue 语句	52
4.6.1 break 语句	52
4.6.2 continue 语句	52
4.7 应用举例.....	53
习题	60
第 5 章 数组	62
5.1 引言.....	62
5.2 一维数组.....	62
5.2.1 一维数组的定义和初始化	62
5.2.2 应用例子	64
5.3 二维数组.....	68
5.3.1 二维数组的定义和初始化	68
5.3.2 应用例子	69
5.4 字符串.....	73
5.4.1 字符数组的一般操作方法	73
5.4.2 字符串处理方法	74
习题	79
第 6 章 指针	80
6.1 变量的指针.....	80

6.2 数组指针	81
6.2.1 一维数组的指针	81
6.2.2 二维数组的指针	83
6.2.3 字符串的指针	85
6.2.4 动态存储分配	88
习题	89
第 7 章 函数	91
7.1 函数的定义和调用	91
7.1.1 函数的定义	91
7.1.2 函数的返回与函数的类型	92
7.1.3 函数的调用	93
7.2 参数传递	93
7.2.1 变量作为函数形参	94
7.2.2 数组作为函数形参	95
7.2.3 指针变量作为函数形参	98
7.2.4 命令行参数	100
7.3 函数的嵌套调用和递归调用	101
7.3.1 函数的嵌套调用	101
7.3.2 函数的递归调用	101
7.4 返回指针的函数	103
7.5 函数的指针和指向函数的指针变量	103
7.6 多文件编译和变量作用域	105
7.6.1 多文件编译	105
7.6.2 局部变量	107
7.6.3 全局变量	110
7.6.4 变量存储类型与模块化程序设计	113
习题	114
第 8 章 编译预处理	115
8.1 宏定义	115
8.2 文件包含	117
8.3 条件编译	118
习题	119
第 9 章 结构体与共用体	120
9.1 结构体的定义与引用	120
9.1.1 结构体类型和结构体变量的定义	120
9.1.2 结构体变量的使用	122

9.2 结构体数组与结构体指针	124
9.2.1 结构体数组	124
9.2.2 结构体指针	125
9.3 链表	126
9.3.1 链表概述	127
9.3.2 链表的基本操作	129
9.4 共用体	137
9.5 枚举	139
9.6 用 typedef 定义类型符	142
习题	143
第 10 章 位运算	144
10.1 位运算符及其功能	144
10.2 位段	146
习题	148
第 11 章 文件	149
11.1 文件的打开和关闭	149
11.1.1 文件类型指针	149
11.1.2 文件的打开	150
11.1.3 文件的关闭	151
11.2 文件的读写	151
11.2.1 fputc 和 fgetc 函数	151
11.2.2 fread 和 fwrite 函数	154
11.2.3 fprintf 和 fscanf 函数	158
11.3 文件定位	159
11.4 文件出错检测	161
习题	162
第 12 章 面向对象程序设计基础	163
12.1 引言	163
12.1.1 传统的结构化程序设计方法	163
12.1.2 面向对象程序设计方法	164
12.1.3 SP 方法与 OOP 方法的比较	164
12.2 C++ 对 C 的扩充	165
12.2.1 输入输出和注释	165
12.2.2 const 常量	166
12.2.3 函数	166
12.2.4 变量引用作为函数参数	168

12.2.5	动态分配内存	169
12.2.6	作用域运算符	171
12.3	类	171
12.3.1	类与对象	171
12.3.2	构造函数	175
12.3.3	析构函数	178
12.3.4	友员函数	179
12.3.5	类的静态成员	185
12.4	继承	187
12.4.1	基类与派生类的继承关系	187
12.4.2	派生类对象的构造和析构	189
12.4.3	多重继承	192
12.5	虚函数和多态性	193
12.5.1	虚函数	194
12.5.2	纯虚函数和抽象基类	196
12.5.3	虚基类	197
	习题	199
附录 A	ASCII 码对照表	201
附录 B	运算符和结合性	202
附录 C	库函数	204
参考文献		213

第 1 章 C 语言概述

1.1 C 语言的特点

C 语言是一种已得到普遍应用的计算机高级语言,在软件开发和教学领域中都具有强大生命力。C 语言可以实现许多汇编语言的功能,并且可以很好地与汇编语言结合开发出各种系统软件。本书所使用的程序例子,除了第 12 章必须在 C++ 系统环境下运行以外,都是以 Turbo C2.0 为背景的,同时也兼顾了 Microsoft Visual C++ 6.0 环境。

