

高等院校信息技术规划教材

C 语言程序设计 案例教程

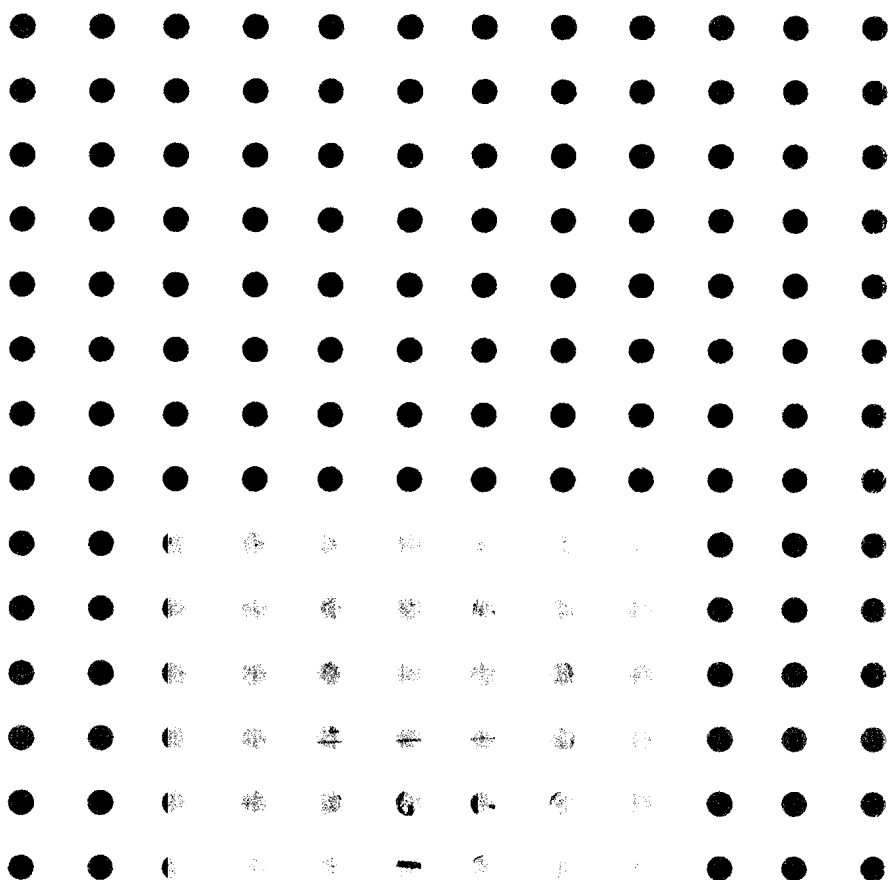
何薇 舒后 编著

清华大学出版社

高等院校信息技术规划教材

C 语言程序设计 案例教程

何薇 舒后 编著



清华大学出版社
北京

内 容 简 介

本书面向 C 语言程序设计的初学者,内容层次分明、由浅入深、简明易懂。

全书共分 10 章,分别为 C 语言概述、简单的 C 程序设计、顺序结构程序设计、分支结构程序设计、循环结构程序设计、数组、函数、指针、结构体、文件。

本书适合作为高等学校计算机及相关专业 C 语言程序设计课程的教材,也可作为 C 语言自学者的教材或参考书。

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 语言程序设计案例教程/何薇,舒后编著. —北京:清华大学出版社,2012.2

(高等院校信息技术规划教材)

ISBN 978-7-302-27900-6

I. ①C… II. ①何… ②舒… 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 008901 号

责任编辑:焦 虹

封面设计:傅瑞学

责任校对:白 蕾

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

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:185mm×260mm 印 张:13.5

字 数:311 千字

版 次:2012 年 2 月第 1 版

印 次:2012 年 2 月第 1 次印刷

印 数:1~3000

定 价:23.00 元

产品编号:037110-01

前言

foreword

对于 C 语言的初学者,必须通过大量的程序实例,由浅入深地从多读程序、多写程序、多调试程序入手,体会 C 语言的语法规则及程序设计的基本思想,才能掌握程序设计的基本方法。因此,本书是以程序案例为中心来组织教学内容的。

