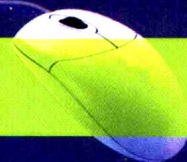


计 算 机 系 列 教 材



C语言程序设计

主编 王先水 阳小兰 尤新华



WUHAN UNIVERSITY PRESS

武汉大学出版社

计 算 机 系 列 教 材

C语言程序设计

主 编 王先水 阳小兰 尤新华

副主编 刘 艳 冯春华 陈 君



WUHAN UNIVERSITY PRESS

武汉大学出版社

1036656

图书在版编目(CIP)数据

C 语言程序设计/王先水,阳小兰,尤新华主编. —武汉:武汉大学出版社, 2012. 8

计算机系列教材

ISBN 978-7-307-10018-3

I. C… II. ①王… ②阳… ③尤… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 169366 号

责任编辑:林 莉 责任校对:黄添生 版式设计:支 笛

出版发行:武汉大学出版社 (430072 武昌 珞珈山)

(电子邮件:cbs22@whu.edu.cn 网址:www.wdp.com.cn)

印刷:湖北民政印刷厂

开本:787×1092 1/16 印张:22.25 字数:560 千字

版次:2012 年 8 月第 1 版 2012 年 8 月第 1 次印刷

ISBN 978-7-307-10018-3/TP·441 定价:39.00 元

版权所有,不得翻印;凡购买我社的图书,如有质量问题,请与当地图书销售部门联系调换。

前言

人类已经步入信息化的 21 世纪, 信息化时代, 使社会经济向知识经济转变。在高等教育逐步实现大众化后, 越来越多的高校面向国民经济发展的第一线, 为行业、企业培养各级各类实用型人才。在明确了高等学校应用型人才培养模式、培养目标、教学内容和课程体系的框架下, 为更好适应独立学院理工科学生学习程序设计语言知识的需要, 在武汉大学出版社的组织下, 我们编写了《C 语言程序设计》这本教材。

本教材在 C89 标准 C 基础上增加了 C99 相关内容, 采用 Visual C++6.0 的编译环境进行开发, 体现了“以语法为基础, 以算法为灵魂, 以设计为创新, 以规范为要求”的理念, 以培养学生良好的编程习惯和开发软件的能力。

本教材从 C 语言程序设计的基本原理及程序设计的基本思想出发, 紧扣基础, 面向应用, 循序渐进引导学生学习程序设计的思想和方法; 在编写过程中突出“三基”(基本概念、基本原理、基本应用) 的讲解与应用; 对大部分例题进行了算法分析的讲解及应用过程中的易错内容的提示, 通过这些内容的讲解为学生学习程序设计思想减少不必要的弯路; 部分章节采用案例形式, 通过提出问题、分析问题、解决问题的编写思路, 使学生更好理解 C 语言的知识, 提高实际编程能力, 从而激发学生学习程序设计浓厚兴趣。

本教材在编写过程中得到了中国地质大学江城学院的大力支持, 同时还得到了武昌理工学院、湖北大学知行学院的大力支持与指导。全书在中国地质大学江城学院机械与电子信息学部指导下, 由计算机科学与技术教研室编写完成, 王先水担任主编并对全书进行了统稿。本教材共分 11 章。第 1 章 C 语言程序设计概述; 第 2 章 C 语言基本数据类型和表达式; 第 3 章顺序结构程序设计; 第 4 章选择结构程序设计; 第 5 章循环结构程序设计; 第 6 章数组; 第 7 章函数与编译预处理; 第 8 章指针; 第 9 章结构体与共用体; 第 10 章文件; 第 11 章综合实训。

本教材在编写过程中, 参考了 C 语言程序设计的相关书籍及杂志等资料, 引用了相关教材的部分内容, 吸取了同行的宝贵经验, 在此谨表谢意。由于编者水平有限, 加上时间仓促, 书中难免有不妥和疏漏之处, 敬请各位读者提出宝贵意见和建议。

