

C程序设计 实践教程

李振立 张慧萍 主编



科学出版社

版权所有,侵权必究

举报电话:010-64030229;010-64034315;13501151303

内 容 简 介

本书是《C 程序设计》的配套实践教材,共分 10 章,各章均由学习指导和实训两部分构成。学习指导包括预习指导、学习指导、典型案例分析和解题方法、分析方法、习题与习题解答等内容;实训包括实验要求、编程方法、程序调试方法、实验报告等栏目与内容。本书从预习、学习、复习、练习、上机、调试和应试各个教学环节为学生提供案例和学习参考内容,训练学生计算思维能力、分析能力和编程能力。

本书由长期工作在教学一线的教师编写,全书各知识单元编排层次清楚、条理分明、结构合理而严谨、案例丰富、详略度把握得体。与《C 程序设计》配套,可作为各类高等院校本、专科非计算机专业的“C 语言程序设计”课程教材,也可以作为独立学院、高职高专、网络学院的教材。

图书在版编目(CIP)数据

C 程序设计实践教程/李振立,张慧萍主编. —北京:科学出版社,2011.11
ISBN 978-7-03 032524 2

I ①C… II. ①李…②张… III. ①C 语言—程序设计—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2011)第 206004 号

责任编辑:张颖兵 程 欣/责任校对:梅 莹
责任印制:彭 超/封面设计:苏 波

科 学 出 版 社 出 版

北京东黄城根北街 16 号

邮政编码 100717

<http://www.sciencep.com>

武汉市新华印刷有限责任公司印刷

科学出版社发行 各地新华书店经销

*

2011 年 10 月第 一 版 开本 787×1092 1/16

2011 年 10 月第一次印刷 印张 14 3/4

印数:1—3 000 字数 338 000

定价:26.00 元

(如有印装质量问题,我社负责调换)

前 言

本书是《C 程序设计》的配套实践教材,两书各章内容相互对应并相互补充。本书从预习、学习、复习、练习、上机、调试和应试各个教学环节为学生提供案例和学习参考内容,从而训练学生计算思维能力、分析能力和编程能力。

本书通过大量的教学案例,为学生提供较为全面的程序设计思想、程序分析方法和解题方法,提供分析和解决问题的基本过程和思路,着重介绍以内存分析为主的图解法。全书的各个章节都可以使用图解分析方法解析程序的存储与操作过程,将源程序中的每条语句解析为对内存数据的操作。

C 语言的练习题目浩如烟海,全国计算机等级考试 C 语言试题、各种考试的 C 语言试题在网上数不胜数,本书将考点进行归类总结,采用经典案例对试题进行分类指导,让学生学会方法,举一反三、融会贯通,品味解题、编程的乐趣。

本书配置了大量运行在 Visual C++ 环境下的例题,以集成开发环境 Visual C++ 6.0 为开发工具,介绍了 Visual C++ 的编辑、编译、运行、排错、调试等使用方法,帮助学生熟悉 C 语言编译器,快速掌握编辑、编译、运行 C 程序的方法,学会使用集成开发环境开发简单应用软件。

全书分为 10 章,分别为 C 语言概述指导与实训、数据类型与表达式指导与实训、顺序结构程序设计指导与实训、选择结构程序设计指导与实训、循环结构程序设计指导与实训、数组指导与实训、函数指导与实训、指针指导与实训、结构体与共用体指导与实训及文件指导与实训。

本书由长期工作在教学一线的教师编写,全书各知识单元编排与教材相同,相互配合、内容互补、结构合理而严谨、案例丰富、分门别类,具有知识性与趣味性。与《C 程序设计》配套,可作为各类高等院校本、专科非计算机专业的“C 语言程序设计”课程教材,也可以作为独立学院、高职高专、网络学院的教材,对于社会计算机学习者,尤其是具有一定计算机基础的广大计算机爱好者,是一本极好的自学读物。