本书每章都是先从简单的实例入手,引出基本的知识点和 C 语言的语法规则,并在此基础上通过典型案例来进一步解释程序设计的方法,对关键程序段所完成的功能作进一步的分析和解释,以便于学习者体会程序设计的思想,并掌握实现设计思想的编程方法。

如何将要解决的问题用算法来描述,并通过 C 语言把它转化为程序,通常是初学者在学习程序设计的过程中遇到的问题。本书力求解决这一问题。书中的每个案例都从算法分析入手,通过流程图来表示算法,再将流程图转化为程序;最后通过程序分析,让读者了解并掌握程序设计思想的实现过程。在每一章的最后,总结了本章的重点概念和知识点,并给出了练习题。

C 语言以其丰富灵活的控制机制和数据结构、高效和简洁的语句表达以及可直接操纵计算机硬件的能力,在计算机应用开发中得到了最广泛的应用。因此,目前 C 语言程序设计作为一门计算机基础课程,已被各高等学校列为理工科学生的必修课程。我们希望通过努力使更多的学生认识到,学好 C 语言并不是一件难事。

本书在编写过程中得到了计算机系同仁的热情帮助,在此表示诚挚的谢意。由于编者水平有限,加之时间仓促,书中难免有错误和不足之处,敬请读者批评指正。

编者电子邮件地址:hewei@bigc.edu.cn

编 者

目录

Contents

第 1 章 C 语言概述	1
1.1 C 语言简介	1
1.2 如何编写 C 程序	1
1.2.1 简单的 C 语言程序设计	1
1.2.2 C 程序的编写格式和语法规则	2
1.3 如何编写和运行 C 程序	3
1.3.1 在 TC 环境下编写和运行 C 程序	3
1.3.2 在 VC 环境下编写和运行 C 程序	3
1.3.3 将 C 程序保存为文件	5
1.4 计算机执行 C 程序的过程	5
1.5 解释 C 程序的执行过程	5
1.6 小结	6
1.7 练习题	6
第 2 章 简单的 C 程序设计	7
2.1 只包含 printf() 的 C 程序	7
2.1.1 求半径为 20mm 的圆面积	7
2.1.2 在屏幕上显示出三角形图案	7
2.2 包含 scanf() 和 printf() 的 C 程序	8
2.2.1 求任意半径的圆面积	8
2.2.2 求任意两个整数之和	9
2.3 简单 C 程序的分析	10
2.3.1 C 程序结构	10
2.3.2 printf() 和 scanf()	10
2.3.3 变量定义语句	12
2.3.4 C 语言的算术表达式	14
2.4 小结	15



2.5 练习题	16
第3章 顺序结构程序设计	17
3.1 程序的算法与逻辑结构	17
3.1.1 流程图	18
3.1.2 求半径为 20mm 的圆面积的算法表示	18
3.1.3 求任意半径的圆面积的算法表示	18
3.1.4 求任意两个整数之和的算法表示	19
3.2 顺序结构程序设计	20
3.2.1 求任意三个整数之和的算法与程序分析	20
3.2.2 将小写字母 e 转换成大写字母 E	21
3.2.3 将输入的任意小写英文字母转换为大写字母	22
3.2.4 求 30 度所对应的弧度值	24
3.2.5 求 $\sin(30^\circ)$ 的值	25
3.2.6 求 10/3 的余数	25
3.3 小结	26
3.4 练习题	28
第4章 分支结构程序设计	30
4.1 分支的引入	30
4.1.1 根据重量计价	30
4.1.2 分支语句	31
4.2 if 语句与逻辑结构	32
4.2.1 if 语句的表现形式	32
4.2.2 在任意三个整数中找出最大数	34
4.2.3 求分段函数的值	35
4.2.4 判断成绩是否及格	36
4.2.5 判断是否为闰年	37
4.2.6 实现对学生成绩的评定	38
4.3 if 语句的嵌套	40
4.3.1 比较任意三个正整数的大小	40
4.3.2 用 if 嵌套实现对学生成绩的评定	44
4.4 多路分支 switch-case 语句	45
4.4.1 switch 语句的一般形式	45
4.4.2 switch 语句的逻辑结构图	45
4.4.3 根据输入的数字输出对应的英文单词	46
4.4.4 输入年份和月份,输出该月的天数	48