编 者

2012 年 7 月于武汉

目 录

第 1 章 C 语言程序设计概述	1
1.1 程序及程序设计语言	1
1.1.1 程序的基本概念	1
1.1.2 程序设计语言	2
1.2 C 语言的发展及特点	3
1.2.1 C 语言的发展概述	3
1.2.2 C 语言的特点	4
1.3 C 语言程序的基本结构	5
1.4 C 语言字符集、标识符和关键字	9
1.4.1 C 语言字符集	9
1.4.2 C 语言标识符	9
1.4.3 C 语言关键字	10
1.5 C 语言程序的开发环境	10
1.5.1 Visual C++6.0 集成开发环境介绍	11
1.5.2 在 Visual C++6.0 环境下建立和运行 C 程序的步骤	11
第 2 章 C 语言基本数据类型和表达式	20
2.1 C 语言的基本数据类型	20
2.1.1 数据类型概述	20
2.1.2 整型数据	21
2.1.3 实型数据	23
2.1.4 字符型数据	23
2.2 常量与变量	25
2.2.1 常量与符号常量	25
2.2.2 变量与变量定义	26
2.3 C 语言表达式与运算符	28
2.3.1 算术运算符与算术表达式	28
2.3.2 赋值运算符与赋值表达式	29
2.3.3 自增自减运算符	30
2.3.4 逗号运算符与条件运算符	31
2.4 数据类型转换	33
2.4.1 自动转换	33
2.4.2 赋值转换	34



2.4.3 强制转换	35
第 3 章 顺序结构程序设计	41
3.1 C 语言的基本语句	41
3.1.1 简单语句	41
3.1.2 赋值语句	42
3.1.3 复合语句	42
3.2 格式输出函数 printf	42
3.3 格式输入函数 scanf	47
3.4 字符输出函数 putchar	49
3.5 字符输入函数 getchar	50
3.6 程序设计举例	51
第 4 章 选择结构程序设计	60
4.1 关系运算符与关系表达式	60
4.1.1 关系运算符	60
4.1.2 关系表达式	61
4.2 逻辑运算符与逻辑表达式	61
4.2.1 逻辑运算符	61
4.2.2 逻辑表达式	62
4.3 if 语句	63
4.3.1 单分支 if 语句	63
4.3.2 双分支 if 语句	64
4.3.3 多分支 if 结构	65
4.3.4 if 语句的嵌套	67
4.4 switch 语句	69
4.5 程序设计举例	71
第 5 章 循环结构程序设计	78
5.1 while 语句	78
5.2 do-while 语句	80
5.3 for 语句	81
5.3.1 for 语句的基本形式	81
5.3.2 for 语句中各表达式含义	82
5.3.3 for 语句的变形	83
5.3.4 for 语句与 while 语句和 do-while 语句比较	84
5.4 break 语句和 continue 语句	84
5.4.1 break 语句	84
5.4.2 continue 语句	86
5.4.3 break 语句和 continue 语句的区别	87



5.5 循环的嵌套结构	87
5.5.1 双重循环的嵌套	88
5.5.2 多重循环的嵌套	90
5.6 程序设计举例	91
第6章 数组	104
6.1 一维数组的定义和引用	105
6.1.1 一维数组的定义方式	105
6.1.2 一维数组元素的引用	106
6.1.3 一维数组元素的存储结构	108
6.1.4 一维数组元素的初始化	108
6.1.5 一维数组程序举例	109
6.2 二维数组的定义和引用	112
6.2.1 二维数组的定义	112
6.2.2 二维数组元素的引用	113
6.2.3 二维数组元素的存储结构	114
6.2.4 二维数组的初始化	114
6.2.5 二维数组程序举例	116
6.3 字符数组	117
6.3.1 字符数组的定义	117
6.3.2 字符数组元素的引用及初始化	117
6.3.3 字符串的概念及存储结构	118
6.3.4 字符串的输入与输出	119
6.3.5 字符串处理函数	120
6.4 程序举例	123
第7章 函数与编译预处理	135
7.1 模块化程序设计 with 函数	135
7.2 函数的定义与调用	136
7.2.1 标准库函数	137
7.2.2 函数的定义	138
7.2.3 函数的调用	140
7.2.4 函数参数传递	143
7.2.5 函数程序设计举例	144
7.3 函数的嵌套调用	146
7.4 函数的递归调用	148
7.5 变量作用域与存储方式	151
7.5.1 变量作用域	151
7.5.2 变量的存储方式	154
7.6 编译预处理	157