C 语言是一种结构化程序设计语言,C 语言的主要特点如下:

(1) 语言简洁、紧凑灵活。书写格式自由,空行、空语句、复合语句极其灵活,一条语句可以分为几行,一行内也可以写几条语句。但要注意 C 语言区分字母大小写,一般情况下都使用小写,特殊标识使用大写。

(2) 数据类型丰富。基本数据类型有字符、整数、浮点型等,浮点型分单精度和双精度,以解决精度要求和资源浪费之间的矛盾,还可以在基本数据类型的基础上构造出各种高级数据类型,如数组、指针、结构体、联合、枚举等。用这些数据类型可以实现各种复杂的数据结构,如链表、树等。指针在 C 语言中得到了充分的应用,例如指针可以作为函数参数传递,可以动态地分配内存空间,处理各种复杂的数据结构。

(3) C 程序的主要结构是函数。函数中一般使用局部变量,模块中数据互相隔离,这使得 C 语言是模块化程序设计语言。

(4) C 语言允许直接访问物理地址。能进行位操作,可以直接对硬件进行操作。而且与汇编语言接口容易,对于直接读计算机 I/O 接口和控制存储器内容等需要用汇编语言来实现的功能,C 语言利用给定的函数可以很容易地实现汇编程序调用。

(5) C 程序可移植性好。汇编语言对应某种机器硬件,并且无法移植,而 C 语言编译程序代码是公共的。因此 C 程序能方便地适用于各种类型的机器。

(6) C 源程序和生成的代码质量高。C 源程序书写紧凑,C 目标代码效率比汇编语言代码效率一般只低 10%~20%,比其他高级语言的系统开销小,适用于编制系统软件,而且对于较大的源程序可以把它分成若干块单独编译。最后,将它们链接到一起,这样修改一次就不必将整个程序重新编译一遍,可以极大地提高程序的调试速度。

(7) C 语言在语法限制方面不太严格,更强调灵活性。这可能会给初学者带来一定的困难,但也正因为如此使得程序设计人员有了更大的自由度。

1.2 简单的程序例子

下面简要介绍几个简单的程序例子,说明 C 语言的组成特点。

【例 1.1】 输出一个字符串“This is an example!”。

```
#include <stdio.h>
void main()
{
    printf("This is an example!\n");           /* 输出一串字符 */
}
```

运行这个程序将输出下面一串字符:

This is an example!

说明:

① 组成 C 程序的基本单位是函数,main 是 C 系统中一个特殊的函数,称为主函数。一个 C 程序必须有一个主函数,程序运行时总是从主函数开始执行,是程序的入口。函数由函数头和函数体两部分组成,void main()为函数头,函数体由后面一对花括号括起来的若干条语句组成,每条语句都以分号结束,函数的功能由各条语句来实现(本例子只有一条语句)。主函数名字之后一定要有一对圆括号,void 是函数的类型,为无值型,在第 7 章将详细介绍函数的使用方法。

② 例子中的“/* ... */”为注释,其目的是为了增加程序的可读性,以便进行程序交流或维护。注释可以插入到程序中需要进行注释的任何位置,但必须以“/*”开头,以“*/”结束。注释的内容没有语法要求,程序编译时将忽略程序中的注释。一个易读的程序应具有必要的注释,在实际应用程序开发过程中,在源程序代码中间加入注释对于程序的调试和维护是非常重要的。

③ 上面程序的功能是输出一串字符,是由 printf("This is an example!\n");语句实现的。该语句调用了 C 系统提供的标准库函数,printf()函数是系统预先编制的一段程序代码,专门完成各种基本类型的数据输出。括号中的“This is an example!\n”为字符串常数,是函数的参数,一对双引号是 C 语言中表示字符串常数必须的。字符串最后的“\n”不能按字符原样输出,它是一种控制字符,起到换行的作用,在输出的字符串中任何位置都可以加“\n”控制符。

