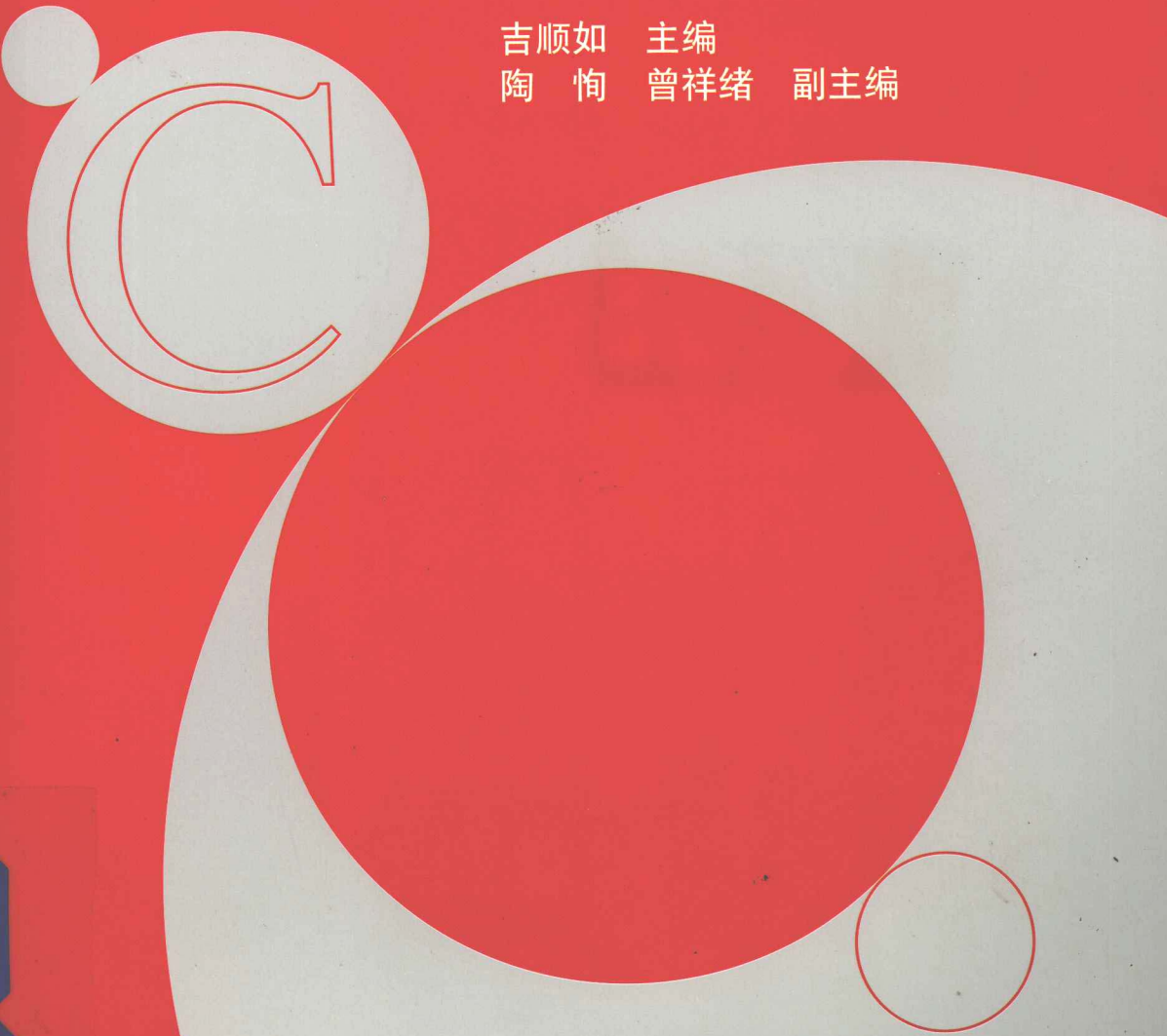


高等院校计算机基础教育规划教材

# C程序设计 教程与实验

吉顺如 主编  
陶 恂 曾祥绪 副主编

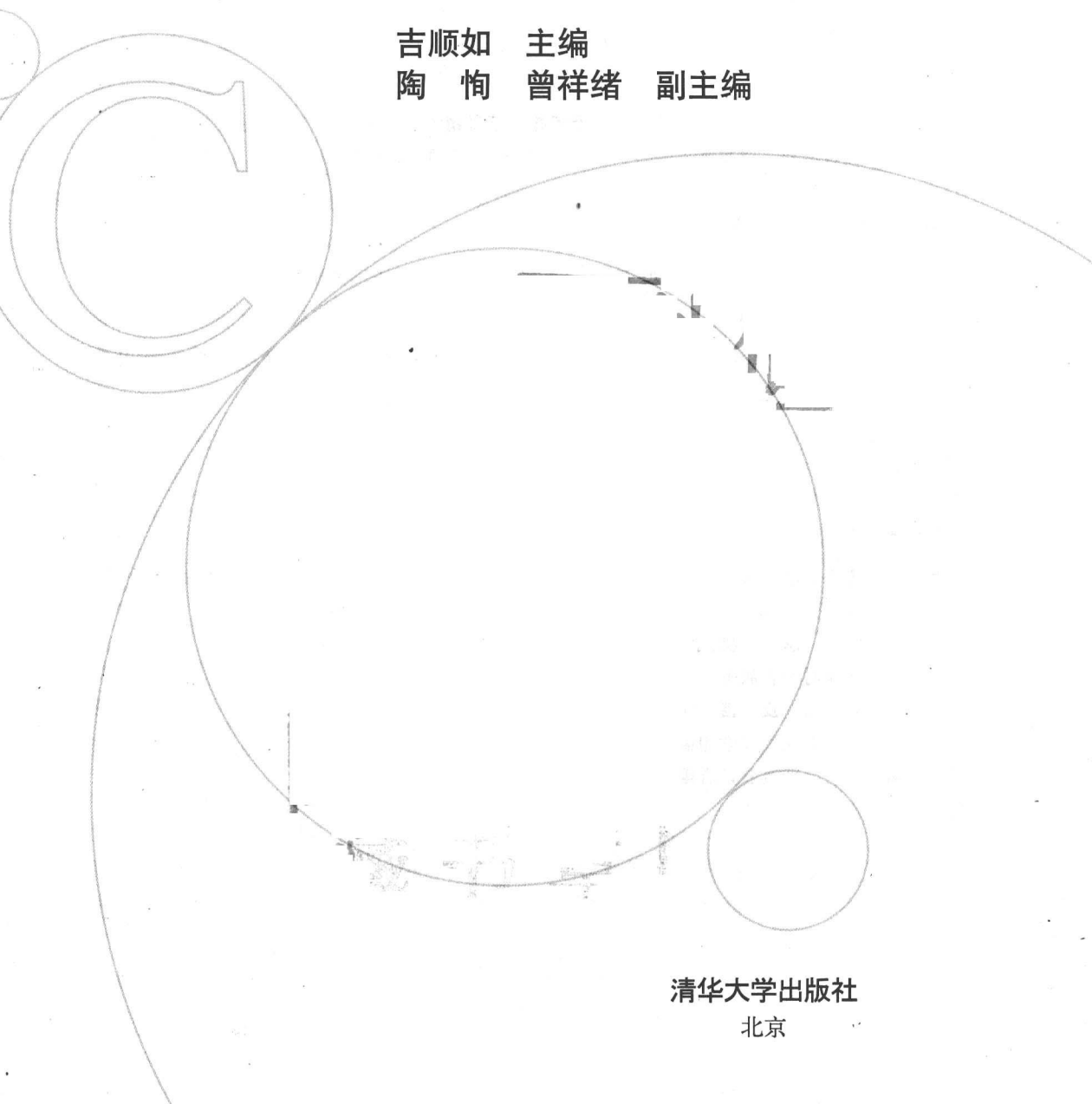


清华大学出版社

高等院校计算机基础教育规划教材

# C程序设计 教程与实验

吉顺如 主编  
陶 恂 曾祥绪 副主编



清华大学出版社  
北京

## 内 容 简 介

本书在内容编排上力求重点突出、难点分散,在语言描述上注重概念清晰、通俗易懂,并通过大量的例题分析将理论知识与实践相结合,以期逐步提高学生编写程序的能力。

全书共分12章,内容包括C语言的基本结构、数据类型及其运算符和表达式、C程序的3种控制结构、数组、函数、指针、结构体与共用体、文件。尤其在第11章通过一款智能充电机的编程实例,介绍了C语言在单片机编程中的具体应用。本书每章均配有精心设计的难易适当的习题供学生课后练习,上机实验题包括改错题、程序填空题及编程题,可以让学生在反复实践中提高设计程序和调试程序的能力。

本书可作为高等院校理工科各专业“高级语言程序设计”课程的教材,也可供对程序设计有兴趣的读者参考。

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程序设计教程与实验/吉顺如主编. —北京:清华大学出版社,2011.6