7.6.1 宏定义	157
7.6.2 文件包含	160
7.6.3 条件编译	161

第 8 章 指针

8.1 指针与指针变量	169
8.1.1 指针的概念	169
8.1.2 指针变量	171
8.1.3 指针变量的定义	171
8.1.4 指针变量的初始化	172
8.1.5 指针运算符	173
8.1.6 指针运算	174
* 8.1.7 多级指针	178
8.2 指针与数组	178
8.2.1 一维数组元素的指针访问	179
8.2.2 二维数组元素的指针访问	184
8.2.3 指向一维数组的指针	188
8.3 字符指针与字符串	193
8.3.1 字符串的表现形式	194
8.3.2 用字符指针表示字符串	195
8.3.3 字符串数组	197
8.4 指针数组	197
8.5 指针与函数	200
8.5.1 指针作为函数参数	200
8.5.2 指向函数的指针	201
8.5.3 返回指针值的函数	203
8.6 动态指针	204
8.7 指针程序设计举例	206

第 9 章 结构体与共用体

9.1 结构体的概念	218
9.1.1 结构体类型的定义	218
9.1.2 结构体类型变量的定义	219
9.1.3 结构体类型变量的引用	222
9.1.4 结构体类型变量的初始化	223
9.2 结构体数组与链表	224
9.2.1 结构体数组的定义与引用	224
9.2.2 结构体数组初始化和应用	225
9.2.3 链表	226
9.3 共用体的概念	235

9.3.1 共用体类型的定义	235
9.3.2 共用体类型变量的定义	236
9.3.3 共用体类型变量的引用	236
9.4 程序设计举例	239
第10章 文件	254
10.1 文件的概述	254
10.1.1 文件数据的存储形式	254
10.1.2 文件的处理方法	255
10.1.3 文件类型的指针	255
10.2 文件的常用操作	256
10.2.1 文件的打开与关闭	256
10.2.2 文件的读写与定位	258
10.2.3 文件的检测	264
第11章 C语言综合实训	268
11.1 分支与循环	268
11.1.1 算法与示例	268
11.1.2 实训题目	270
11.2 数组与函数	270
11.2.1 算法与示例	270
11.2.2 实训题目	272
11.3 指针	273
11.3.1 算法与示例	273
11.3.2 实训题目	275
11.4 结构体	275
11.4.1 算法与示例	275
11.4.2 实训题目	277
11.5 综合实训	277
11.5.1 简单的银行自动取款机系统	277
11.5.2 总体设计	277
11.5.3 详细设计	278
11.5.4 设计代码	287
11.5.5 系统测试	293
11.5.6 综合实训题目	295
附录 I ASCII 表	298
附录 II 运算符及优先级	300
附录 III C语言常用语法提要	301



附录 IV	2008 年 9 月全国计算机等级考试笔试试卷二级公共基础知识和 C 语言程序设计.....	305
附录 V	C 语言函数.....	314
附录 VI	第三届“蓝桥杯”全国软件专业人才设计与创业大赛预选赛 (本科组 C 语言) 试题.....	335
参考文献		344



第1章 C 语言程序设计概述



【学习目标】

1. 了解程序设计语言的发展和 C 语言的特点。
2. 掌握 C 语言程序的基本结构。
3. 能正确运用 C 语言的标识符及关键字。
4. 能熟练运用 Visual C++ 集成开发环境创建 C 语言源程序、编译、连接和运行的基本操作要点。
5. 建立程序设计的基本思想。

自 1946 年第一台计算机问世以来, 计算机学科的发展逐步引起人们的高度关注, 计算机学科的应用越来越广泛。人们使用计算机管理大量的数据, 处理纷繁复杂的办公事务, 使用计算机完成复杂的科学计算, 加快科学研究的进程, 使用计算机实现网络通信拉近人们的空间距离。计算机本身是无生命的机器, 要使计算机能为人类完成各种各样的工作, 就必须让它执行预先编制好并存储于计算机内存的程序, 这些程序是依靠程序设计语言编写出来的。但在众多的程序设计语言中, C 语言有其独特之处, C 语言不仅能编写操作系统软件, 还能编写应用软件, 是一种高级语言, 学习起来很容易, 是众多高级语言学习基础语言。C 语言具有直接操作计算机硬件的功能, 具备低级语言即汇编语言的特点, 也是目前盛行的嵌入系统中应用的语言之一。如果你认真学习本书, 认真思考本书介绍的知识, 并在本书的指导下认真上机实践, 将会很快掌握使用 C 语言编写程序的方法, 并逐渐领悟到 C 语言的精妙所在。你想编写出人们喜爱的实用程序吗? 你想成为一个优秀的程序设计员吗? 那我们就一起走进 C 语言的世界吧。