本书由李振立、张慧萍主编,张胜利、楚维善、高金兰、郭盛刚、顾梦霞、陈小娟、贺红艳、吴佩、罗宏芳、周雪芹、王世畅、李珺参与了编写。另外,在本书的策划、编写、出版过程中,王学文给予了大力支持,在此深表谢意。

作 者
2011 年 8 月

目 录

前言

第 1 章 C 语言概述指导与实训	1
1.1 教材的预习及学习指导	1
1.1.1 教材预习指导	1
1.1.2 教材学习指导	1
1.2 分析方法	7
1.2.1 案例分析	7
1.2.2 编程方法	8
1.3 习题与解答	10
1.3.1 练习题	10
1.3.2 习题答案	13
1.4 C 语言基本操作实验	15
1.4.1 集成开发环境 VC++	15
1.4.2 实验报告	19
第 2 章 数据类型与表达式指导与实训	23
2.1 教材的预习及学习指导	23
2.1.1 教材预习指导	23
2.1.2 教材学习指导	23
2.1.3 补充教材——数制与编码	28
2.2 分析方法	34
2.2.1 案例分析	34
2.2.2 编程方法	38
2.3 习题与解答	40
2.3.1 练习题	40
2.3.2 习题解答	42
2.4 数据类型与表达式实验	45
2.4.1 程序调试方法	45
2.4.2 实验预习与实验报告	47
第 3 章 顺序结构程序设计指导与实训	51
3.1 教材的预习及学习指导	51
3.1.1 教材预习指导	51
3.1.2 教材学习指导	51

3.2 分析方法	54
3.2.1 案例分析	54
3.2.2 算法分析	57
3.3 习题与解答	60
3.3.1 练习题	60
3.3.2 习题答案	64
3.4 顺序结构程序设计实验	66
3.4.1 顺序结构程序设计	66
3.4.2 实验报告	68
第4章 选择结构程序设计指导与实训	71
4.1 教材的预习及学习指导	71
4.1.1 教材预习指导	71
4.1.2 教材学习指导	71
4.2 分析方法	74
4.2.1 案例分析	74
4.2.2 算法分析	78
4.3 习题与解答	81
4.3.1 练习题	81
4.3.2 习题答案	85
4.4 选择结构程序设计实验	89
4.4.1 选择结构程序设计	89
4.4.2 预习作业与实验报告	92
第5章 循环结构程序设计指导与实训	95
5.1 教材的预习及学习指导	95
5.1.1 教材预习指导	95
5.1.2 教材学习指导	95
5.2 分析方法	97
5.2.1 算法分析	97
5.2.2 案例分析	101
5.3 习题与解答	104
5.3.1 练习题	104
5.3.2 习题解答	107
5.4 循环结构程序设计实验	112
5.4.1 循环结构程序设计	112
5.4.2 预习作业与实验报告	116
第6章 数组指导与实训	119
6.1 教材的预习及学习指导	119
6.1.1 教材预习指导	119
6.1.2 教材学习指导	119

6.2 分析方法	123
6.2.1 算法分析	123
6.2.2 案例分析	126
6.3 习题与解答	131
6.3.1 练习题	131
6.3.2 习题解答	135
6.4 数组实验	137
6.4.1 数组编程方法	137
6.4.2 实验	140
第7章 函数指导与实训	143
7.1 教材的预习及学习指导	143
7.1.1 教材预习指导	143
7.1.2 教材学习指导	143
7.2 分析方法	147
7.2.1 数据传递分析	147
7.2.2 案例分析	151
7.3 习题与解答	154
7.3.1 练习题	154
7.3.2 习题解答	159
7.4 函数实验	161
7.4.1 函数编程方法	161
7.4.2 实验报告	167
第8章 指针指导与实训	170
8.1 教材的预习及学习指导	170
8.1.1 教材预习指导	170
8.1.2 教材学习指导	170
8.2 分析方法	175
8.2.1 指针分析	175
8.2.2 案例分析	178
8.3 习题与解答	182
8.3.1 练习题	182
8.3.2 习题解答	186
8.4 指针实验	187
8.4.1 编程方法	187
8.4.2 实验报告	191
第9章 结构体与共用体指导与实训	194
9.1 教材的预习及学习指导	194
9.1.1 教材预习指导	194
9.1.2 教材学习指导	194