ISBN 978-7-302-25358-7

I. ①C… II. ①吉… 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(2011)第070537号

责任编辑:刘士平 刘翰鹏

责任校对:袁芳

责任印制:杨艳

出版发行:清华大学出版社

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

社总机:010-62770175

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质量反馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

地 址:北京清华大学学研大厦A座

邮 编:100084

邮 购:010-62786544

印 刷 者:北京市人民文学印刷厂

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185×260 印 张:21 字 数:498千字

版 次:2011年6月第1版 印 次:2011年6月第1次印刷

印 数:1~3000

定 价:33.00元

产品编号:042555-01

C语言是国内外广泛使用的计算机程序设计语言,是高等院校理工科相关专业重要的专业基础课程。由于C语言功能丰富,使用灵活方便,程序执行效率高,可移植性好,既可以用来编写系统程序,又可以用来编写应用程序,因而它越来越受到人们的欢迎。

本书是根据教育部《关于进一步加强高等学校计算机基础教学的意见》的教学基本要求和高等院校计算机基础教学改革的需要,结合作者多年讲授C语言程序设计课程的教学经验编写而成。

本书的教学目标是培养学生的逻辑思维能力和程序设计能力,因此在内容编排上力求重点突出、难点分散;在语言描述上注重概念清晰、通俗易懂;并通过大量的例题分析将理论知识与实践相结合,以期逐步提高学生编写程序的能力。

本书共分12章,内容安排如下。

第1章主要介绍C程序的基本结构以及C程序的开发过程。

第2章主要介绍C语言的数据类型、常量和变量、运算符和表达式。

第3章主要介绍算法的概念、特性和表示方法;基本语句,预处理命令,输入/输出函数。

第4章主要介绍关系运算符和关系运算表达式、逻辑运算符和逻辑运算表达式、选择结构控制语句。

第5章主要介绍循环控制语句、循环的嵌套。

第6章主要介绍一维数组、二维数组和字符数组及其应用。

第7章主要介绍函数的定义和调用、变量的作用域和存储类别。

第8章主要介绍指针的概念、指针变量的定义以及指针与数组、指针与函数的关系。

第9章主要介绍结构体、链表、共用体的定义及应用。

第10章主要介绍文件的基本操作和使用规则。

第11章通过一款智能充电机的编程实例,介绍C语言在单片机编程中的具体应用。

第12章包括与第1~10章内容配套的实验题目,每套题目均包含改错题、程序填空题和编程题。

由于C语言程序设计是一门理论性、实践性均较强的课程,为了帮助学生掌握有关的基本概念和程序设计方法,每章后面均精心设计了难易适当、题型丰富的习题供学生课后练习。这些习题和第12章的实验题目结合使用,可以使学生在反复实践中提高设计程序和调试程序的能力。

本书由上海电机学院的吉顺如主编,计春雷主审。编写分工为:第 1、6、10、12 章及附录由吉顺如编写;第 2 章由张艳编写;第 3、4 章由陶恂编写;第 5、7 章由杨立勤编写;第 8、9 章由王中华编写;第 11 章由曾祥绪编写。全书由吉顺如统稿。本书的程序编译环境是 VC++ 6.0。

在本书的编写过程中得到许多老师的帮助,在此表示诚挚的谢意。由于编者水平有限,书中错误在所难免,恳请读者批评指正。

编 者

2011 年 1 月 10 日

# 目 录

Contents

第 1 章 C 语言程序设计概述 .....	1
1.1 C 程序简介 .....	1
1.2 C 程序的结构 .....	3
1.3 C 语言的特点 .....	4
1.4 C 程序的书写格式 .....	5
1.5 C 程序的开发过程及 VC++ 6.0 集成开发环境 .....	5
1.5.1 C 程序的开发过程 .....	5
1.5.2 VC++ 6.0 集成开发环境 .....	6
1.6 习题 .....	13
第 2 章 数据类型、运算符和表达式 .....	16
2.1 关键字与标识符 .....	16
2.2 常量与变量 .....	17
2.2.1 常量 .....	17
2.2.2 变量 .....	18
2.3 基本数据类型 .....	18
2.3.1 整型数据 .....	18
2.3.2 实型数据 .....	21
2.3.3 字符型数据 .....	22
2.4 运算符和表达式 .....	26
2.4.1 算术运算符和算术运算表达式 .....	26
2.4.2 赋值运算符和赋值表达式 .....	27
2.4.3 逗号运算符和逗号表达式 .....	29
2.4.4 强制类型转换运算符 .....	30
2.4.5 sizeof 运算符 .....	30
2.4.6 不同数据类型的数据间的混合运算 .....	31
2.5 自增、自减运算 .....	32
2.6 位运算 .....	33
2.7 应用举例 .....	35
2.8 习题 .....	36