1.1 程序及程序设计语言

我们都知道, 计算机完成的任何工作, 都是计算机运行程序的结果。而计算机运行的程序又都是使用某种程序设计语言编写的, 自从计算机诞生以来, 程序设计语言已经经历了机器语言、汇编语言和高级语言等几个主要发展阶段。

1.1.1 程序的基本概念

让计算机按人们的意图处理事务, 人们必须事先设计好完成任务的方法, 用计算机语言来描述算法(即方法)称计算机程序。程序是用计算机语言描述的解决某一问题的具体步骤和方法且符合一定语法规则的符号序列。人们借助计算机语言, 告诉计算机要做什么(即要处理哪些数据), 计算机如何处理(即按什么步骤来处理)这个过程称为程序设计。计算机语言是面向计算机的人造语言, 是进行程序设计的工具, 称计算机语言为程序设计语言。



在解决某一问题时,由于人们的思维方式不同,所设计的程序有所不同,不同的程序在执行时其效率也是不同的,影响程序的执行效率因素主要有算法和程序运行中的参数的数据类型。作为一名优秀的程序设计人员设计的程序符合:一是程序设计的算法是最优的;二是程序的执行效率是最快的即时间最短;三是程序的开销是最小的即程序存储空间小。

1.1.2 程序设计语言

程序设计语言的发展经历了机器语言、汇编语言和高级语言三个主要阶段。了解程序设计语言的发展,更加有助于读者加深对程序设计语言的认识,更好地运用程序设计语言来解决一些实际问题。

1. 机器语言

机器语言是人们最早使用的程序设计语言。由于计算机硬件只能识别和处理 0 和 1 所组成的代码。因此机器语言是 0 和 1 两个代码组成的机器指令序列,控制硬件完成指定的操作。

例如,以下是某计算机的两条机器指令:

加法指令: 10000000

减法指令: 10010000

用机器语言编写的程序能够被计算机直接识别并执行,程序的执行效率特别高,这是机器语言的最大优点。但机器语言与人们习惯使用的自然语言相差太大,难读、难记、难写、难修改、用它来编写程序很不方便。并且硬件设备不同的计算机,它的机器语言也有差别,编写的程序缺乏通用性。编写机器语言程序时,要求程序员必须相当熟悉计算机的硬件结构,因而现在人们一般不使用机器语言编写程序来解决一些实际应用问题。

2. 汇编语言

20 世纪 50 年代中期,为了减轻人们使用机器语言编程的负担,开始使用一些助记符号来表示机器语言中的机器指令,便产生了汇编语言。助记符采用代表某种操作的英文单词的缩写。例如:上例中的两条指令用汇编程序描述如下:

ADD A, B ; 其中的 A、B 表示的是两个操作数。

SUB A, B

上述两条汇编指令计算机不能直接执行,它必须经过一个汇编程序的系统软件翻译成机器指令后才能执行。用汇编语言编写的程序称汇编语言源程序,将汇编语言源程序翻译成机器能执行的程序称为汇编程序。

汇编语言指令和机器语言指令之间具有一一对应的关系,因此不同的计算机其汇编语言也不尽相同,并且程序编写时仍需要对计算机的内部结构比较熟悉,但相对于机器语言就简单多了。

早期的操作系统软件主要是用汇编语言编写的,汇编语言和机器语言一样,对不同的计算机硬件设备,需要使用不同的汇编语言指令,因此汇编语言程序不利于在不同计算机系统之间移植。所以,现在的汇编语言一般在专业程序设计员中使用,而非专业程序设计员编写应用程序则较少使用汇编语言。