④ 以“#”开头的行,称为编译预处理命令行,在程序编译前执行该命令。编译预处理命令#include <stdio.h>不是 C 语句,后面不能加分号。stdio.h 是 C 系统提供的一个头文件,其中包含对 printf()函数参数的说明(调用方法)。include 预处理命令是由编译程序完成的,它将 stdio.h 文件包含到当前源文件中来,以便编译时检查 printf()函数调用参数中的个数和类型是否符合要求。

【例 1.2】 计算两个整数的和。

```
/* 计算 c=a+b 的值 */
#include <stdio.h>
void main()
```

```

{
    int a,b,c;                /* 定义变量名字及类型 */
    a=3;                      /* 为变量 a 提供原始数据 */
    b=7;                      /* 为变量 b 提供原始数据 */
    c=a+b;                    /* 计算 */
    printf("c=%d\n",c);       /* 输出结果 */
}

```

该程序的输出结果为：

```
c=10
```

说明：

① 函数体中包括说明语句和可执行语句。`int a,b,c;`语句说明 `a`、`b` 和 `c` 为三个整型变量,以便计算机在内存中分配三个存放整型数据的存储单元。称 `a`、`b` 和 `c` 为变量,是因为其存储单元中的数据是可以改变的,可以通过赋值操作改变变量的值。

② `a=3;`语句功能是将整数 3 送到 `a` 的存储单元中。语句 `c=a+b;`的功能是先取 `a` 和 `b` 两个存储单元中的数据在运算器中相加,然后将结果保存在 `c` 变量单元中。

③ 计算结果为整型数 `c=10`,由 `printf()`函数将其输出。输出语句 `printf("c=%d\n",c);`可以简化为: `printf("%d",c);`,该函数的参数包括两部分,`c` 是要输出的数据,“`%d`”是数据输出的格式控制符字符串,控制符 `%d` 的作用是按整数格式输出 `c` 的值。“`c=`”是输出数据的提示说明,是可以省略的。

④ C 程序的书写格式非常自由,一个语句可分成若干行,一行也可以写多个语句(但分号不能省略),所以行与行之间的文字排列没有什么特殊要求。为了使程序的书写格式更能适应人们的阅读,还是要做格式上的安排,增加程序的可读性,如缩进对齐和用空格间隔等。

⑤ C 程序中区分字母大小写,一般情况都是使用小写字母,特殊的标识习惯上使用大写字母。

【例 1.3】 求两个整数的和。

```

/* 求两个整数的和 */
#include <stdio.h>
int sum(int x,int y);          /* 函数声明 */
void main()
{
    int a,b,s;                /* 变量定义 */
    a=23;b=12;                /* 提供原始数据 */
    s=sum(a,b);                /* 调用 sum 函数求 a+b 的值,并赋给 s */
    printf("两个数的和=%d\n",s); /* 输出结果 */
}

int sum(int x,int y)
{
    int z;                    /* 变量说明 */
    z=x+y;
    return z;                 /* 返回 */
}

```

程序运行结果为：

```
两个数的和=35
```

在解决实际问题中几万行的源程序被认为是小程序,如果都在主函数内编写,不仅编写难度大,后面调试也更加困难。在编写程序前,都是将一个复杂问题分解成若干个相对独立的子问题,如果必要还可以进一步分解,最后将一个复杂问题分解成若干简单的模块,每个模块的功能用一个函数来实现。组成 C 语言程序的基本单位为函数,程序运行从主函数开始,在主函数中可以调用其他函数,包括用户定义的函数和语言系统提供的库函数。

第 7 章将专门介绍函数的定义与调用。

1.3 运行 C 程序的方法

1.3.1 运行 C 程序的一般过程

由高级语言编写的程序称为源程序,计算机不能直接识别和执行,必须由语言处理程序将其翻译成由 0 和 1 构成的二进制指令代码。运行 C 程序的一般过程如下:

第 1 步:在编辑环境下将源程序输入到扩展名为 c 的文件中,Turbo C 集成系统提供编辑环境。

第 2 步:使用 C 语言处理系统提供的编译程序将源程序编译成二进制目标程序,目标程序的文件扩展名为 obj。如果程序中存在语法错误,就不能生成目标文件,并且显示错误信息。根据提示的错误信息,在编辑环境下重新修改源程序后再次编译,直到没有语法错误,并生成目标文件后进行下一步操作。