9.2 数据类型分析	197
9.2.1 结构体数据类型	198
9.2.2 共用体数据类型	198
9.2.3 自定义类型	199
9.2.4 枚举类型	199
9.2.5 链表	200
9.3 习题与解答	201
9.3.1 练习题	201
9.3.2 习题答案	205
9.4 结构体与共用体实验	207
9.4.1 实验要求	207
9.4.2 实验报告	207
第 10 章 文件指导与实训	211
10.1 教材的预习及学习指导	211
10.1.1 教材预习指导	211
10.1.2 教材学习指导	211
10.2 习题与解答	215
10.2.1 练习题	215
10.2.2 习题解答	220
10.3 文件实验	222
10.3.1 文件编程方法	222
10.3.2 实验预习与实验报告	226

第 1 章 C 语言概述指导与实训

1.1 教材的预习及学习指导

1.1.1 教材预习指导

本章主要介绍 C 语言的程序架构、程序逻辑顺序、程序风格和程序的组成元素及 C 语言词法中的单词,简单介绍 C 语言常用的编译器如 VC++,Dev C++,TC 等。

本章分为 4 节,第 2 节 C 语言的程序架构是预习的重点。

第 1 节 C 语言的发展史,预习时只需浏览 C 语言的起源、C 语言的特点、C 语言的集成开发环境等内容。第 2 节从 C 语言程序的基本架构入手,了解 C 语言函数模块一般形式。通过读程序了解什么是编译预处理命令,什么是函数的定义,函数中声明部分用于声明变量或函数,C 语言中变量或函数必须要满足先定义后使用的规则,执行部分常用哪些语句。注意 C 语言程序中输入、处理和输出三者的逻辑顺序。学习 C 语言的书写风格。第 3 节 C 语言的单词,单词包括分隔符、注释符、关键字、标识符、常量、运算符等。预习的重点包括 C 语言基本字符集、关键字、标识符、常量与常量的类型、运算符的使用方法,掌握 C 语言的词法构成。第 4 节为 Dev C++集成开发环境,预习的重点是 Dev C++的使用方法,新建源文件,编辑源文件,编译和调试文件,运行文件。

1.1.2 教材学习指导

1. C 语言基本概念

◎ C 语言集高级语言和低级语言的优点于一身,适于作为系统描述语言,用于编写大型的操作系统,编写编译系统,编写应用软件。

◎ C 语言属于面向过程的程序设计语言,采用结构化、模块化的方法设计源程序。

◎ 面向过程的程序=算法+数据结构。

◎ ISO:国际标准化组织。

◎ ANSI:美国国家标准协会。

◎ GNU:是一个自由软件工程项目,由自由软件社团开发和维护。

◎ “K & R”是以名著《The C Programming Language》的作者命名的早期 C 语言的事实标准,称为经典 C 语言。

◎ C 语言的美国国家标准(ANSI C)是 1983 年美国国家标准协会(ANSI)对 C 语言进行扩充和规范制定的标准。

◎ C89 是指 1989 年 ISO/IEC 提出的国际标准草案,1990 年公布的 C 语言正式标准

称为 C89,有时也称为 C90。

◎ C99 是指 1999 年 12 月 16 日 ISO/IEC 推出的 C 语言标准:ISO/IEC 9899:1999 (Programming languages-C)称为 C99。

◎ C 语言具有语言简洁、紧凑,使用灵活、方便,运算符丰富,表达能力强,数据结构丰富,结构化程度高,生成目标代码质量高,程序执行效率高,可移植性好诸多特点。

◎ C 语言集成开发环境 IDE 都是由编辑器、编译器、连接器集合而成,常用的 C 与 C++集成开发环境有 Borland 公司的 Turbo C, Borland C++, C++ Builder;微软的 Microsoft C, Visual C++, Visual studio. NET;多平台的 C++集成开发环境有 GCC, MinGW, QT, eclipse+CDT 等软件。

