

C 语言程序设计系列

吕凤翥 编著

C语言程序设计

教师使用参考书

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

清华大学出版社



C 语 言 程 序 设 计 系 列

吕凤翥 编著

C 语 言 程 序 设 计

教师使用参考书

清华大学出版社
北京

内 容 简 介

本书是 C 语言程序设计系列丛书的一部分, 专门针对教员编写, 主要用于指导教员教好 C 语言程序设计这门课程。书中介绍了作者讲授这门课的体会, 指导教员的具体教学, 应重点讲解的内容与概念, 特别是应该让学生掌握的语法和常用算法, 还提供了典型示例用于学习和分析。本书按照教材的顺序对应分为 10 章, 提供了上机练习题及答案。

本书适合作为高等学校中 C 语言课程的教学参考书。

版权所有, 翻印必究。举报电话: 010-62782989 13501256678 13801310933

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。

本书防伪标签采用特殊防伪技术, 用户可通过在图案表面涂抹清水, 图案消失, 水干后图案复现; 或将表面膜揭下, 放在白纸上用彩笔涂抹, 图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

C 语言程序设计教师使用参考书 / 吕凤翥编著. —北京: 清华大学出版社, 2006.6

(C 语言程序设计系列)

ISBN 7-302-12846-4

I. C… II. 吕… III. C 语言-程序设计-高等学校-教学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 033468 号

出 版 者: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

社 总 机: 010-62770175 客户服务: 010-62776969

组稿编辑: 丁 岭

文稿编辑: 孙建春

印 装 者: 北京市昌平环球印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 185×260 印张: 10.25 字数: 254 千字

版 次: 2006 年 6 月第 1 版 2006 年 6 月第 1 次印刷

书 号: ISBN 7-302-12846-4 / TP · 8171

印 数: 1 ~ 3000

定 价: 16.00 元

关于这本书

(代前言)

写这本书的原因

1. 作者本人有能力写这本书

作者自 20 世纪 80 年代初便开始讲 C 语言, 至今已有 20 多年了, 先后给本科生、大专生、中专生和高中生以及自学考试的考生讲过 100 多次。接触到了各种不同层次的学生, 他们都提出过不同的问题, 学习后有过各种感受。作者还针对着不同层次的 C 语言学习者写过各种不同层次的 C 语言教材共 6、7 种。可以说, 作者对讲授 C 语言课程有丰富的经验, 回答过千奇百怪的问题, 经历过各种各样的考验, 因此, 有能力写好这本书。

2. 社会上有对这种书的需求

随着高等、中等教育事业的发展, 计算机专业不断扩招, 讲授 C 语言课程的教员越来越多, 他们中间有许多人是初讲者, 缺乏讲授 C 语言课程的经验, 特别需要在讲授经验上得到帮助。近些年来, 也曾有过一些年轻教员与作者一起讨论关于 C 语言的教学问题。作者也尽力将自己的一些体会告诉初次讲授 C 语言课的教员, 帮助他们提高教学水平。因此, 作者一直想寻找机会将多年来讲授 C 语言的经验和体会让更多人知道, 并能从吸取一点帮助。写这本书便提供这样一个窗口。

3. 读过这本书能有一点帮助和启发便给作者最大的欣慰

作者已年过花甲, 回顾以前走过的路, 可以说一辈子与书结缘。概括有生之年所干过的事, 可归纳为 6 个字: 读书、教书、写书。念过 18 年的书, 又教过 36 年的书, 还写过 30 多种计算机方面的书。

读过作者的书的读者, 如果能从书中得到一些帮助和启发, 便给予作者最大的欣慰, 这也是写这本书的原因和目的。

如何阅读这本书

这本书与《C 语言程序设计——基础理论与案例》和《C 语言程序设计习题解答和上机指导》一起作为 C 语言程序设计系列的教材。本书是针对教员而写的, 希望能够通过这本书指导教员教好 C 语言程序设计这门课程。书中传授了作者讲授这门课的体会, 并根据

这些体会告诉教员应该讲些什么，应该重点讲解哪些内容，有哪些概念是要讲清楚的，有哪些语法和常用算法是要让学生掌握的，还有哪些典型例子可供学习和分析。因此，本书按讲课的教材对应分为 10 章，每章由如下各节组成：

- (1) 本章讲述的内容；
- (2) 重点内容；
- (3) 重要概念；
- (4) 常用语法和算法；
- (5) 典型例子；
- (6) 练习题。

读者可以有针对性地去阅读每章的有关章节。

阅读本书时一定要与《C 语言程序设计——基础理论和案例》一书结合起来，为了避免重复，有些内容在本书中便省略了。

本书每章后面又备有练习题，可以选用一些作为学生的作业，也可以选用一些作讲课时的例题。

这本书将告诉读者如何教好 C 语言

给读者带来一些帮助和启发，这是作者的最大心愿。作者以前写过一些书深受读者欢迎，这是激励作者继续写书的动力。一个人做一件对别人有用的事，是值得庆贺的。

作者写这本书所贯穿的思想，也是自己教好 C 语言这门课程的体会，总来说，就是“三讲”、“三要”。

1. 重点内容详细讲

C 语言课程教材分为 10 章，每章都有一些要讲的内容，在这些内容中又有重点内容。讲课时一定要抓住重点，细致讲解，讲清楚讲明白。例如，在第 8 章“函数”章中，重点内容是函数的调用。要讲清楚函数的调用目的、调用过程和实现调用的方法。让学生明确函数调用要干什么，怎样做。像这样的重点内容每章都有，不止一个，都要详细地给学生讲清楚，为此教员自己必须先下工夫搞清楚。

2. 重要概念反复讲

C 语言是一种比较难学的高级语言。它之所以难学，其中有一个原因就是概念多、规则多，这些概念要理解，这些规定要记住。每章都要有许多概念，其中有一些是重要概念。例如，在第 2 章“单词、常量和变量”中，给变量赋初值和给变量赋值就是两个重要的概念，有的人学完了 C 语言后，对赋初值和赋值这两个概念还搞不清楚。因此，讲解变量时一定要把给变量赋初值和赋值的时间、方法和作用讲清楚，且在以后的例题中，在其他与此相关的地方还要反复讲。例如，在讲函数调用时，再次指出将函数实参传递给形参实际上是实参给形参赋值，而不是赋值。总之，一些重要的概念，仅讲一次是不够的，要抓住机会、创造条件多讲几次。

3. 典型例子深入讲

讲课中要多讲例题，要选择一些有代表性的典型例题给学生讲清楚，在讲清楚例题中的语法现象和功能后，还要再改变一些语句，讲解改变前后的不同，并且让学生上机调试，看看改变某些语句后的结果。这样经过深入讲解例题，才能使学生深入了解一些概念和技巧。作者编写的教材中，尽量选用典型例题，把这些例题搞清楚，学生便可以照猫画虎地进行编程练习，逐渐学会编程方法，不断提高编程技能。学会编程是从学会分析例题开始的。

4. 教学内容要少而精

由于讲课的时间有限，讲课时要对内容进行选择，挑那些难以理解的内容进行讲解，一些比较简单的内容留给学生自学。一堂课安排讲的内容不要太多，要留出一些给学生思考的时间，因此就要求教员的授课内容要少而精。教员应懂得一个道理：灌多了，学生得到的反而小；有重点地讲，学生反而记住的多。没有经验的教员往往是满堂灌，将一章的内容面面俱到，不停地讲了一节课，学生却什么也没记住，这样的实际效果并不好。

5. 能力培养要贯穿始终

让学生在课堂上学到一些知识点、一些具体的规则、规定、格式和方法固然很重要，但是更重要的是，要教学生学会解决问题分析问题的方法。在讲述一个概念时，不仅要讲述“是什么”，而且要讲清“为什么”，往往从“为什么”中可以获得思考问题的方法。例如，讲数组概念时首先要讲述什么是数组，指出它是怎样的一种数据类型，还要讲清楚数组这种类型有什么作用，什么时候用数组。总之，在讲课的过程中，一方面让学生提问题，另一方面教员也不断地就某个概念或内容提出问题，让学生明白为什么要这样，能不能那样。教员要帮助学生爱思考，培养学生爱思考，通过不断的思考来培养能力。