<b>第 3 章 顺序结构程序设计</b> .....	<b>39</b>
3.1 算法简介 .....	39
3.1.1 算法的概念 .....	39
3.1.2 算法的特性 .....	40
3.1.3 算法的表示 .....	41
3.2 基本语句 .....	43
3.2.1 程序的结构 .....	43
3.2.2 C 语言的基本语句 .....	44
3.2.3 赋值语句 .....	45
3.3 预处理命令 .....	46
3.3.1 不带参数的宏定义 .....	46
3.3.2 带参数的宏定义 .....	47
3.3.3 文件包含 .....	49
3.3.4 条件编译 .....	50
3.4 输入/输出 .....	52
3.4.1 格式输入/输出函数 .....	52
3.4.2 字符输入/输出函数 .....	58
3.5 应用举例 .....	59
3.6 习题 .....	61
<b>第 4 章 选择结构程序设计</b> .....	<b>65</b>
4.1 关系运算符和关系运算表达式 .....	65
4.2 逻辑运算符和逻辑运算表达式 .....	66
4.3 选择结构控制语句 .....	67
4.3.1 if 语句 .....	67
4.3.2 switch 语句 .....	70
4.4 条件运算符 .....	72
4.5 应用举例 .....	73
4.6 习题 .....	79
<b>第 5 章 循环结构程序设计</b> .....	<b>84</b>
5.1 循环结构的控制语句 .....	84
5.1.1 while 循环语句 .....	84
5.1.2 do...while 循环语句 .....	85
5.1.3 for 循环语句 .....	86
5.2 循环的嵌套 .....	87
5.3 break 语句和 continue 语句 .....	89

5.3.1	break 语句	89
5.3.2	continue 语句	90
5.4	应用举例	91
5.5	习题	94
<b>第 6 章</b>	<b>数组</b>	<b>101</b>
6.1	一维数组	101
6.1.1	一维数组的定义和引用	101
6.1.2	一维数组的初始化	102
6.1.3	一维数组的应用举例	103
6.2	二维数组	105
6.2.1	二维数组的定义和引用	105
6.2.2	二维数组的初始化	106
6.2.3	二维数组的应用举例	107
6.3	字符数组	110
6.3.1	字符数组的定义和引用	110
6.3.2	字符数组的初始化	111
6.3.3	字符串	112
6.3.4	常用的字符串处理函数	113
6.4	应用举例	118
6.5	习题	127
<b>第 7 章</b>	<b>函数</b>	<b>133</b>
7.1	函数的概念	133
7.1.1	函数的分类	133
7.1.2	函数的定义	134
7.2	函数的调用	134
7.2.1	函数调用的一般形式	134
7.2.2	函数的参数和返回值	135
7.2.3	函数声明	136
7.2.4	函数的嵌套调用	137
7.2.5	函数的递归调用	138
7.3	数组作为函数参数	139
7.4	变量的作用域和存储类别	140
7.4.1	变量的作用域	140
7.4.2	变量的存储类别	143
7.5	内部函数和外部函数	145
7.5.1	内部函数(静态函数)	145

7.5.2	外部函数	145
7.6	应用举例	145
7.7	习题	150
<b>第 8 章</b>	<b>指针</b>	<b>155</b>
8.1	指针概述	155
8.1.1	指针的概念	155
8.1.2	指针变量的定义	157
8.1.3	指针变量的引用	158
8.2	指针与数组	161
8.2.1	通过指针引用数组元素	163
8.2.2	指针与数组名	167
8.3	指针与字符串	170
8.4	指针与函数	173
8.4.1	指针作为函数参数	173
8.4.2	返回指针值的函数	177
8.4.3	指向函数的指针	179
8.5	多级指针	181
8.5.1	指向指针的指针	181
8.5.2	指针数组	183
8.5.3	指向多维数组行的指针	188
8.6	应用举例	191
8.7	指针小结	193
8.8	习题	194
<b>第 9 章</b>	<b>结构体与共用体</b>	<b>200</b>
9.1	结构体概述	200
9.1.1	结构体变量的定义和初始化	200
9.1.2	结构体成员的引用	204
9.2	结构体数组	204
9.3	指向结构体的指针	206
9.4	结构体与函数	210
9.5	链表	211
9.5.1	动态内存管理	211
9.5.2	链表概述	214
9.5.3	链表的相关操作	215
9.6	共用体概述	222
9.6.1	共用体变量的定义和初始化	222