◎ C 程序的基本架构由编译预处理命令、参数说明、函数说明、主函数、函数等成分组成。其中主函数由函数头部、函数体组成。函数头部包括函数类型、函数名、参数说明,函数体由声明部分、语句部分等成分组成。

◎ 编译预处理命令有文件包含、宏定义和条件编译三类命令。

◎ C 语言是函数型语言,主程序和程序模块都是函数。

◎ 函数定义的参数为虚参,函数调用的参数为实参,参数按位置虚实对应进行传递。

◎ 变量、函数应该满足先定义后调用的规则。如果主函数在前,函数的定义在后,需要用函数原型说明函数,并说明函数的类型、参数类型和个数。

◎ C 语句用分号“;”作结束符,编译预处理命令结束没有分号“;”。

◎ 注释分为块注释和行注释,块注释用一对符号“/* */”作程序中注释的定界符,表示“/*”和“*/”之间的内容是注释;用“//”引导行注释。

◎ C 程序的逻辑顺序依次按数据声明、数据输入、数据处理、数据输出的次序排列的。

◎ 局部说明定义变量和函数,确定数据的类型和取值范围。数据输入包括函数参数传递的数据,变量初始化时输入的数据,输入函数和赋值语句输入的数据等。数据处理是根据解题的算法编制的程序,数据输出是用输出函数输出指定的数据。

◎ C 语言的风格指 C 程序的风范格局,是程序员对程序书写形式一贯性的体现。

◎ C 语言符号体系的基础是基本字符集,基本字符集是编写源程序时准用字符的集合,C 语言编译程序能够识别集合中的字符。准用字符包括大写字母、小写字母、数字、空白符、图形符号等。转义符号是由“\”开头,后跟指定的字符表示转换成另外意义的符号。

◎ 单词是基本字符集中的若干字符组合成一个具有独立意义的最小词法单位,是一组形式化的数据符号。单词包括分隔符、注释符、关键字、标识符、常量、运算符 6 类。

◎ 词法分析负责从构成源程序的基本字符集中识别和分离出单词,为语法分析提供单词类别和单词自身值的信息。词法分析对源程序进行行编辑,删除源程序中的注释、空白符以及对语法分析无关的信息。区别单词是关键字还是标识符,为语法分析作准备。

◎ 分隔符是用来分隔程序的正文、语句或单词,用来表示某个程序实体的结束和另一个程序实体的开始。C 分隔符由空白符和标点符号组成。

◎ 空白符包括空格符、水平制表符(HT)、垂直制表符(VT)、回车符(CR)、换行符(LF)、换页符(FF)等,空白符用于语句行之间的分割。

◎ 常用的标点符号包括逗号、分号、冒号、花括号{}等。

◎ 关键字是由编译程序预定义具有固定含义的单词,关键字有特定含义的专门用途,用户不能用关键字作为常量、变量、类型或函数的名字。

◎ 标识符是定义符号常量、标号、变量、类型、函数、对象的定义符。

◎ 标识符的命名规则是以英文字母或下划线开头的英文大小字母、数字和下划线符组成的序列,标识符中的字母区分大小写,标识符的长度为1~31个字节,C99标准规定标识符的长度为63字节。标识符的中间不能有分割符,不能使用C的关键字作为标识符。

◎ 常量指在程序运行和处理的过程中其值始终不能被改变的量。

◎ 常量的基本类型包括整型常量、字符型常量、字符串常量、浮点型常量4种基本类型。

◎ 整型常量是由字符和数字组成的序列,用以表示整数包括正整数、负整数和零。

◎ C语言整型常量包括十进制、八进制和十六进制三种数制表示方法。

◎ 整型常数的后缀表示有L(或l)作后缀表示长整数,U(或u)作后缀表示无符数。整型常量后缀可以是U和L(u或l)的组合,表示unsigned long类型的常量。