3. 高级语言

汇编语言和机器语言是面向机器的,它们属于低级语言的范畴。人们在使用它设计程序时,要求对机器比较熟悉。为了克服低级语言的缺点,将程序设计的重点放在解决问题(即算



法)方面。于是产生了面向过程或面向对象的高级语言。例如: C语言、C++语言、Visual C++语言、Java语言。由于高级语言是面向过程或面向对象的计算机语言, 它们在形式上非常接近于人们习惯使用的自然语言, 它不直接对计算机的硬件操作, 因此, 用高级语言编写的程序可以适合用于不同硬件设备的计算机, 这给人们用计算机语言编程来解决实际应用问题带来了极大的方便。

1.2 C语言的发展及特点

C语言是一种结构化的高级语言, 它简洁、紧凑、使用方便灵活。用C语言编写的程序执行效率高, 可移植性好, 基本不做修改就能用于各种型号的计算机和各种操作系统, 并且C语言还能实现汇编语言的大部分功能。C语言有着广泛的应用领域, 不仅用来编写操作系统软件, 也可用来编写应用软件, 特别是近几年C语言成为嵌入系统开发的首选语言。

1.2.1 C语言的发展概述

C语言是在20世纪70年代初由美国贝尔实验室的Dennis M. Ritchie设计的, 并首先安装在UNIX操作系统的DEC PDP-11计算机上实现, 因而最初的C语言是为了描述和实现UNIX操作系统而设计的。

到了1973年, K.Thompson和Dennis M. Ritchie两人合作把UNIX的90%以上内容用C语言改写(即UNIX第5版)。后来, 人们对C语言进行了多次的改进, 其主要还是在贝尔实验室内部使用, 直到1975年UNIX第6版公布后, C语言的突出优点才引起人们的广泛关注。

1978年由美国(AT&T)贝尔实验室正式发表了C语言。同时由B.W.Kernighan和D.M.Ritchie合著了著名的*THE C PROGRAMMING LANGUAGE*一书。简称为K&R, 也称为K&R标准。但是, 在K&R中并没有定义一个完整的标准C语言, 后来由美国国家标准协会(American National Standards Institute)在此基础上制定了一个C语言标准, 于1983年发表, 称为ANSI C。

ANSI C标准于1989年被采用, 该标准一般称为ANSI/ISO Standard C, 于是1989年定义的C标准定义为C89。其中详细说明了使用C语言书写程序的形式, 规范对这些程序的解释。包括: C语言的表示法, C语言的语法和约束, 解释C程序的语义规则, C程序输入和输出的表示, 一份标准实现了限定和约束。

到了1995年, 出现了C的修订版, 增加了库函数, 出现了初步的C++, C89便成了C++的子集。由于C语言的不断发展, 在1999年又推出了C99, C99在保留C特性基础上增加了一系列新的特性, 形成了C99标准。

目前在微机上最流行的C语言版本主要有: Microsoft C/C++, Turbo C, Quick C, Visual C/C++。

这些C语言版本不仅实现了ANSI C标准, 而且在此基础上各自作了一些扩充, 使之更加方便、完美。

本教材以Visual C++6.0为集成开发环境。在此环境下对C语言做介绍。Visual C++6.0是Microsoft公司推出的在Windows环境下, 进行应用程序开发的可视化与面向对象程序设计软件开发工具。



1.2.2 C 语言的特点

C 语言是一种用途广泛、功能强大、使用灵活的过程性编程语言，既可用于编写应用软件，又能用于编写系统软件。学习和使用 C 语言的人越来越多，同时也是工科院校学生必修的一门计算机语言课程。掌握 C 语言成了计算机开发人员的一项基本功。C 语言的特点概括起来主要有以下几点：

1. 中级语言

C 语言具有高级的易读易写的特点，将高级语言的设计思想与汇编语言进行了有机整合。具有可以直接访问内存物理地址的汇编语言特性，能进行位、字节和地址的操作，具有对硬件进行编程实现对系统的控制，是目前嵌入式系统开发的首选语言。

2. 易学易用

C 语言简洁、紧凑，使用方便、灵活。ANSI C 一共只有 32 个关键字。如表 1-1 所示。34 种运算符和 9 种结构化控制语句。程序书写格式较为自由，特适合初学程序设计的人员学习使用。