6. 上机训练要反复强调

学习计算机语言课是离不开上机调试的。在教学环节中，应该把上机实践放在一个重要的位置。作者在这套教材中，专门为每章编写了上机要求和上机练习题，并在许多例题中提到了如何上机调试。要求学生保证一定的上机时间，还要教会学生通过上机来回答所提出的疑难问题。例如，变量的值可分为有意义的、无意义的，有意义的变量值又分为初值、默认值等等，这些都可以通过上机来验证，搞清楚在什么情况下会出现哪种值。为此，本书的练习题部分专门提出了许多要上机验证的问题。通过这些例子让学生学会自己提问题，自己上机验证，回答所提出的问题，这也是对学生能力的培养。在课堂上要告诉学生：不上机是学不好 C 语言的。

0 在 C 语言中的重要性

下面列举出一些关于 0 在 C 语言中的重要用途，以引起人们对 0 的重视。

(1) C 语言中数组下标是从 0 开始的。

- (2) C 语言中字符串的结束符是 ASCII 码为 0 的字符, 表示为 '\0'。
- (3) 枚举类型的定义中, 枚举表中枚举符默认整型数值是从 0 开始的。
- (4) 打开文件失败时, 打开文件函数 (`fopen()`) 返回值为 0, 通常用它来判断文件是否打开。
- (5) C 语言系统定义的符号常量 `NULL` 的值为 0。

本书中的一些观点和提法愿与同行讨论研究。由于水平有限, 难免存在错误和不足之处, 诚心欢迎读者提出批评和意见。谢谢关注本书的所有读者。

作 者

于 2006 年 5 月写于燕北园

目 录

第 1 章 C 语言概述	1
1.1 本章讲述的内容	1
1.2 重点内容	1
1.2.1 C 语言的特点	1
1.2.2 C 语言程序在结构上的特点	4
1.2.3 C 语言程序的实现	4
1.3 重要概念	6
1.4 常用语法和算法	7
1.4.1 键盘输入操作方法	7
1.4.2 屏幕输出操作方法	8
1.5 典型例子	9
练习题	12
第 2 章 单词、常量和变量	15
2.1 本章讲述的内容	15
2.2 重点内容	15
2.2.1 类型	15
2.2.2 字符常量和字符串常量	16
2.2.3 全面认识变量	17
2.3 重要概念	18
2.4 常用的语法和算法	20
2.4.1 常量	20
2.4.2 变量	21
2.5 典型例子	22
练习题	24
第 3 章 运算符和表达式	28
3.1 本章讲述的内容	28
3.2 重点内容	29
3.2.1 具有副作用的运算符	29
3.2.2 运算符的优先级和结合性	31
3.2.3 表达式的值和类型	32
3.3 重要概念	33

3.4 常用语法和算法	35
3.4.1 运算符和表达式	35
3.4.2 类型	37
3.5 典型例子	37
练习题	41
第4章 预处理功能	44
4.1 本章讲述的内容	44
4.2 重点内容	46
4.2.1 两种不同格式的宏定义命令	46
4.2.2 文件包含的使用	49
4.3 重要概念	50
4.4 常用语法和算法	51
4.4.1 预处理命令	51
4.4.2 类型定义	52
4.5 典型例子	53
练习题	56
第5章 语句	59
5.1 本章讲述的内容	59
5.2 重点内容	60
5.2.1 选择语句	60
5.2.2 循环语句	62
5.3 重要概念	64
5.4 常用语法和算法	66
5.4.1 本章基本语法要求	66
5.4.2 常用算法	66
5.5 典型例子	67
练习题	73
第6章 数组	77
6.1 本章讲述的内容	77
6.2 重点内容	78
6.2.1 数组的定义	78
6.2.2 数组的初始化	79
6.2.3 字符数组	79
6.3 重要概念	80
6.4 基本语法和基本算法	81
6.4.1 本章基本语法	81