◎ 浮点型常量指带有小数的十进制数。

◎ 浮点型常量有两种表示形式:一种是带小数的十进制表示形式它由整数部分和小数点后的尾数部分组成;另一种方法是指数形式,由带小数的十进制数后加e(E)及整数的指数部分表示,e(E)指数部分表示10的整数次方,如1.234e2表示 1.234×10^2 。

◎ 浮点型常量分为单精度(float)、双精度(double)和长双精度(long double)三类。浮点型常量在默认情况下为double型,若要表示float型常量,则在实数后加f(F);表示long double则在实数后加l(L)。

◎ C字符常量是用单引号括起来的一个基本字符。

◎ 字符常量有两种表示形式:一种是准用字符表示形式,用单引号括起准用字符,如'a';另一种是转义符号表示形式,用单引号括起转义符号,如'\n','\t','\b'等。

◎ 字符串常量用一对双括号括起来的字符序列。例如,"Hello!","a"等。字符序列中的字符包括空格符、准用字符、转义字符和扩展的ASCII码字符等。

◎ 符号常量按照先定义后使用的原则,先用"#define 标识符 字符串"进行宏定义,然后在程序中使用符号常量。例如"#define PI 3.1415926"。

◎ C语言运算符分为单目运算符、双目运算符、三目运算符。单目运算符指只对一个操作数进行操作的运算符;双目运算符是对两个操作数进行操作的运算符;三目运算符是对三个操作数进行操作的运算符,需要操作数最多的运算符是三目运算符即条件运算符,条件运算符格式为“表达式1? 表达式2:表达式3”。

◎ 运算符分为十六种优先级和两类结合性,优先级指运算符的优先次序,结合性指当一个运算量两边的运算符级别相同时指定的结合方向。

2. C语言语法与词法

(1) C语言的函数模块一般形式如表1.1.1所示。

表 1.1.1 C 语言函数模块

函数模块格式	源程序	函数模板
编译预处理命令 函数类型 函数名 (函数形式参数) { 声明语句 执行语句 }	<pre>#include<stdio.h> main() { int iA=6,iB=8,iC; iC=iA+iB; printf("iC=%d\n",iC); }</pre>	<pre>#include<stdio.h> main() { }</pre>
函数类型 函数名 (参数表列) { 局部变量声明 语句执行 [return 语句] }	<pre>int iMax(int iX,int iY) { int iZ; if(iX>iY)iZ=iX; else iZ=iY; return(iZ); }</pre>	<pre>int iMax(int iX,int iY) { return(); }</pre>

主函数模板由编译预处理命令、主函数头部、函数体开始与函数体结束等部分组成。

(2) 编译预处理命令格式,见表 1.1.2。

表 1.1.2 编译预处理命令

命令格式	命令实例	功能
# include <filename. h>	例: # include <stdio. h>	从标准库目录开始搜索
# include "filename. h"	例: # include "stdlib. h"	从用户工作目录开始搜索
# define 标识符 字符串	例: # define PI 3. 1415926	宏定义,定义符号常量 PI

(3) 常用程序语句,见表 1.1.3。

表 1.1.3 常用程序语句

命令类型	语句格式	语句
输入	scanf("输入格式",输入项地址表列);	scanf("%d,%d",&iA,&iB);
赋值	变量=表达式;	iC=iA+iB;
输出	printf("输出格式",输出表列);	printf("iC=%d\n",iC);

(4) C 语言基本字符集,见表 1.1.4。

表 1.1.4 基本字符集

类型	基本字符
大写字母	A B C D E F G H I JX Y Z
小写字母	a b c d e f g h i jx y z
数字	0 1 2 3 4 5 6 7 8 9
空白符	空格符、水平制表符(HT)、垂直制表符(VT)、回车符(CR)、换行符(LF)、换页符(FF)

续表

类型	基本字符
转义符号	\n 换行(LF) \t 水平制表(HT) \a 响铃(BEL) \b 退格(BS) \f 换页(FF) \r 回车(CR) \v 垂直制表(VT) \反斜杠 \'单引号 \"双引号 \? 问号 \0 空字符(NULL) \ddd 三位八进制数 \xhh 二位十六进制数
图形符号	~!#%&?' `*+=+ - _/\“ ’ () [] { } < > , . : ; ?

(5) 分隔符。

表 1.1.5 分隔符

类型	字符
空白符	空格符、水平制表符(HT)、垂直制表符(VT)、回车符(CR)、换行符(LF)、换页符(FF)
标点符号	逗号、分号、冒号、花括号{}

(6) 注释符。

表 1.1.6 注释符

注释名	符号	举例	说明
块注释	/*.....*/	/* 注释内容 */	所有 C 标准
行注释	//.....	//注释内容直到行尾	C99 标准

(7) 关键字。

表 1.1.7 关键字

关键字类型	保留字
C 关键字(基本集)32 个	auto break case char const continue default do double else enum extern float for goto if int long register return static short signed sizeof struct switch typedef unsigned union void volatile while
Turbo C 扩展 7 个关键字	asm cdecl far huge interrupt near pascal
C99 新增 5 个关键字	restrict inline _Complex _Imaginary _Bool

(8) 标识符。

表 1.1.8 标识符

对象	定义格式	定义举例	标识符
符号常量	#define 标识符 字符串	#define PI 3.1415926	PI
标号	标号名:	Lable:	Lable
类型	类型名	char,int,float,double	int
变量与数组	类型名 变量名表	int iK,iA[3]={1,2,3}	iK,iA
函数	[类型说明符] 函数名([形参表])	int iMax(int iA,int iB)	iMax

(9) 常量。

表 1.1.9 常量

对象	常量格式	常量举例	后缀
整型常量	ddd, 0ddd, 0xdd	-123, 0384, 0x4A	73L 长整型, 65U 无符号数
字符型常量	'基本字符'、'转义符'	'A', 'b', '\n', '\t'	
字符串常量	"字符序列"	"Hello! "	
浮点型常量	±ddd.dd, ±ddd.dde±dd	-123.45, 1.2e4	-3.8f 单精度, 6.9L 长双精度

(10) 运算符。

表 1.1.10 运算符

名称	运算符	举例	优先级	结合性
小括号	()	(a+b)*c		左到右
数组下标	[]	array[4]		左到右
函数调用	函数名()	func(a,b)		左到右
成员运算符	.(结构或联合成员)	Stu.num	1	左到右
指向运算符	->(结构或联合成员)	Stu->num		左到右
自增(后缀)	(变量)++	a++		左到右
自减(后缀)	(变量)--	b--		左到右
自增(前缀)	++(变量)	++a		右到左
自减(前缀)	--(变量)	--b		右到左
取地址	& 变量	&a		右到左
取内容	* 指针	*p		右到左
一元正号	+	+5	2	右到左
一元负号	-	-8		右到左
按位求反	~	~a		右到左
逻辑非	!	!a		右到左
计算所需空间	sizeof	sizeof(a)		右到左
强制类型转换	(类型)	(float)a	3	右到左
乘、除、整除、求余	*, /, /, %	a*b, a/2.0, a/b, a%b	4	左到右
加、减	+, -	a+b, a-b	5	左到右
位左移、位右移	<<, >>	a<<2, a>>4	6	左到右
关系	<, <=, >, >=	b<3, b<=4, b<2, b>=4	7	左到右
等于、不等于	==, !=	a==4, a!=4	8	左到右
按位与	&	a&b	9	左到右
按位异或	^	a^b	10	左到右
按位或		a b	11	左到右
逻辑与	&&	a&& b	12	左到右
逻辑或		a b	13	左到右
条件(三目运算符)	?:	a>b? a:b	14	右到左
赋值	=, +=, -=, *=, /=, %=, &	a=b+3	15	右到左
逗号	,	a=8, b=3	16	左到右

1.2 分析方法

1.2.1 案例分析

1. 分析转义字符

例 1.2.1 字符串中包含转义字符,分析简单 C 语言源程序的输出结果。(程序名 syl_1.c)

```
#include <stdio.h>           //包含标准输入输出头文件;
main()                       //经典 C 的主函数头部;
{                             //函数体开始;
    printf("He\154\x6Co!\tok!\n"); //调用标准输出函数;
}                             //函数体结束。
```