9.6.2 共用体变量的引用	223
9.7 类型定义符 typedef	225
9.8 应用举例	227
9.9 习题	231
<b>第 10 章 文件</b>	<b>238</b>
10.1 文件概述	238
10.2 常用文件操作函数	239
10.2.1 文件的打开与关闭	239
10.2.2 文件的读/写	241
10.2.3 文件的定位	248
10.2.4 文件读/写中的检测	250
10.3 应用举例	250
10.4 习题	256
<b>第 11 章 单片机 C 语言编程实例</b>	<b>261</b>
11.1 功能要求	261
11.2 硬件线路	261
11.3 编程分解	263
11.3.1 充电控制	263
11.3.2 时钟中断	264
11.3.3 端口操作	265
11.4 单片机 C 语言编程小结	266
11.5 智能充电机完整源代码	267
<b>第 12 章 实验</b>	<b>279</b>
实验 1 熟悉 VC++ 6.0 环境及运行 C 程序	279
实验 2 数据类型、运算符和表达式	281
实验 3 顺序结构程序设计	282
实验 4 选择结构程序设计	285
实验 5 循环结构程序设计	288
实验 6 数组程序设计	291
实验 7 函数程序设计	294
实验 8 指针程序设计	298
实验 9 结构体与链表程序设计	302
实验 10 文件程序设计	307

附录.....	311
附录 A 常用字符与 ASCII 代码对照表 .....	311
附录 B C 语言的关键字 .....	312
附录 C 运算符和结合性 .....	312
附录 D C 库函数 .....	313
附录 E C 程序中常见错误分析 .....	318
参考文献.....	326

# C 语言程序设计概述

计算机处理问题是由程序来控制的。程序是人们根据解决问题的思路,利用某种程序设计语言编制的,用于指挥计算机进行一系列操作的指令集合。程序设计是指用某种程序设计语言编制计算机程序的过程。

C 语言是一种计算机程序设计语言,起源于 20 世纪 70 年代,最初用于编写 UNIX 操作系统,后来由于 C 语言强大的功能及可移植性,使得 C 语言迅速得到推广,并成为世界上应用最为广泛的程序设计语言。

本章通过几个简单的例子来认识 C 语言,了解 C 语言的结构特点、书写格式以及在 VC++ 6.0 集成环境下调试 C 语言程序的方法。

## 1.1 C 程序简介

为了说明 C 程序的结构特点以及书写格式,下面通过几个例子来认识 C 程序。

**【例 1-1】** 在屏幕上输出一行信息。

```
#include <stdio.h>           /* 编译预处理命令 */
void main()                 /* 主函数 */
{                           /* 函数体开始 */
    printf("This is my first C Program!\n"); /* 在屏幕上输出信息 */
}                           /* 函数体结束 */
```

运行结果为:

```
This is my first C Program!
```

程序说明:

(1) 程序的第 1 行是 C 语言中以“#”开头的编译预处理命令,以 .h 为扩展名的文件称为头文件,通过文件包含命令 #include 将头文件 stdio.h 包含进自己编写的 C 程序中。所谓头文件是系统内置的已经编写好的程序,用户通过文件包含命令实现对头文件的调用,头文件用尖括号 <> 或双引号 "" 括起来。此处的 stdio.h 是标准的输入/输出函数头文件。若要在程序中使用输入/输出函数,如该程序中使用了 printf() 输出函数,则必须在程序的开始处写上预处理命令 #include <stdio.h>。

(2) 程序的第 2~5 行是 C 程序的主函数 main() 的定义。在 C 语言程序中,必须有且只能有 1 个“main()”主函数,C 程序的执行总是从 main() 函数开始。从花括号

“{”开始,到花括号“}”结束的部分称为函数体。第 4 行的 printf()是输出函数,其功能是在屏幕上输出信息,双引号内的\n 表示换行,即在信息输出后,光标定位在下一行。最后的分号“;”是 C 程序语句结束的标记。

(3) 在 C 语言中,以“/\*”开头、“\*/”结束的内容是程序的注释,注释可以出现在程序的任何位置,用以帮助阅读和理解程序。运行程序时,注释部分将不被执行。