4.5	小结	50
4.6	练习题	52
第 5 章	循环结构程序设计	53
5.1	循环的引入	53
5.1.1	重复打印 5 行“*****”	53
5.1.2	循环语句	54
5.2	循环语句与逻辑结构	54
5.2.1	循环语句	54
5.2.2	循环语句的执行过程	55
5.3	使用三种循环语句编写程序	55
5.3.1	通过用户输入控制打印多行“*****”	55
5.3.2	用 do-while 循环改写程序	57
5.3.3	打印 n 行“*****”。	58
5.3.4	完成 1~10 的累加运算	59
5.3.5	完成 n! 运算	60
5.3.6	判断正整数 n 是否为素数	61
5.4	break 语句和 continue 语句	63
5.4.1	输入正整数并将偶数累加	64
5.4.2	输入以 '*' 结束的字符串,显示其中不是数字的字符	65
5.5	循环的嵌套	66
5.5.1	计算 $1!+2!+\cdots+10!$	66
5.5.2	求 100 以内的全部素数	68
5.5.3	输出 9×9 乘法表	70
5.5.4	输出数字金字塔	72
5.6	小结	74
5.7	练习题	76
第 6 章	数组	77
6.1	数组的引入	77
6.1.1	比较数组元素与简单变量	77
6.1.2	一维数组的定义和引用	80
6.1.3	产生 Fibonacci 数列的前 12 项	81
6.1.4	对任意输入的 n 个整数从小到大排序后输出	82
6.2	二维数组及其应用	84
6.2.1	矩阵的转置	84
6.2.2	二维数组的定义和引用	86



6.2.3	打印杨辉三角形	87
6.3	字符数组及其应用	89
6.3.1	输入字符串,统计其中大写字母、小写字母、数字及其他 字符的个数	89
6.3.2	字符数组与字符串处理函数	91
6.3.3	字符串逆序输出	94
6.3.4	简单的密码检测问题	95
6.4	小结	97
6.5	练习题	99
第7章	函数	100
7.1	自定义函数的引入	100
7.1.1	用户自定义函数 call_1()	100
7.1.2	函数	101
7.2	函数的基本概念	101
7.2.1	定义计算圆面积的函数 area()	101
7.2.2	函数的定义与调用	103
7.2.3	函数的形参与实参	104
7.2.4	无返回值的函数定义	105
7.2.5	输出数字金字塔的函数	105
7.2.6	函数声明	106
7.3	运用函数编写程序	107
7.3.1	定义判断奇偶数的函数	107
7.3.2	定义求两数中最大数的函数	108
7.3.3	求三个整数中的最大数	109
7.4	函数的嵌套调用	110
7.4.1	计算圆环的面积	110
7.4.2	计算 $S=1!+2!+3!+\cdots+20!$	112
7.5	函数的递归调用	114
7.5.1	了解递归的过程	114
7.5.2	逆序输出字符串	115
7.5.3	用递归法计算 $n!$	116
7.6	局部变量和全局变量	118
7.6.1	局部变量的作用域	118
7.6.2	全局变量的作用域	120
7.7	小结	121
7.8	练习题	122

第 8 章 指针	124
8.1 什么是指针	124
8.1.1 按从大到小的顺序输出任意输入的两个整数	124
8.1.2 指针、指针变量和指针运算符	125
8.1.3 通过指针变量访问整型变量	127
8.2 指针与数组	128
8.2.1 使用不同的数组元素引用方式输出数组的全部元素值	128
8.2.2 指针与一维数组	129
8.2.3 指针与二维数组	130
8.2.4 用行列指针输出一个二维数组的所有元素	133
8.2.5 用行指针实现求三个学生四门课成绩的最高分	134
8.2.6 指向字符串的指针	136
8.2.7 用指针进行字符串的复制	137
8.2.8 输出字符串中第 n 个字符后的所有字符	138
8.2.9 指针数组	139
8.2.10 用指针数组输出一个已知数组的所有元素	140
8.2.11 用指针数组,对若干门课程的课程名按升序进行排序	141
8.3 指针与函数	143
8.3.1 指针作为函数的参数实现两个数的交换	143
8.3.2 返回指针的函数	144
8.3.3 指向函数的指针	146
8.4 小结	147
8.5 练习题	151
第 9 章 结构体	152
9.1 结构体类型	152
9.1.1 简单的学生信息处理	152
9.1.2 结构体类型的定义	156
9.1.3 结构体变量的定义及初始化	157
9.1.4 结构体变量的引用	159
9.1.5 结构体数组	161
9.1.6 用结构体存入学生信息再按成绩排序输出	162
9.2 指向结构体的指针	164
9.2.1 利用指向结构体的指针实现简单的应用	164
9.2.2 指向结构体的指针(亦称结构指针)	165
9.3 结构体和函数	167