分析:在输出的字符串中,\154 是八进制数表示字符“l”,\x6C 是十六进制数也表示字符“l”,转义字符\t 表示过一个制表位,源程序编译、运行后,输出结果为

```
Hello!      ok!
```

2. 用赋值语句计算和处理数据

例 1.2.2 已知电路参数 $L=2.0$ 和 $C=1.0$,求谐振频率 f_0 。(程序名 syl_2.c)

```
#include <stdio.h>           //包含标准输入输出头文件;
#include <math.h>            //包含数学头文件
main()                       //经典 C 的主函数头部;
{                             //函数体开始;
    double L=2.0,C=1.0,f0;   //输入 L,C 的值;
    f0=1/(2*3.1415926*sqrt(L*C)); //用赋值语句计算谐振频率 f0;
    printf("f0=%f",f0);      //调用标准输出函数 printf 输出;
}                             //函数体结束。
```

分析:编译预处理命令和语句的功能如程序右边所示,程序由输入数据 L,C,计算谐振频率 f_0 ,输出谐振频率 f_0 的值。

3. 用宏定义设置符号常量

例 1.2.3 程序如下,试分析宏代换的方法,并写出程序运行后的结果。(程序名 syl_3.c)

```
#define N 6
#define M 2
main()
{
    int iS;
    iS=N/M;
```

```
printf("%d",iS);
}
```

分析:宏定义命令定义符号常量 N 为 6, M 为 2, 在编译处理时对源程序中所有的符号常量 N, M 进行宏代换, 代换后 $iS=6/2$ 运行程序时计算 iS 的值。

4. 输入数据的提示

例 1.2.4 已知直角三角形的斜边长为 10, 一直角边长为 8, 求另一直角边之长。
(程序名 syl_4.c)

```
#include <stdio.h>           //包含标准输入输出头文件;
#include <math.h>             //包含数学头文件
main()                       //经典 C 的主函数头部;
{                             //函数体开始;
    double s,a,b;            //声明双精度变量 s,a,b;
    printf("Input s,a");     //在屏幕上显示待输入的变量 s,a
    scanf("%d,%d",s,a);      //输入 s,a 的值 10.0,8.0;
    b=sqrt(c*c-a*a);         //用赋值语句计算另一直角边;
    printf("b=%f",b);        //调用标准输出函数 printf 输出数据;
}                             //函数体结束。
```

分析:程序编译、运行时,执行到 scanf 语句时,编译器等候用户输入数据,如果没有提示信息,往往不知道应该输入什么数据,因此,在输入数据之前用 printf 提示用户输入数据是一个良好的习惯,方便用户输入数据。

1.2.2 编程方法

编程是从主函数模板开始编写的,根据主函数的定义,从简单到复杂,逐步增加编译命令和语句,形成不同功能的程序。

1. 编写输出字符和文字的程序

编写输出字符和文字的程序时,需要用标准输出函数 printf。因此,要包含标准输入输出头文件;主函数定义时,一般采用经典 C 的规范;然后,用标准输出函数 printf 输出字符和文字。

例 1.2.5 编写输出字符串“Turbo C”的源程序。

解 编写输出字符串“Turbo C”的源程序,只需要在主函数模板中加入 printf("Turbo C\n");语句,形成最简单的源程序。(程序名 syl_5.c)

```
#include <stdio.h>           //包含标准输入输出头文件,程序调用标准输出函数;
main()                       //经典 C 的主函数头部;
{                             //函数体开始;
    printf("Turbo C\n");     //调用标准输出函数;
}                             //函数体结束。
```

编译、运行程序输出结果如下:

Turbo C

2. 编写用赋值语句计算和处理数据的程序

编写用赋值语句计算和处理数据的程序时,要根据程序的逻辑顺序,在主函数的模板中先编写输入数据。可以在声明变量时为变量赋初值,也可以用标准输入函数输入指定的数据,还可以用宏定义指定符号常量的值,再写计算和处理数据的语句。然后,用标准输出函数 printf 输出数据。

例 1.2.6 已知物体的质量 $m=2$,重力加速度 $g=9.8$,编程求物体所受的重力 f 。

解 根据重力公式, $f=m * g$,用三种方法输入数据。(程序名 syl_6_1.c, syl_6_2.c, syl_6_3.c)

声明变量时为变量赋初值	用标准函数输入数据	用宏定义输入数据
<pre>#include <stdio.h> main() { double m=2,g=9.8,f; f=m*g; printf("f=%f\n",f); }</pre>	<pre>#include <stdio.h> main() { double m,g,f; scanf("%f,%f",&m,&g); f=m*g; printf("f=%f\n",f); }</pre>	<pre>#include <stdio.h> #define m 2 #define g 9.8 main() { double f; f=m*g; printf("f=%f\n",f); }</pre>
f=19.600000	f=19.600000	f=19.600000

3. 根据程序的逻辑顺序编程

例 1.2.7 将华氏温度 f 变换成摄氏温度 c ,计算公式为 $c=5 * (f-32)/9$,若华氏温度 $f=50$ 度,试编程求摄氏温度 c 。

解 程序的逻辑顺序由变量声明、数据输入、数据处理和数据输出四条语句构成,先声明变量 c 和 f 为双精度型变量,调用标准输入函数 scanf 输入变量 f 的值 50.0,再用赋值语句计算摄氏温度 c 的值,然后调用标准输出函数 printf 输出 c 的数据。(程序名 syl_7.c)

```
#include <stdio.h>           //包含标准输入输出头文件;
main()                       //经典 C 的主函数头部;
{                             //函数体开始;
    double c,f;              //定义双精度变量 c,f;
    scanf("%d",&f);           //调用标准输入函数,输入变量 f 的值;
    c= 5*(f-32)/9;           //用赋值语句求 c 的值;
    printf("c=%f\n",c);      //调用标准输出函数,输出变量 c 的值;
}                             //函数体结束。
```

4. 用 9 区图形字符显示菜单文本

例 1.2.8 用 C 语言输出如图 1.2.1 所示的菜单文本,试编写程序。

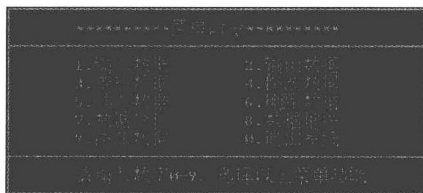


图 1.2.1 菜单文本

解 菜单文本中包含 9 区图形字符字符可以使用软键盘中的按键绘制,操作方法如下:

首先打开中文输入方式的状态窗口,右击“软键盘”按钮,打开快捷菜单,选择“制表符”,打开制表符键盘,在需要输入图形字符时输入字符。VC 中支持在字符串中显示中文,可以在字符串中直接输入中文文字,编制程序如下:(程序名 sy1_8.c)

```
#include <stdio.h>
main()
{
    printf("┌───────────────────────────────────┐ \n");
    printf("│ *****菜单命令***** │ \n");
    printf("└───────────────────────────────────┘ \n");
    printf("│      1.输入数据          2.输出数据      │ \n");
    printf("│      3.查找数据          4.修改数据      │ \n");
    printf("│      5.插入数据          6.删除数据      │ \n");
    printf("│      7.数据分析          8.数据排序      │ \n");
    printf("│      9.添加数据          0.退出系统      │ \n");
    printf("└───────────────────────────────────┘ \n");
    printf("│      请输入数字 0-9,选择以上功能性      │ \n");
    printf("└───────────────────────────────────┘ \n");
}
```

1.3 习题与解答

1.3.1 练习题

1. 判断题(共 10 小题,每题 1 分,共 10 分)

- (1) C 语言是在 B 语言的基础上发展起来的。 ()
- (2) C 语言集高级语言和低级语言的优点于一身,适用于作为系统描述语言。 ()