**【例 1-2】** 从键盘输入两个整数,求它们的乘积。

```
#include <stdio.h>
void main()
{
    int a,b,cj;                /* 定义 3 个整型变量 */
    printf("Please Input Two Integers:\n"); /* 在屏幕上输出提示信息 */
    scanf("a=%d,b=%d",&a,&b); /* 从键盘输入两个整数,分别放入变量 a 和 b 中 */
    cj=a*b;                    /* 将 a 和 b 的乘积赋给变量 cj */
    printf("cj=%d\n",cj);     /* 在屏幕上输出乘积 cj 的值 */
}
```

运行结果为:

```
Please Input Two Integers:
a=7,b=9
cj=63
```

程序说明:

(1) 程序的第 4 行是变量的定义语句。C 语言用变量来存放数据,此处用定义整型的符号 int 定义了 3 个变量 a、b 和 cj,表示这 3 个变量中可以存放整数。在定义变量时,多个变量之间以逗号“,”分隔。

(2) 程序的第 6 行是“scanf()”输入函数,其功能是从键盘输入两个整数并分别放入变量 a 和 b 中,输入方法是双引号中普通字符原样输入。%d 表示从键盘输入数据的类型是十进制整数;“&”是取地址符,表示从键盘输入的数放到 & 符号后面变量所对应的存储地址中。假如从键盘输入的两个数是 7 和 9,则此处 scanf()函数的输入方式为: a=7, b=9。

(3) 程序的第 7 行是赋值语句,“=”是赋值运算符,其功能是把“=”右边表达式 a \* b 的乘积值赋给左边的变量 cj。

(4) 程序的第 8 行功能是在屏幕上输出乘积 cj 的值,其输出格式是双引号中的普通字符原样输出。%d 表示输出数据的类型是十进制整数,乘积 cj 的值对应输出在 %d 的位置。

**【例 1-3】** 从键盘输入两个实数,比较它们的大小,在屏幕上输出其中较小的数。

```
#include "stdio.h"                /* 编译预处理命令 */
float min1(float x,float y);     /* 函数声明语句 */
void main()
{
    float a,b,min;                /* main()的函数体开始 */
    printf("Please Input a,b: "); /* 定义 3 个 float 实数类型的变量 */
                                /* 输出提示信息 */
}
```

```

scanf("%f,%f",&a,&b);          /* 从键盘输入两个数并分别存入变量 a 和 b * 中 /
min=min1(a,b);                /* 调用函数 min1(), 并将返回值赋给变量 min * /
printf("min=%f\n",min);      /* 输出结果 * /
}                               /* main() 的函数体结束 * /
float min1(float x,float y)   /* 定义函数 min1() * /
{                               /* min1() 的函数体开始 * /
    float z;                   /* 定义变量 z 是 float 实数类型 * /
    if(x<y)                    /* 条件判断语句, 判断 x 是否小于 y * /
        z=x;                   /* 如果 x<y, 则执行该行赋值语句 * /
    else                        /* 否则, 即 x 不小于 y, 则执行下一行赋值语句 * /
        z=y;
    return(z);                 /* 返回 z 的值 * /
}                               /* min1() 的函数体结束 * /

```

运行结果为:

```

Please Input a,b: 1.2,5.4 ✓
min=1.200000

```

程序说明:

(1) 本程序中定义了两个函数: main() 函数和 min1() 函数。其中 main() 函数是 C 语言程序必不可少的; min1() 函数是用户根据功能需求自己定义的, 称为用户自定义函数。这两个函数是相互平行的, 它们可以通过调用发生联系。所有被调用的函数都必须先定义后使用, main() 主函数不可以被其他函数调用。

(2) 程序的第 2 行是一条函数声明语句, 当被调用函数写在调用函数的后面时, 必须对被调用函数进行声明, 以使系统在编译时识别。

(3) 程序的第 11~19 行是用户自定义函数 min1(), 其功能是比较两个实数的大小, 并将其中较小的数返回调用该函数的语句。

## 1.2 C 程序的结构

通过前面 3 个例子的分析, 可以看出 C 语言程序的结构组成如下:

(1) C 程序由函数构成。一个 C 源程序可以由一个 main() 主函数和若干个用户自定义函数构成, 其中 main() 主函数必须有且只有一个。函数是 C 程序的基本单位。

(2) 函数由函数首部和函数体两部分组成。如例 1-3 中的用户自定义函数 min1(), 其中 float min1(float x, float y) 是函数首部, 包括函数名 min1、函数类型 float、参数类型 float 和参数 x 和 y。注意: 函数名后面的一对圆括号不可少。此函数中各项的含义为:

float	min1	(float	x,	float	y)
↓	↓	↓	↓	↓	↓
函数类型	函数名	参数类型	参数名	参数类型	参数名

函数体是函数首部后面一对大括号{}内的内容。函数体一般由两部分组成：对所使用的变量进行定义的说明部分和完成各种操作的执行部分。

(3) C 程序的执行总是从 main() 函数开始, 并到 main() 函数结束。main() 函数在整个程序中的位置可以任意。

(4) C 程序的语句以分号“;”结束。

(5) C 程序中包含注释, 以方便阅读, 在程序中用“/\* \*/”括起来, 注释不参与程序的执行。

## 1.3 C 语言的特点

程序设计语言一般可以分为机器语言、汇编语言和高级语言。

机器语言是计算机能够直接识别的语言, 其操作指令只能由二进制代码 0 和 1 构成。汇编语言是为方便记忆和编写程序, 用一些助记符表示二进制代码, 是一种与机器语言对应的符号化的语言。汇编语言编写的程序不能被计算机识别, 必须通过专门的汇编程序将符号转换成二进制代码才能执行。高级语言是 20 世纪 50 年代发展起来并使用人们习惯的自然语言编写的计算机程序, 这种计算机程序也不能直接识别和执行, 必须通过专门的编译程序将高级语言编写的程序转换成机器语言才能执行。其转换的方式有两种: 一种是解释方式, 即将高级语言编写的程序翻译一句执行一句; 另一种是编译方式, 即将高级语言编写的程序文件全部翻译成机器语言, 生成可执行文件后再执行。

C 语言是一种介于汇编语言和高级语言之间的程序设计语言, 它有以下特点。

(1) 程序结构简洁、紧凑

C 程序由若干函数构成, 各函数是相互独立的, 它们通过调用发生联系。C 语言是一种模块化程序设计语言, 共有 32 个关键字, 9 种控制语句, 程序书写形式自由。

(2) 表达能力强且应用灵活

C 语言运算符丰富, 共有 34 种, 可以组成各种类型的表达式以提高运算效率。C 语言数据类型丰富, 包括整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等, 能实现各种数据结构的运算, 从而可以满足不同的程序需求。

(3) 生成的目标程序质量好, 执行效率高

C 语言具有汇编语言的许多特性, 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, 仅比汇编语言生成的目标程序执行效率低 10%~20%。

(4) C 程序可移植性好

C 语言通过调用输入/输出函数实现输入/输出功能, 而这些函数是系统库函数, 因此 C 语言不依赖于计算机硬件系统, 从而便于在不同的计算机之间实现程序的移植。

由于 C 语言具有上述众多特点, 已经成为程序设计的主要语言之一, 被广泛应用于计算机的系统软件和应用软件的开发。

## 1.4 C程序的书写格式

为便于阅读和理解,C程序的书写一般遵循以下规则。

### (1) 一个语句占一行

虽然用C语言书写程序时较为自由,既可以一行写一个语句,也可以一行写多个语句,还可以一个语句分多行来写,但为了清晰,建议一个语句占一行。

### (2) 英文字母严格区分大小写

C程序中英文字母严格区分大小写,一般书写C程序时使用小写字母。C语言规定了32个有特定意义的单词,称为关键字,这些关键字在使用时必须是小写字母。

### (3) 采用缩进格式的书写方法

为了看清C程序的层次结构,便于阅读和理解程序,C程序一般都采用缩进格式的书写方法。缩进格式要求在书写程序时,不同结构层次的语句,从不同的起始位置开始;同一结构层次中的语句,缩进同样个数的字符位置。

### (4) 为了便于阅读和理解程序,在程序中适当添加注释信息

在编写程序时应力求遵循以上书写规则,以养成良好的编程习惯。

## 1.5 C程序的开发过程及 VC++ 6.0 集成开发环境

### 1.5.1 C程序的开发过程

一个C程序从编写到得到最终结果,其开发过程如图1-1所示。

#### 1. 编辑源程序

用高级语言编写的程序称为源程序。将C语言编写的源程序输入到编辑器,并保存为文件,后缀名为.c。

#### 2. 编译程序

编译程序就是将C源程序转换成机器语言程序。编译的作用是对源程序进行语法检查和句法检查,如果没有错误,则生成目标程序,文件后缀名是.obj。如果存在错误,编译系统会给出两种出错信息:一种是错误(error);一种是警告(warning)。凡是检查出error类的错误,就不能生成目标程序,必须改正后重新编译。