9.3.1	结构体类型的变量作为函数参数	167
9.3.2	结构体类型的变量作为函数返回值	168
9.4	typedef 类型定义	169
9.5	动态存储分配函数	170
9.5.1	动态分配函数的应用	170
9.5.2	动态存储分配函数的调用与功能	171
9.6	小结	172
9.7	练习题	175
第 10 章	文件	176
10.1	文件的概念	176
10.1.1	文件的引入	176
10.1.2	将输入的数据存入文件	176
10.1.3	文件的分类	178
10.1.4	文件操作	178
10.2	文件的打开与关闭	178
10.2.1	文件类型指针	178
10.2.2	文件的打开与关闭	179
10.3	文件的读写操作	181
10.3.1	将输入的字符写入文件	181
10.3.2	文件内容复制	183
10.3.3	输入字符串写入文件	184
10.3.4	输入整数写入文件	186
10.3.5	输入学生信息写入文件	188
10.3.6	使用数据块读写函数处理数据信息	190
10.3.7	使用数据块读写函数处理结构体类型数组	191
10.4	文件的随机读写	194
10.4.1	文件随机读写的相关函数	194
10.4.2	rewind()函数的应用	195
10.4.3	随机读取学生信息	196
10.4.4	追加学生信息	198
10.5	小结	200
10.6	练习题	203
参考文献		204

C 语言概述

本章重点

- C 语言是一种什么语言；
- 如何用 C 语言编写 C 程序；
- 如何执行 C 程序。

1.1 C 语言简介

C 语言是一种什么语言？C 语言是计算机语言。什么是计算机语言？就是能让计算机读懂的语言。用 C 语言写出来的“东西”叫 C 程序。C 程序中编写的内容是什么呢？是让计算机要完成的任务。把写好的 C 程序提交给计算机去执行，执行后的结果有三种可能：