表 1-1

ANSI C 关键字

auto	break	case	char	const	continue	default
do	double	else	enum	extern	float	for
goto	if	int	long	register	return	short
signed	static	sizeof	struct	switch	typedef	union
unsigned	void	volatile	while			

说明：在 C99 标准中增加了新的关键字。如：_Bool、_Imaginary、restrict、_Complex、inline。在 C89、C99 的 C 语言中，关键字都是小写的。

3. 结构化语言

C 语言程序的基本单位是函数，一个 C 语言程序文件是由一系列的函数组成，一个函数能较好实现某一特定功能，这就便于进行模块化程序设计，使程序设计人员较好运用“自顶向下、逐步求精”的结构化程序设计技术。

4. 可移植性好

用 C 语言编写的程序不依赖于计算机的硬件系统特性和操作系统，在不同的计算机系统间移植。

5. 目标代码质量高速度快

C 语言程序编译后生成目标代码质量高，程序执行速度快。C 语言编写的程序一般只比汇编程序生成的目标代码效率低 10%~20%。

C 语言的这些特点对初学者来说一时还难以体会其精髓，通过 C 语言的学习后再来领悟就会有不同的理解。

C 语言虽然有很多优点，但是也存在一些缺点，如运算优先级太多、数值运算能力也不像其他高级语言强，语法定义不严格，对数组下标越界不做检查等。尽管 C 语言目前还存在一些不足，由于它目标代码质量高、使用灵活、数据类型丰富、可移植性强而得到广泛的普



及和迅速的发展，成为广大用户非常喜爱的程序设计语言。

1.3 C语言程序的基本结构

为了更进一步理解C语言源程序结构的特点，下面通过几个源程序来进行说明。这几个程序由简到难，表现了C语言源程序在组成结构上的特点。虽然其有关内容还没有介绍，但可从这几个例子中了解到组成一个C源程序的基本部分和书写格式。

【例题 1.1】 在显示器上输出：“The university welcomes you!”

程序分析：对初学者来说，要完成这样的一个程序，首先要熟悉 Visual C++6.0 集成开发环境，其次要了解C语言源程序的基本结构。

程序代码：

```
#include <stdio.h>           //编译预处理命令
int main()                   //定义主函数
{                             //函数开始标志
    printf("The university welcomes you! \n"); //输出字符串信息
    return 0;                //函数执行完返回函数值 0
}                             //函数结束标志
```

include 称为文件包含命令，扩展名为.h 的文件称为头文件。其作用是将输入输出函数包含到本程序文件中。

main 是函数名，表明是一个主函数并且 main 函数的第一条可执行语句开始执行。该函数的功能是在显示器上打印一串字符 The university welcomes you!

每一个C源程序都必须有且只能有一个主函数即 main 函数。main 函数是C语言编译系统使用的专用名字。

main 函数后面用花括号对“{ }”括起来的部分是函数体即程序实现的功能。

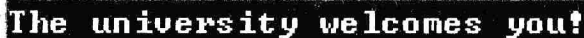
函数调用语句，printf 函数的功能是把要输出的内容（The university welcomes you!）送到显示器上显示出来。

printf 函数是一个由系统定义的标准函数，可在程序中直接调用。

分号：是C语言语句的结束标志。

从这个简单C程序中可知：C语言程序的基本结构由包含命令、主函数、花括号，实现功能要求的函数或语句组成。

调试运行：



如果程序只能在显示器打印出字符串，那就没有什么意义。程序如何要解决数据的计算并将结果打印在屏幕上。

【例题 1.2】 编写程序：计算两个指定数的和并将结果打印在屏幕上。

程序分析：主函数功能实现两个数的计算，如何表示这两个数，两数相加的和又如何表示，这就需要用到变量的概念。现用符号 value1、value2 表示两个数分别为 5、2，用符号 sum 表示这两个数的和。运用加法对其进行运算，调用输出函数 printf()将其结果



打印在屏幕上。

程序代码:

```
#include <stdio.h>

int main()
{ int value1,value2,sum;
  value1=5;
  value2=2;
  sum=value1+value2;
  printf("The sum is%d\n",sum);
  return 0;
}
```

本题同例题 1.1 相比较有以下不同之处:

定义三个整型数据变量 value1,value2 和 sum。

分别对变量 value1、value2 赋给整型数值 5、2。

将两个变量的值相加得到的和保存在变量 sum 中。

printf 函数的双撇内的“%d”位置上输出一个具体的整型数值,其值是逗号后的变量 sum 的值。双撇内的“\n”是输出结果后光标换行。

调试运行:



思考:

(1) 若将 printf("The sum is %d\n",sum) 中的 sum 换成 10, 则程序输出的结果是什么?