编译中没有出现错误,只能说明程序中没有语法和句法错误。

#### 3. 连接程序

使用系统的“连接程序”将目标文件与系统的库文件和系统提供的其他信息连接起来,生成可执行的二进制文件,后缀名是“.exe”。

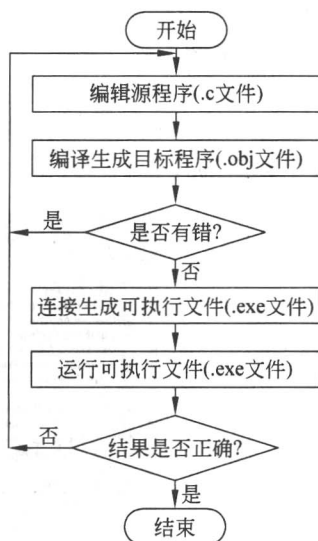


图 1-1 C程序的开发过程

#### 4. 运行程序

运行“.exe”可执行文件,得到运行结果。若运行结果不正确,则需检查并修改源程序。重复上述步骤,直到得到正确的运行结果为止。

### 1.5.2 VC++ 6.0 集成开发环境

VC++ 6.0 集成开发环境(Integrated Development Environment, IDE)是 Microsoft 公司出品的基于 Windows 环境的 C/C++ 开发工具,它是 Microsoft Visual Studio 套装软件的一个组成部分,其功能强大。此处仅介绍 C 源程序如何在 VC++ 6.0 集成开发环境中进行编译、连接和运行。

#### 1. VC++ 6.0 主窗口

单击“开始”|“所有程序”|Microsoft Visual Studio 6.0|Microsoft Visual C++ 6.0 命令,打开 VC++ 6.0 主窗口,如图 1-2 所示。

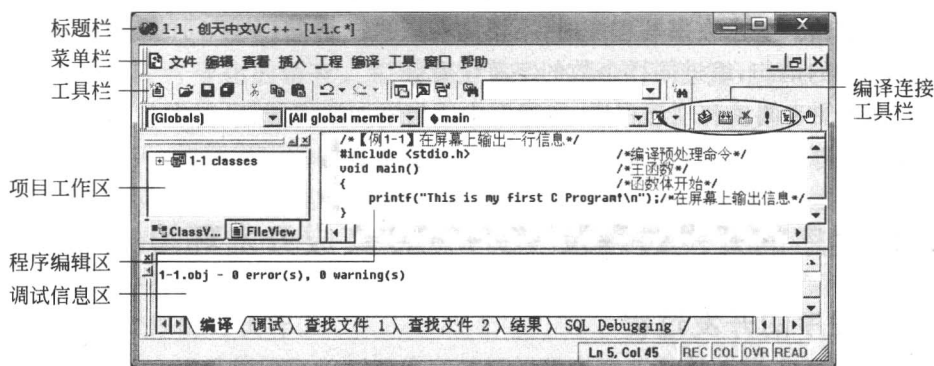


图 1-2 VC++ 6.0 主窗口

#### 2. 创建新工程

执行“文件”|“新建”命令,打开“新建”对话框,选择“工程”选项卡,如图 1-3 所示。从列表中选择 Win32 Console Application 项(Win32 控制台应用程序),在右侧的“工程”文本框中输入工程名(如:eg1),在“位置”文本框中设定该工程的保存路径,单击“确定”按钮。

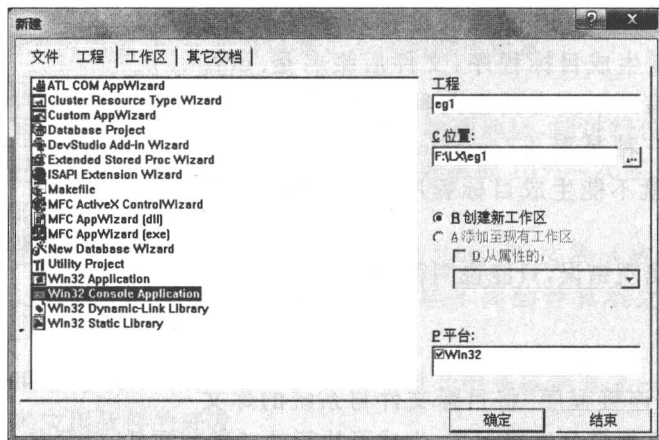


图 1-3 “新建”对话框中的“工程”选项卡