- (1) 正确完成任务，得到需要的正确结果。
- (2) 完成任务，但不是你预期的结果。
- (3) 没有完成任务。

当出现第(2)、(3)种情况时，就说明我们写的 C 程序有错误，需要修改。这个过程通常叫做调试程序。用 C 语言编写程序的过程，通常叫做 C 语言程序设计。

1.2 如何编写 C 程序

简单地说，C 程序由一组语句构成，每一条语句完成某一功能。按照一定的逻辑结构安排好这些语句的顺序，编写出来的 C 程序就可提交给计算机去执行。

1.2.1 简单的 C 语言程序设计

例 1-1 在计算机屏幕上显示 C program design。

在 C 程序中只需一条语句：

```
printf("C program design ");
```

这条语句的功能是：在屏幕上输出 C program design。

提交给计算机执行时，却没有在屏幕上显示出 C program design，说明 C 程序有错误。计算机给出的错误信息如图 1-1 所示。

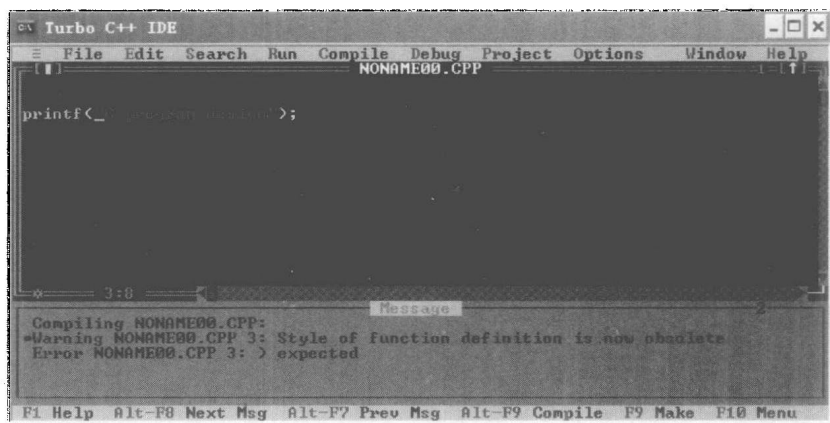


图 1-1 运行程序给出错误信息

错误信息告诉我们程序有语法错误。

任何一种语言都有它特定的语法规则和编写格式，C 语言也不例外。它有自己的语法规则、编写格式和字符集。

用 C 语言写出的程序一定要符合它的语法规则、编写格式和字符集要求，即使是再简单的 C 程序也是如此。

按照 C 语言的语法规则和编写格式，把上面的 C 程序改写为：

```
#include<stdio.h>
main()
{
    printf("C program design");
}
```

提交给计算机执行后，屏幕上显示出 C program design，运行结果如图 1-2 所示。

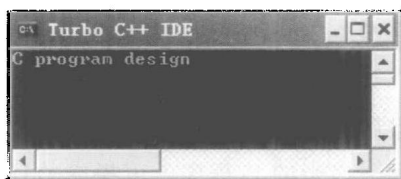


图 1-2 屏幕上显示出 C program design

1.2.2 C 程序的编写格式和语法规测

在上面的例子中，C 程序的编写格式包括两部分。

1. 头文件包含

```
#include<stdio.h>
```

2. 主函数

```
main()
{
    printf("C program design");
}
```

- 一组大括号{ }中的内容为主函数体,在主函数体中包含了一条语句:

```
printf("C program design ");
```

- 语句要以分号“;”结束。

1.3 如何编写和运行 C 程序

1.3.1 在 TC 环境下编写和运行 C 程序

打开 TC 软件,在 TC 环境中,编写 C 程序,如图 1-3 所示。

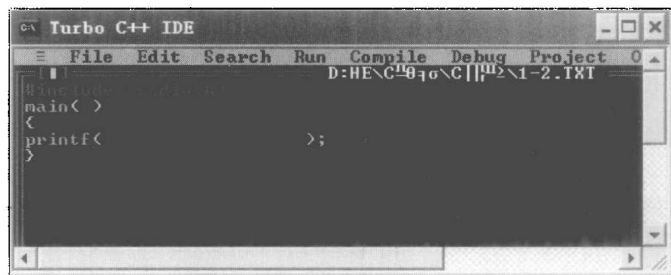


图 1-3 在 TC 环境中编写 C 程序

运行该程序,可得到程序的运行结果,如图 1-2 所示。

1.3.2 在 VC 环境下编写和运行 C 程序

打开 Microsoft Studio 软件,在 VC 环境中编写 C 程序的步骤如下。

- (1) 打开 VC 后,工作界面如图 1-4 所示,选择“新建”。
- (2) 选择“文件”选项卡中的 C++ Source File 选项,选择目录和输入文件名后,单击“确定”按钮,如图 1-5 所示。
- (3) 输入 C 程序后,选择工具栏中的“编译”图标,编译后的结果会出现在信息栏中,产生了一个 .obj 文件。
- (4) 选择工具栏中的“运行”图标,就可运行程序。程序运行结果如图 1-6 所示。

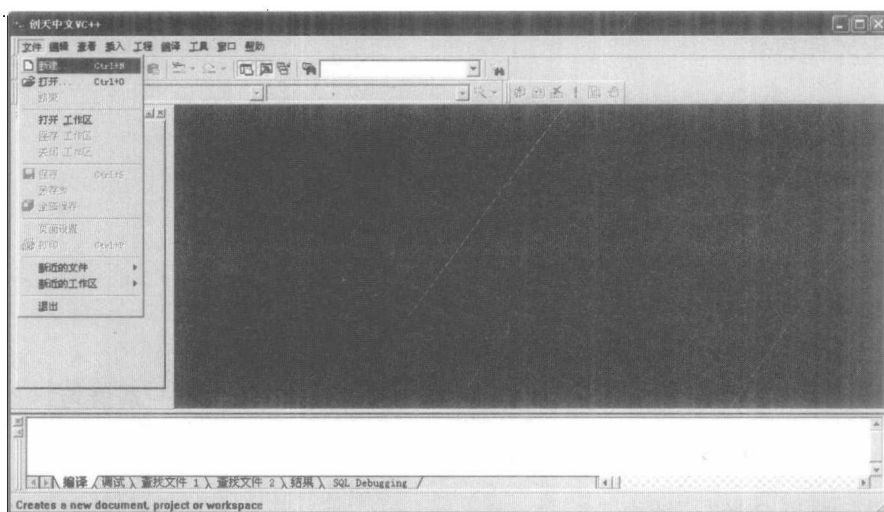


图 1-4 VC 工作界面

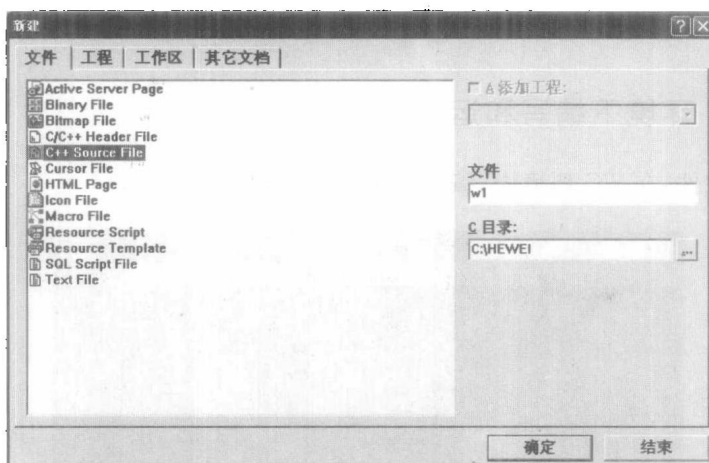


图 1-5 VC 环境中建立 C 程序

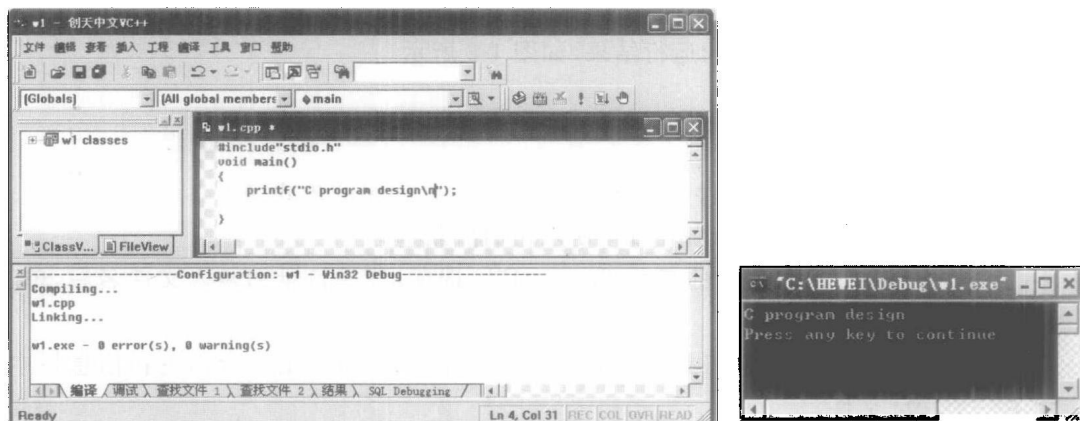


图 1-6 VC 环境中得到程序的运行结果

1.3.3 将 C 程序保存为文件

无论在 TC 环境还是在 VC 环境下编写的 C 程序,都称为 C 源程序,都可以作为文件永久保存。保存后的文件将带有扩展名“.c”。带有扩展名.c 的文件称之为 C 程序的源文件,也叫 C 源程序文件。

1.4 计算机执行 C 程序的过程

计算机只能识别二进制的 0、1 代码,那么计算机又是如何执行 C 源程序的呢?在 TC 环境或 VC 环境下打开一个正确的 C 源程序文件,运行该程序并获得结果时,事实上这一过程蕴涵了两个步骤:编译和链接。