(2) 若要输出算术算式, 则如何修改 printf 函数双撇号内的表现形式?

【例题 1.3】编写程序: 要求计算任意两个数的和并以算式形式输出。

程序分析: 在例题 1.2 中用符号来表示指定的两个数表明是静态的, 而本例中任意两个数表明是动态的, 那如何将一个动态的值给这两个符号呢? 这就需要调用 C 语言的输入函数 scanf。当两个符号取得值后, 运用加法进行运算。如何以算术算式形式输出, 这就需要在输出函数 printf 中用三个格式符 %d 分别表示被加数、加数及和的形式输出。

程序代码:

```
#include <stdio.h>

int main()
{
  int value1,value2,sum;
  printf("请通过键盘输入两个整数: \n"); //程序运行时在屏幕上打印提示信息
  scanf("%d%d",&value1,&value2);
  sum=value1+value2;
  printf("%d+%d=%d\n",value1,value2,sum);
  return 0;
}
```



本题同例题 1.1、例题 1.2 相比较有以下不同之处:

“/”之后的内容构成 C 语言程序的注释部分。不参与程序的编译和执行,只是起说明作用,增加程序的可读性。读者在学习编程时应养成添加注释的习惯。

函数调用语句。scanf 的功能是从键盘获得输入的两个整型数据 value1、value2。scanf 是一个由系统定义的标准函数,可在程序中直接调用。

scanf("%d%d",&value1,&value2)语句中的双撇内容指的是输入的两个数按整型数据格式输入且输入完第一个数后,按一下空格键或按回车键再输入第二个数。符号&是地址运算符,用以指明输入的两个数被存放在计算机内存以 value1、value2 为名称的单元地址中。

printf("%d+%d=%d\n",value1,value2,sum)语句中的双撇的“%d”位置上输出具体的值并分别对应其逗号后三个变量 value1、value2、sum 的值。其中的“+”和“=”按原样形式输出。

printf 和 scanf 这两个函数分别称为格式输出函数和格式输入函数,其意义是按指定的格式输出输入值,其具体使用方法在后续章节中进行学习。

在使用 scanf 和 printf 两个标准函数时,一定要用到 stdio.h 头文件将其包含到 C 源程序中。

【例题 1.4】 编写程序:要求实现输入任意的两个整数输出其中的较大数。

程序分析:通过例题 1.3 的认识,能做到通过键盘输入任意的两个数给指定的符号。如何对两个符号的量的大小进行判断,这就需要用到条件语句即 if 语句。但下列编写的程序代码是采用函数调用方式来实现的,其目的是让学生明白用 C 语言编写程序时采用不同的形式更能充分发挥自己潜在智能,从而激发学生学习程序设计语言的浓厚兴趣。

程序代码:

```
#include <stdio.h>
int main()                                /*主函数*/
{
    int max(int a,int b);                  /* 对被调函数的 max 的声明*/
    int value1,value2,largenumber;        /* 定义三个变量 */
    printf("请通过键盘输入两个整数: \n"); /* 提示输入两个整数 */
    scanf("%d%d",&value1,&value2);        /* 输入 value1,value2 变量的值 */
    largenumber=max(value1,value2);        /* 调用 max 函数,得到的值赋给 largenumber*/
    printf("largenumber=%d\n",largenumber);
    return 0;
}
int max(int a,int b)                      /*定义 max 函数,其值为整型,形式参数 x、y 也是整型*/
{int c;                                  /* max 函数中声明部分,定义本函数中用到的变量 c */
    if(a>b)
        c=a;
    else
        c=b;
    return(c);                            /* 将 c 的值返回,通过 max 带回到调用函数的位置 */
}
```

本题同上面几个例子相比较有以下不同之处: