

新世纪高等学校计算机学科系列教材

# 悦语言程序设计

刘德恒摇李盘林摇张晓燕摇编著

刘德恒摇李盘林摇张晓燕摇编著

北京·机械工业出版社

## 内 容 简 介

全书共分 10 章,介绍了悦语言的基本概念、语法规则和利用悦语言进行程序设计的方法。包括:悦语言概述、数据类型、运算符及表达式、语句、数组及其应用、函数和变量作用域、指针、结构、联合与枚举、位运算、编译预处理、文件等。

摇摇本书是根据作者多年从事悦语言的教学经验和实际应用的体会写成的,内容通俗易懂,并且每章均附有大量的例题和应用举例。在例题和习题的选择上,除安排了部分基本概念题和编程题外,还安排了大量的选择题和填空题,在附录 A 中给出了各章习题的参考答案。

摇摇本书不仅可作为高等院校教材和计算机培训班学习悦语言程序设计的教材,还可供工程技术人员使用。

摇摇未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

摇摇版权所有,翻版必究。

摇摇图书在版编目(CIP)数据

摇摇悦语言程序设计 刘德恒等编著 北京:电子工业出版社, 2004

摇摇 21 世纪高等学校计算机学科系列教材

摇摇 ISBN 7-302-10000-0

摇摇 I. 刘... II. 刘... III. 悦语言—程序设计—高等学校—教材 IV. 725.601.6

摇摇中国版本图书馆 CIP 数据核字(2004)第 000000 号

丛 书 名: 21 世纪高等学校计算机学科系列教材

书 名 名 称: 悦语言程序设计

编 著 者: 刘德恒 李盘林 摇摇 张 晓 燕

责任编辑: 张孟玮

特约编辑: 朱强国

排版制作: 电子工业出版社计算机排版室

印 刷 者:

装 订 者:

出版发行: 电子工业出版社 地址: 北京 邮编: 100013

北京市海淀区万寿路 193 号信箱 邮编: 100036

经 销 经 销: 各地新华书店

开 本 开 本: 185mm×260mm 印 张: 10.5 字 数: 250 千字

版 次 版 次: 2004 年 1 月第 1 版 2004 年 1 月第 1 次印刷

书 号 书 号: ISBN 7-302-10000-0

定价: 20.00 元

印 数 印 数: 5000 册 定 价: 20.00 元

凡购买电子工业出版社的图书,如有缺页、倒页、脱页、所附磁盘或光盘有问题者,请向购买书店调换;若书店售缺,请与本社发行部联系调换。电话: 010-68995166

## 前摇摇言

摇摇悦语言是目前国内外使用最广泛的高级程序设计语言之一。悦语言是根据结构化程序设计原则设计的。它的处理功能丰富,表达能力强,使用方便灵活,应用面广,目标代码效率高。它为结构化程序设计提供了各种数据结构和控制结构,既具有高级语言的特点又能够实现汇编语言中的大部分功能,而且用悦语言编写的程序具有特别好的可移植性。

摇摇本书是根据作者多年的悦语言教学经验和实际应用的体会而写成的,主要特点如下:摇摇员爱在章节的安排上,既顾及悦语言本身内容的完整性和知识的系统性,又尽量保持与考试大纲内容和要求的一致性。

摇摇圆爱按照循序渐进、由浅入深的原则,在文字叙述上力求条理清晰、简明扼要。在讲解基本概念和语法时,考虑到读者的接受能力,阐述上尽量做到浅显易懂,并通过反复强调和大量例题,使得读者既能把握要点又不感到抽象和枯燥。

摇摇猿爱各章节中的例题基本上只涉及本章和此前章节所讲解过的内容,而不含有尚未学过的内容,这样可以减少读者阅读程序和分析问题的困难,便于集中精力掌握好本章节的知识要点。

摇摇源爱按照“题对考路”的思想,根据我们对历年试题的分析和研究,在例题和习题的选择上,除安排了部分基本概念题和编程题外,还安排了大量的选择题和填空题,这样可使读者在学习知识点的同时,也熟悉了考试题型。

摇摇每章的最后都附有一定量的习题,这些习题对于读者巩固已学习的内容,是大有益处的。在附录圆中给出了各章习题的参考答案,可供读者参考。

摇摇作者认为,要学好悦语言,除了掌握悦语言的基本理论之外,还必须加强实践环节,本书中的所有例题都在微机上调试通过,希望读者边学习边上机实践,这样不仅可以加快学习进度,也能提高学习效率。

摇摇由于作者水平有限,经验不足,本书中一定有不少缺点和错误,敬请同行和广大读者批评指正。

编著者摇摇  
圆园园年 苑月 撰

## 序

为什么要组织《新世纪高等学校计算机学科系列教材》?根据什么组织这套教材?这些都是在这篇序言中要回答的问题。

计算机学科是一个飞速发展的学科,尤其是近十年来,计算机向高度集成化、网络化和多媒体化发展的速度一日千里。但是,从另一方面来看,目前高等学校的计算机教育,特别是教材建设,远远落后于现实的需要。现在的教材主要是根据《计算机学科教学计划(试行)》的要求组织编写的。这个教学计划,在制订过程中主要参照了美国陈其耘和粤语《悦悦(试行)》(简称悦悦(试行))。

近年来,计算机学科已有了长足发展,这就要求高等学校计算机教育必须跟上形势发展的需要,在课程建设和教材建设上作出相应调整,以适应新世纪高等学校计算机教育的要求。这是组织这套教材的初衷。

为了组织好这套教材,全国高等学校计算机教育研究会课程与教材建设委员会在天津召开了“全国高等学校计算机学科课程与教材建设研讨会”,在北京召开了“教材编写大纲研讨会”。在这两次会议上,代表们深入地研讨了全国高校计算机专业教学指导委员会、中国计算机学会教育委员会制订的《计算机学科教学计划(试行)》和美国陈其耘、粤语《悦悦(试行)》(简称悦悦(试行))。这是这套教材参照的主要依据。

陈其耘和粤语发表的悦悦(试行)是在总结了从悦悦(试行)到现在,计算机学科十年来发展的主要成果的基础上诞生的。它认为新世纪计算机学科应包括10个主科目,其中5个主科目为核心主科目。它们是:算法与分析(粤)、体系结构(粤)、离散结构(粤)、计算科学(悦)、图形学、可视化、多媒体(粤)、网络计算(粤)、人机交互(粤)、信息管理(粤)、智能系统(粤)、操作系统(粤)、程序设计基础(粤)、程序设计语言(粤)、软件工程(粤)、社会、道德、法律和专业问题(粤)。其中除悦和粤为非核心主科目外,其他8项均为核心主科目。

将悦悦(试行)与悦悦(试行)比较可看出:

在悦悦(试行)中,离散结构只作为数学基础提出,而在悦悦(试行)中,则作为核心主科目提出,显然,提高了它在计算机学科中的地位。

在悦悦(试行)中,未提及网络计算,而在悦悦(试行)中,则作为核心主科目提出,以适应网络技术飞速发展的需求。

图形学、可视化与多媒体也是为适应发展要求新增加的内容。

除此之外,悦悦(试行)在下述缘个方面作了调整:

程序设计语言引论调整为程序设计基础;将人机通信调整为人机交互;将人工智能与机器人学调整为智能系统;将数据库与信息检索调整为信息管理;将数值与符号计算调整为计算科学。

显然,这些变化使悦悦(试行)更具有科学性,也更好地适应了学科发展的需要。

在组织这套教材的过程中,充分考虑了这些变化和调整,在软件和硬件的课程体系、界面划分时均做了相应的调整,使整套教材更具有科学性和实用性。

另外,还要说明一点,教材建设既要满足必修课的要求,又要满足限选课和任选课的要求。

因此,这套教材按系列组织,反映了整个计算机学科的要求;采用大拼盘结构,对于美国麻省理工学院和香港发表的《悦国司中提出的五个主科目,这套系列教材均涵盖,能够满足不同层次院校、不同教学计划的要求,各校可根据自己的需求进行选拼使用。

这套教材由全国高等学校计算机教育研究会课程与教材建设委员会主任李大友教授精心策划和组织,参加这套教材组织工作的还有哈尔滨工业大学蒋宗礼教授、西安交通大学董渭清副教授、武汉大学张焕国教授、吉林大学张长海教授、福州大学王晓东教授、太原理工大学余雪丽教授、西安电子科技大学王闵教授等。

摇摇本套教材的编者均为具有丰富教学实践经验的专家和教授。所编教材体系结构严谨、层次清晰、概念准确、论理充分、理论联系实际、深入浅出、通俗易懂。

李大友摇摇

圆园园年 远月摇

# 目 录

第 1 章 悦语言概述 .....	( 1 )
1.1 悦语言简介 .....	( 1 )
1.2 悦语言特点 .....	( 1 )
1.3 悦语言程序结构和书写风格 .....	( 1 )
1.4 简单的悦语言程序 .....	( 1 )
1.5 悦语言程序结构 .....	( 1 )
1.6 悦程序的书写风格 .....	( 1 )
1.7 悦语言程序的编译和执行 .....	( 1 )
习 题 .....	( 1 )
第 2 章 数据类型、运算符及表达式 .....	( 2 )
2.1 悦语言的数据类型 .....	( 2 )
2.2 数据类型的概念 .....	( 2 )
2.3 悦语言的数据类型 .....	( 2 )
2.4 常 量 .....	( 2 )
2.4.1 整型常量 .....	( 2 )
2.4.2 实型常量 .....	( 2 )
2.4.3 字符常量 .....	( 2 )
2.4.4 字符串常量 .....	( 2 )
2.4.5 符号常量 .....	( 2 )
2.5 变 量 及其类型定义 .....	( 2 )
2.5.1 变 量 .....	( 2 )
2.5.2 变量类型定义 .....	( 2 )
2.5.3 变量的初始化 .....	( 2 )
2.6 数据类型之间的转换与运算 .....	( 2 )
2.6.1 自动类型转换 .....	( 2 )
2.6.2 强制类型转换 .....	( 2 )
2.7 运算符和表达式 .....	( 2 )
2.7.1 算术运算符和算术表达式 .....	( 2 )
2.7.2 赋值运算符和赋值表达式 .....	( 2 )
2.7.3 关系运算符和关系表达式 .....	( 2 )
2.7.4 逻辑运算符和逻辑表达式 .....	( 2 )
2.7.5 逗号运算符和逗号表达式 .....	( 2 )
2.7.6 条件运算符和条件表达式 .....	( 2 )
2.7.7 求字节运算符 .....	( 2 )
2.8 运算符的优先级及其结合性 .....	( 2 )



摇摇习题	(远)
第 缘章 摇摇函数和变量作用域	(远)
摇摇缘.1 摇摇函数的概念	(远)
摇摇缘.2 摇摇函数的定义和调用	(苑)
摇摇缘.2.1 摇摇函数的定义	(苑)
摇摇缘.2.2 摇摇函数的调用	(苑)
摇摇缘.3 摇摇函数的返回值及其类型	(苑)
摇摇缘.4 摇摇函数的参数及其传递方式	(苑)
摇摇缘.4.1 摇摇非数组作为函数参数的传值方式	(苑)
摇摇缘.4.2 摇摇数组名作为函数参数的传址方式	(苑)
摇摇缘.5 摇摇函数的嵌套调用和递归调用	(苑)
摇摇缘.5.1 摇摇函数的嵌套调用	(苑)
摇摇缘.5.2 摇摇函数的递归调用	(苑)
摇摇缘.6 摇摇变量的作用域及其存储类型	(愿)
摇摇缘.6.1 摇摇局部变量及其存储类型	(愿)
摇摇缘.6.2 摇摇全局变量及其存储类型	(愿)
摇摇缘.7 摇摇内部函数和外部函数	(愿)
摇摇缘.7.1 摇摇内部函数	(愿)
摇摇缘.7.2 摇摇外部函数	(愿)
摇摇缘.8 摇摇应用举例	(愿)
摇摇习题	(愿)
第 远章 摇摇指针	(怨)
摇摇远.1 摇摇指针的基本概念	(怨)
摇摇远.2 摇摇指针变量的定义和引用	(怨)
摇摇远.2.1 摇摇指针变量的定义	(怨)
摇摇远.2.2 摇摇指针变量的初始化	(怨)
摇摇远.2.3 摇摇指针变量的引用	(怨)
摇摇远.3 摇摇指针的运算	(怨)
摇摇远.4 摇摇指针与数组	(怨)
摇摇远.4.1 摇摇用指针访问数组元素	(怨)
摇摇远.4.2 摇摇用指针访问字符串	(怨)
摇摇远.5 摇摇指针数组	(怨)
摇摇远.6 摇摇指针与函数	(怨)
摇摇远.6.1 摇摇指针作为函数的参数	(怨)
摇摇远.6.2 摇摇数组名作为函数参数	(怨)
摇摇远.6.3 摇摇函数的返回值为指针	(怨)
摇摇远.6.4 摇摇指向函数的指针	(怨)
摇摇远.7 摇摇命令行参数	(怨)
摇摇远.8 摇摇多级指针	(怨)
摇摇远.9 摇摇应用举例	(怨)