(1) 编译:把 C 源程序转换为 0、1 代码的目标程序,由此产生一个带有.obj 扩展名的目标程序文件。

(2) 链接:目标程序是不能直接执行的二进制程序。链接就是将目标程序、库函数或其他目标程序链接装配成一个可执行的程序,由此产生一个带有.exe 扩展名的可运行程序文件。

整个过程中,共拥有 3 个文件:带有.cpp 扩展名的 C 源程序文件,带有.obj 扩展名的 C 目标程序文件,带有.exe 扩展名的 C 可执行文件。计算机执行 C 程序的过程可以用图 1-7 表示。

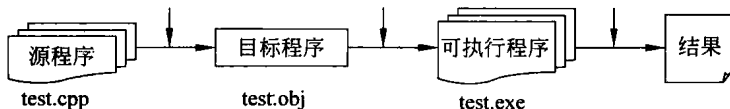


图 1-7 计算机执行 C 程序的过程

1.5 解释 C 程序的执行过程

程序的第一行是:

```
#include<stdio.h>
```

符号#后面紧跟关键字include,用来告诉编译器在当前程序编译时还要包括另一个文件,被包括的文件名是stdio.h,文件名用尖括号(或双引号)括起来。stdio.h 含有 C 中标准输入输出设备,在需要执行输入输出语句的程序中,都必须包含这个文件。我们的程序有输出语句,所以包含了这个文件。

第二行是:

```
main()
```



这是每个 C 程序的入口,必须存在。程序体通常用左花括号和右花括号括起来,表示程序的打开和关闭。如下面的语句:

```
{  
    程序体  
}
```

下一行:

```
printf("C program design");
```

调用了 printf(),用来向标准输出设备输出一个字符串。通常认为终端显示器就是标准输出设备,因此在这个例子中,C program design 就显示在屏幕上。

1.6 小 结

本章讲述了何为 C 程序,如何编写、编译和运行 C 程序。C 程序中值得注意几点:

- (1) 包含文件必须存在。
- (2) main()是程序入口,必须存在。
- (3) 程序体在一对花括号{}中。
- (4) 除了 include、main()和花括号之外的每条语句都必须以分号结束。

1.7 练 习 题

1. 熟悉 TC 和 VC 的 C 程序编写环境,并输入以下 C 程序,观察运行结果。

```
#include<stdio.h>  
main()  
{  
    printf("*****");  
    printf("*****");  
    printf("*****");  
}
```

2. 输入以下 C 程序,观察编译后的提示信息与第 1 题有什么不同,从中体会 void 的意义。

```
#include<stdio.h>  
void main()  
{  
    printf("*****");  
    printf("*****");  
    printf("*****");  
}
```

3. 在输入的 C 程序中去掉语句后面的分号,观察编译后的提示信息。

简单的 C 程序设计

本章重点

- printf()格式输出函数的使用;
- scanf()格式输入函数的使用;
- 变量定义语句的使用;
- 算数表达式。

简单地说,C 程序由一组语句构成,每一条语句完成某一功能。按照一定的逻辑结构去编排好这些语句的顺序,书写出来的 C 程序就可提交给计算机执行。

计算机在执行程序的过程中,会按照事先编排好的语句顺序依次执行。因此程序设计的关键就是编排这些语句的顺序,并牢记 C 语言的语法规则、书写格式和字符集。

2.1 只包含 printf()的 C 程序

printf()是输出语句,是任何程序都不能缺少的语句。C 语言中使用最频繁的输出语句就是 printf()。

2.1.1 求半径为 20mm 的圆面积

例 2-1 求半径为 20mm 的圆面积。

```
#include<stdio.h>
void main()
{
    printf("circle area=%f\n",20*20*3.14);
}
```

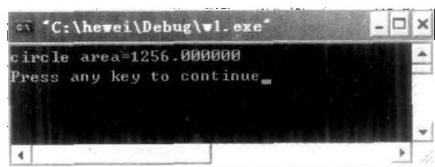


图 2-1 半径为 20mm 的圆面积

程序运行结果如图 2-1 所示。

2.1.2 在屏幕上显示出三角形图案

例 2-2 在屏幕上显示出由 * 号组成的三角形图案。