第 3 步:编译后的目标文件尽管是二进制代码文件,但计算机还不能直接执行该程序,必须使用 C 语言处理系统提供的连接程序,生成扩展名为 exe 的可执行文件。程序中各函数间的调用结合是由连接程序完成的,系统提供的输出函数和用户定义的函数都要进行连接。如果连接过程中出现错误信息,也必须回到第 1 步重新修改源程序,并重新开始编译和连接,直到生成可执行文件。

第 4 步:运行程序,并检查运行结果。如果是算法错误,只能回到第 1 步修改源程序,再重新编译连接和运行,直到得到正确的结果。

1.3.2 Microsoft Visual C++ 6.0 环境简介

1. VC6 系统的启动

单击“开始”菜单,选择“程序”|Microsoft Visual Stdio 6.0|Microsoft Visual C++ 6.0。

2. 运行一个 C++ 程序的过程

(1) 在“文件”菜单下选择“新建(New)”命令,打开“新建”对话框,如图 1.1 所示。

(2) 在“工程(Projects)”选项卡中指定下列选项:

- ① 选中 Win32 Console Application 工程类型。
- ② 在“工程(Project Name)”栏中输入工程名称,如输入:Hello。
- ③ 单击“位置(Location)”栏后的 ,可指定(或默认)应用程序存放的位置。

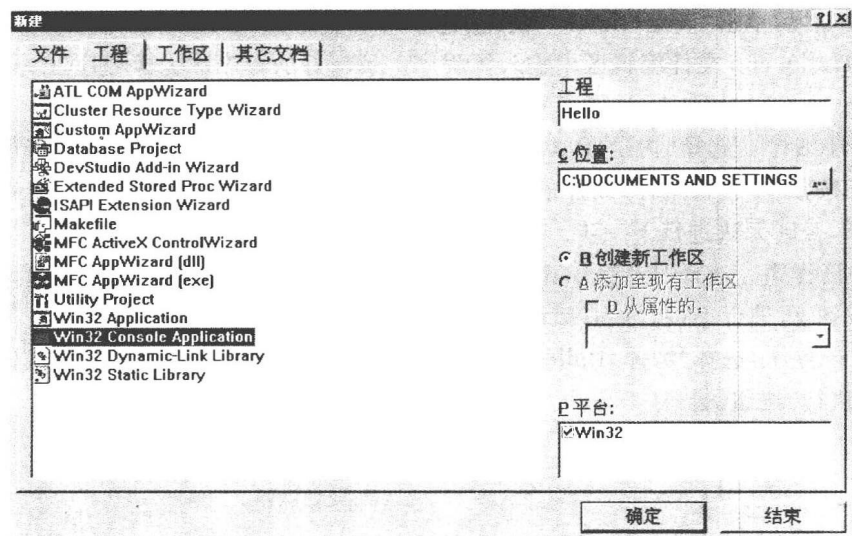


图 1.1 “新建”对话框

④ 选中 **B 创建新工作区** (Create New Workspace)(默认)。

⑤ 在 **P 平台** (Platforms) 中, 选中 **Win32** (默认)。

设置完成后, 单击“确定(OK)”按钮, 打开 AppWizard 对话框。

(3) 在应用向导(AppWizard)对话框中, 选中 An empty project. 然后单击“完成(finish)”按钮, 显示“新建工程信息”(New Project Information)对话框, 内容如下:

Empty console application.

No files will be created or add to the project.