6.4.2 基本算法	83
6.5 典型例子	83
练习题	86
第7章 指针	90
7.1 本章讲述的内容	90
7.1.1 指针的概念	90
7.1.2 指针的应用	91
7.2 重点内容	91
7.2.1 指针的概念	91
7.2.2 指针在数组方面的应用	92
7.3 重要概念	95
7.4 重要的语法和算法	100
7.5 典型例子	102
练习题	105
第8章 函数	108
8.1 本章讲述的主要内容	108
8.2 重点内容	108
8.2.1 函数的定义和函数的说明	108
8.2.2 函数的返回值	109
8.2.3 函数的调用方式	110
8.3 重要概念	111
8.4 常用语法及算法	113
8.4.1 常用语法	113
8.4.2 主要算法	113
8.5 典型例子	114
练习题	119
第9章 结构和联合	123
9.1 本章讲述的主要内容	123
9.2 重点内容	123
9.2.1 结构变量	123
9.2.2 结构数组	126
9.2.3 联合成员的共址性	127
9.3 重要概念	129
9.4 常用语法和算法	130
9.4.1 常用语法	130
9.4.2 常用算法	131

9.5 典型例题	132
练习题	136
第 10 章 文件操作	139
10.1 本章讲述的内容	139
10.2 重点内容	140
10.2.1 标准文件的读写操作	141
10.2.2 一般文件的操作	142
10.3 重要概念	145
10.4 常用语法	146
10.5 典型例子	147
练习题	151

第1章 C语言概述

1.1 本章讲述的内容

本章讲述如下内容：

- (1) C语言的由来和发展。
- (2) C语言的特点和应用。
- (3) C语言程序在结构上的特点和书写格式上的要求。
- (4) C语言程序中出现的键盘输入操作和屏幕输出操作的方法。
- (5) 使用 Visual C++ 6.0 编译系统来实现的单文件程序和多文件程序的操作步骤。

本章是本书的第1章，通过讲述本章内容，实际上是给学生回答下述一些问题：

- (1) 什么是C语言？它为什么使用较为普遍？
- (2) C语言与其他语言有何不同？
- (3) C语言程序是什么样子的？在结构上有哪些特点？
- (4) 编写的C语言程序如何获得其输出结果？如何学好C语言？

通过对本章内容的讲述让学生得到如下收获：

- (1) 激发学生学习C语言的兴趣，使学生一开始就下定决心学好C语言编程。
- (2) 增强学生学好C语言的信心。通过课堂上认真听讲，课后看书复习完成作业，加强上机实践，存在的问题及时解决，并不难学好这门功课。
- (3) 掌握好上机实践这一重要环节。从一开始就要重视上机实践，学会通过上机实践来解决疑难问题，培养自学能力和动手能力。

1.2 重点内容

本章的重点内容有如下3点。

1.2.1 C语言的特点

讲述C语言的特点时，一定要一分为二：既要讲述C语言吸取其他语言的经验和教训，使C语言超越其他语言特点，这些特点作为C语言的优点受到许多编程者的赞许，又要讲述C语言由于过于强调它的某些特点而带来的一些不足，或者说是“隐患”，而被后来的C++语言进行了改进。总之，讲述C语言要实事求是，让学生对C语言有个全面的正确的认识，从而会对任何事物总要用一分为二的观点来认识。

1. C 语言的优点

讲述 C 语言特点时, 主要讲述它的优点, 即它吸引编程者的魅力所在。

(1) C 语言语句简练、书写方便

在这一点上, 很多语言难以与 C 语言相比, 主要表现在以下几个方面。

- 使用缩写的类型说明符, 使得书写方便。特别是对最常用的类型, 如整型用 `int`, 字符型 `char`, 只用英文单词的前边一部分, 而不像有的语言使用全部英文单词。
- 运算符比较多, 有的运算符功能很强。运算符大多可将许多操作使用表达式来表示, 而不必用函数实现, 这样做不仅书写简单, 而且可读性也好。C 语言中引进一个三目运算符, 这是其他语言少有的, 这个三目运算符功能强, 具有简单条件语句 (`if-else` 语句) 的功能。自然, 使用条件表达式语句比用条件语句简练。
- 采用花括号作定界符, 比用语句或其他方法更为方便。C 语言中, 函数体的定界符、条件语句的 `if` 体、`else if` 体和 `else` 体的定界符以及循环语句的循环体的定界符都用一对花括号 (`{}`), 这样做既简练, 又方便。这比有些语言使用 `begin` 和 `end` 等标识符要简单得多。特别是又规定了如果 `if` 体、循环体是一条语句就不用定界符了。
- 讲到书写方便, 不得不讲讲预处理功能。特别是该功能的文件包含命令, 将多个程序中相重复的部分提取出来, 写在一个被包含的文件中, 只要在需要这部分内容的程序中包含该文件就可以了, 而不必重复写一遍。

此外, 符号常量的引入也在某种程度为书写带来一定方便。

(2) C 语言的功能特强

C 语言功能强主要表现在它既具有高级语言的特性, 又具有低级语言的特性。有人说它是一种中级语言, 但是这种讲法反映不出它的功能之强大。

C 语言功能强表现在如下几个方面。

- C 语言具有高级语言的所有功能, 此外, 在字符处理方面采用了字符指针等措施, 更加便于字符处理。
- C 语言具有地址操作和位操作以及寄存器操作等功能, 这是功能是一般高级语言所不具有的, 这是兼容了汇编语言的功能。
- C 语言引进指针的概念, 为地址操作带来了方便, 同时指针又为数组元素的表示及作为函数的参数带来了操作上的方便和功能上的增强。例如, 传址调用可实现函数之间信息的传递等。
- C 语言还具有丰富的数据类型和存储类。在数据类型方面, 除了基本数据类型外, 还提供了数组、结构、联合和枚举等构造类型, 使 C 语言程序可以方便地实现各种复杂数据类型的操作, 多种的存储类也为 C 语言适应不同情况的需要提供方便, 进一步增强了该语言的功能。

(3) C 语言是一种结构化的程序设计语言

结构化程序设计语言的出现是编程史上的一个里程碑。结构化程序设计语言要具备构成结构化程序设计的三种基本结构模式的语句。这三种基本结构模式分别是:

- 顺序结构模式。

- 分支结构模式。
- 循环结构模式。

C语言具有构成上述三种结构模式的语句，所以C语言是人们比较喜欢的一种结构化程序设计语言，因为它具有前边讲述的两大特点。

但是，还应该看到，C语言是一种不完全的结构化程序设计语言。它的不完全性表现在如下两点：

- C语言允许使用 goto 语句，但是对它的 goto 范围进行限制只允许在函数体内进行转向。因此，C语言中最小的模块是函数。
- 完全的结构化程序设计语言要求一个模块只有一个入口和一个出口。不过，C语言中函数作为最小模块，它虽然有一个入口，但是可以有两个或两个以上的出口。因为函数内允许有多个 return 语句。

(4) C语言的移植性好

移植性好是C语言的另一个优点，这也是C语言能够广泛用于各种型号机器上去的原因。C语言移植性好的原因是因为该语言的编译系统较小，再加上它具有若干预处理命令，可在编译前就处理一些命令，这都给C语言的移植带来了方便。

2. C语言的不足

C语言过于强调它的某些特点，即灵活和简洁，也由此带来了一些不足。

(1) 运算符过多，优先级较繁杂，使用者觉得难用难记

运算符共有44个，优先级有15种，结合性还有两类。这无疑对数据运算带来方便，但是，对编程者带来了难记和难用的不便。例如，运算符星号(*)具有两个功能，作为单目运算符表示取内容，作为双目运算符表示相乘运算等等。

(2) C语言是一种弱类型的语言

C语言为了类型转换的方便，对类型的要求很不严格，在许多情况下不做类型检查。在有些情况下，C语言要求类型一致，由于不做类型检查，出现不一致时也不报错，结果造成了运算结果的错误。因此，在编程中对类型处理一定要慎重，应尽量避免因类型不一致造成的差错。

(3) 对数组进行动态赋值时不做越界检查

这样，当给某个数组进行动态赋值时，若赋值个数超过了数组元素的个数，就可能造成数据上的混乱。

指针的使用上也要特别注意，只有给指针赋以确定值时才可以使用。由于字符指针被赋值后它所指向新的字符串大小被确定，当用该指针再指向新的字符串时，要注意字符串长度可能引起的越界。因此，在有些情况下，使用字符指针不合适时，应使用字符数组。

(4) 产生二义性

C语言出于优化的考虑，允许不同的编译系统对表达式或参数表中的操作数或数据项的计算顺序有所不同，即有的编译系统规定计算顺序从左至右，有的编译系统规定计算顺序从右至左。这对于有副作用运算符组成的操作数或数据项来讲，有可能产生二义性，编程时应尽量避免。

1.2.2 C 语言程序在结构上的特点

从 C 语言的具体程序实例中可以看到, C 语言程序在结构上的特点如下:

- C 语言程序是由一个或多个文件组成的。
- 组成 C 语言程序的每个文件是由一个或多个函数组成的。因此, 可以说 C 语言程序是函数串, 即 C 语言程序是由放在不同文件中的若干个函数组成的, 函数是组成 C 语言程序的最小模块。
- 组成 C 语言程序的若干个函数中, 有且仅有一个主函数 `main()`。主函数所在的文件称为主文件。执行一个 C 语言程序时, 先从主函数开始执行, 程序中其他的函数都是被主函数和主函数来调用的函数来调用。程序中若干个函数先后顺序基本无关。
- 函数是由函数头和函数体组成。函数头给出了函数名字、函数类型、函数参数的个数和类型。函数体由一对花括号括起的若干条语句组成。语句是由若干个单词组成的。

另外, 由于 C 语言程序的可读性较差, 因此, 在书写上要求遵循着如下的约定。

- 一行通常写一条语句。有些语句可以写成多行。分行书写时不得将一个单词分开。
- 每条语句末尾加分号 (;), 不是语句结束不应加分号。
- 花括号的书写按规定的格式。本书规定: 每个括号占一行, 左右花括号都与使用它们的语句对齐, 花括号内语句向右缩进。
- 语句书写时要按约定进行缩进, 以便提高可读性。
- 适当使用注释信息, 可提高可读性。

1.2.3 C 语言程序的实现

C 语言程序的实现指的是如何使编写的 C 语言程序获得正确的输出结果。

C 语言程序的实现方法与其他高级语言的实现方法相同, 通常都是通过编辑、编译和运行三个步骤来实现。

本书选用了 Visual C++ 6.0 集成工具软件系统来实现 C 语言程序。这是一个 C++ 语言的编译系统, 也能用来编译 C 语言的程序。使用该系统实现 C 语言程序时, 与使用 TC2.0AK 3.0 系统等其他 C 语言编译系统有如下几点不同。

- 每个 C 语言程序文件存盘时应加 .C 后缀, 不应默认扩展名。
- 每个 C 语言程序开头都加如下命令:

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

- C 程序中所有函数都不可省略类型说明, 无返回值的函数加 `void`, 返回值为 `int` 型的函数加 `int`。
- 函数说明应用原型, 即在简单说明的基础上加上函数参数和个数的说明。
- 定义函数时, 函数参数类型说明放在圆括号内, 而不再另起一行进行说明。

上述几点都是 C++ 语言程序所要求的。

一开始就让学生学会在 Visual C++ 6.0 系统上运行 C 语言程序的方法。下面分别讲述单文件程序和多文件程序的实现方法。

1. 单文件 C 语言程序的实现方法

单文件 C 语言程序的实现方法如下。

(1) 运行 Visual C++ 6.0 系统, 出现该系统的主窗口。

(2) 选择菜单栏中的 File 菜单项, 在下拉菜单中选择 New 选项, 出现 New 对话框。

(3) 在 New 对话框中, 选择 Files 选项卡, 出现 Files 选项卡的内容。在其中双击 C++ Source File 选项, 出现文本编辑窗口。

(4) 将 C 语言程序逐行输入到计算机中, 并可以进行全屏幕编辑。使用 File 菜单中的 save 选项存盘。在“另存为”对话框的“文件名”文本框内输入文件名, 并以 .c 作扩展名。在输入文件前应先选好存放 .c 文件的文件夹。

(5) 选择菜单栏中的 Build 菜单项, 在该下拉菜单中, 选择 Build 选项, 这时系统对工作区中的当前文件进行编译、连接, 生成可执行文件。如果有编译错和连接错误, 错误信息将自动出现在屏幕下方窗口中, 用户双击某条编译错误信息后, 系统将在编辑窗口中使用右箭头指出出错的行。这时用户可根据出错信息修改错误, 修改后再继续编译, 直到没错为止。最终生成可执行文件。

(6) 选择 Build 菜单项中的 Execute 命令运行该程序, 将输出结果显示在一个 DOS 窗口中。查看完毕后, 按任意键返回原窗口。

编译运行后第一个程序后, 在编辑下一个新程序前, 应将工作区内的文件清除掉, 其方法如下:

选择 File 下拉菜单中 Close Workspace 选项, 出现一个对话框, 选择“是”时, 将清除工作压中的所有文件, 选择“否”时, 保留当前文件, 清除其他文件。

开始上机时, 最好让学生先不清除原来文件就进行运行, 看一个会出现什么现象? 这样可使学生记住以后出现这种现象应该如何处理。

初次上机的学生多遇到一些错误对他们有好处, 他们可从错误中学会解决方法, 逐渐积累调试程序的经验。

2. 多文件 C 语言程序的实现方法

多文件 C 语言程序的实现方法如下。

(1) 将多个 C 语言程序的文件依次编辑后, 存放在指定的文件夹中。

(2) 选择 File 菜单项中 New 选项, 出现 new 对话框, 单击 Projects 选项卡, 在出现的选项中, 单击 Win32 Console Application 选项, 在 Project name 文本框中输入项目文件名, 并在 Location 选框中输入项目文件所需的源文件所在的路径, 选择单选按钮 Create new workspace 后, 单击 OK 按钮。

(3) 出现 Win 32 Console Application 对话框, 选择 An empty project 按钮 (默认值), 单击 Finish 按钮。

(4) 出现 New Project Information 对话框, 单击 OK 按钮。完成了创建一个空项目文件的任务。

下面接着的操作是在向空项目文件中加入文件。

(5) 选择菜单项 **Project** 的下拉菜单中的 **Add To Project** 选项, 再在它的级联菜单中单击 **Files** 选项, 出现打开文件对话框。在该对话框中, 把要添加的文件加到该项目中。

(6) 选择 **Build** 菜单项, 对创建的项目文件进行编译连接, 生成可执行文件。

再使用运行命令 **Execute** 运行该命令, 输出结果出现在 **DOS** 窗口中。

关于单文件和多文件的 C 语言程序的实现, 教科书中都有较详细的描述, 请参看。

多文件程序还可以通过文件包含命令将某个或某些文件包含在文件中, 运行主文件便可以, 有关这方面的操作请参考第 4 章“预处理功能”。

1.3 重要概念

本章讲述 3 个重要概念。

1. 程序和文件

C 语言程序是由文件组成的。一个 C 语言的程序可以是一个文件, 也可以是多个文件。在多个文件的程序中只有一个主文件, 即含有主函数的文件。

要求学生把 C 语言程序和文件这两个概念区分开来。初学者容易把文件和程序看成是一回事。认为文件就是程序, 程序就是文件, 这是不对的。一定要让学生牢固树立程序是由文件组成的这个概念。由于我们多数情况下认为 C 语言是由一个文件组成的, 于是容易造成误解, 认为 C 语言程序就是一个文件。因此, 需要在第 1 章中就让学生运行一个多文件的程序, 尽管学生对该程序可能不太理解, 这没有关系, 关键要让学生知道 C 语言程序可由多个文件组成。

2. 输入语句和输出语句

在 C 语言程序中, 总会出现输入语句和输出语句, 特别是输出语句, 几乎每个程序中都要出现, 因为通过它才会把程序的结果输出显示出来。

输入语句用来实现输入操作, 最方便的输入操作是从计算机的键盘上输入信息。实现键盘输入操作的常用函数是标准格式输入函数 `scanf()`。该函数又称为读函数, 从标准设备键盘上向计算机读入信息。因此, 输入语句从外部向计算机程序中的变量读入信息, 它是通过读函数来实现的。

输出语句用来实现输出操作, 最常用的输出操作是将计算出结果输出到计算机显示器的屏幕上去进行显示, 实现这一操作的常用函数是标准格式输出函数 `printf()`。该函数又称为写函数, 它将计算机内部的信息写到输出设备屏幕上去。因此, 输出语句从计算机中将信息传递到输出设备上, 它是通过写函数来实现的。

由于在 C 语言程序中, 经常出现读函数 `scanf()` 和写函数 `printf()`, 因此要将 C 语言程序的输入输出操作搞清楚。

3. 源文件和可执行文件

使用高级语言编写的程序文件称为源文件, 源文件名是由主文件名和扩展名组成的,