单击“确定(OK)”按钮, 系统自动创建了一个 Hello 类。

(4) 源程序编写。单击“文件”菜单中的“新建”命令, 则弹出如图 1.2 所示的“新建”对话框。

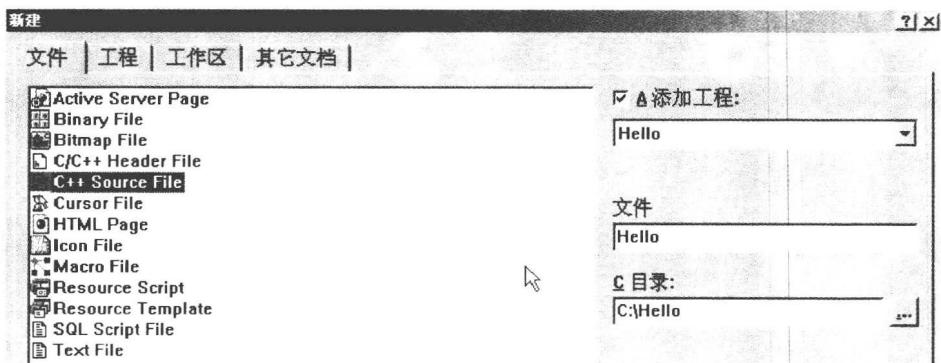


图 1.2 “新建”对话框中的“文件”选项卡

① 在弹出的“新建(New)”对话框的“文件(File)”选项卡中, 选中“C++ Source File”。在“文件(File Name)”栏中输入新建的 C++ 源文件名, 如 Hello, 确认“添加工程”复选框被

选中(默认),然后单击“确定(finish)”按钮,建立一个空的 Hello.cpp 文件。

② 在 Hello.cpp 文件编辑区中输入源程序。编辑好的程序称为源程序,C++ 源程序的扩展名为.cpp。所以,Hello 源程序的全称是 Hello.cpp。

(5) 程序编译。选择“构建(Build)”菜单中的“编译(Compile)”命令,或选择编译工具栏中的“编译”按钮进行编译,编译信息显示在输出窗口中。如有错误,则必须修改源程序再重新编译,否则无法进行下一步。

(6) 连接程序。选择“构建(Build)”菜单中的“构建(Build)”命令,或选择编译工具栏中的“构建”按钮进行连接。如有错误,则必须修改源程序再重新编译和连接。

(7) 运行程序,选择“编译(Build)”菜单中的“执行(Build Execute)”命令,或选择编译工具栏中的“执行”按钮。

习题

1.1 上机运行例 1.1 和例 1.2 两个程序例子,熟练掌握 C 程序的运行方法。

1.2 编写一个 C 程序,输出下面字符。

```
*****
This is a C program!
*****
```

1.3 假定有两个整数 $a=5$, $b=3$ 。编写程序计算 $s=a+b$, $x=a-b$, $y=a*b$, $z=a/b$ 的值。

第 2 章 数据的类型与基本操作

计算机是一种数据处理设备,学习 C 语言编程就是要通过高级语言命令完成数据处理工作。这就需要掌握计算机的控制器、运算器、存储器,以及输入输出设备的使用方法。因为控制器是执行程序命令的部件,所以应该重点掌握其他四种部件的用法:

- (1) 在运算器中完成数据处理和计算。
- (2) 在存储器(主要是内存)中读写(存取)数据。
- (3) 从输入设备获取数据,最基本的输入设备为键盘。
- (4) 将主机中的数据送到输出设备,最基本的输出设备为显示器。

2.1 基本数据类型与常数的表示

2.1.1 基本数据类型

数据是程序中的必要组成部分,是程序处理的对象。现实中的数据是有类型差异的,如姓名由一串字符组成,年龄是数字符号组成的整数,而身高包含整数和小数两部分。C 语言为不同类型的数据使用了不同的格式存储,占用内存单元的字节数也不同。编写高级语言程序虽然不需要了解数据在内存中的具体存储方法,但一定要具有类型的概念,因为处理不同类型的数据所使用的语句命令是有区别的。

C 语言提供的数据类型如图 2.1 所示。

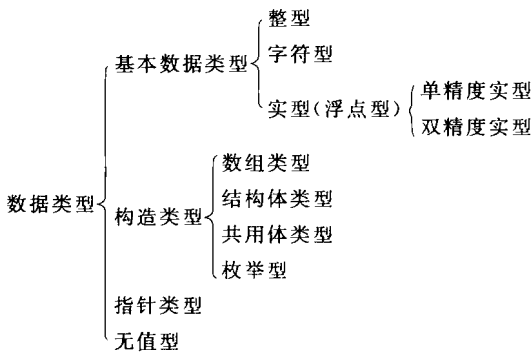


图 2.1 C 语言的数据类型

基本数据类型是 C 语言内部预先定义的数据类型,是最常用的数据类型。C 语言处理系统内部为使用基本类型的数据操作提供了非常方便的使用环境,规定了各种基本类型数据的存储格式,为完成数据运算定义了许多运算符(如加、减、乘、除等),还提供了大量的库函数(如输入输出函数)。对于初学者首先要掌握基本类型数据的表示、存储、运算和输入输