摇摇习题 .....	( 员愿)
第 苑章摇结构、联合与枚举 .....	( 员园)
摇摇苑.1 摇结构类型及其变量的定义 .....	( 员园)
摇摇苑.2 摇结构类型变量的引用 .....	( 员源)
摇摇苑.3 摇结构变量的初始化 .....	( 员缘)
摇摇苑.4 摇结构作为函数参数 .....	( 员远)
摇摇苑.5 摇结构与数组 .....	( 员愿)
摇摇苑.6 摇结构与指针 .....	( 员园)
摇摇摇摇苑.6.1 摇指针指向结构对象 .....	( 员园)
摇摇摇摇苑.6.2 摇指针作为结构的成员 .....	( 员员)
摇摇摇摇苑.6.3 摇链表 .....	( 员圆)
摇摇苑.7 摇结构的嵌套 .....	( 员远)
摇摇苑.8 摇联合 .....	( 员愿)
摇摇苑.9 摇枚举 .....	( 员园)
摇摇苑.10 摇应用举例 .....	( 员圆)
摇摇习题 .....	( 员愿)
第 愿章摇位运算 .....	( 员园)
摇摇愿.1 摇二进制位运算 .....	( 员园)
摇摇摇摇愿.1.1 摇“按位与”运算符 & .....	( 员园)
摇摇摇摇愿.1.2 摇“按位或”运算符   .....	( 员圆)
摇摇摇摇愿.1.3 摇“按位异或”运算符 ^ .....	( 员猿)
摇摇摇摇愿.1.4 摇“按位取反”运算符 ~ .....	( 员源)
摇摇摇摇愿.1.5 摇左移运算符 << .....	( 员源)
摇摇摇摇愿.1.6 摇右移运算符 >> .....	( 员缘)
摇摇愿.2 摇位复合赋值运算符 .....	( 员远)
摇摇愿.3 摇位段 .....	( 员远)
摇摇愿.4 摇应用举例 .....	( 员愿)
摇摇习题 .....	( 员员)
第 怨章摇编译预处理 .....	( 员猿)
摇摇怨.1 摇宏定义 .....	( 员猿)
摇摇摇摇怨.1.1 摇符号常量定义 .....	( 员猿)
摇摇摇摇怨.1.2 摇带参数的宏定义 .....	( 员缘)
摇摇怨.2 摇文件包括 .....	( 员远)
摇摇怨.3 摇条件编译 .....	( 员苑)
摇摇怨.4 摇应用举例 .....	( 员员)
摇摇习题 .....	( 员圆)
第 10 章摇文件 .....	( 员缘)
摇摇10.1 摇文件概述 .....	( 员缘)
摇摇10.2 摇缓冲文件系统 .....	( 员远)
摇摇摇摇10.2.1 摇文件打开函数 fopen .....	( 员远)



# 第 1 章 C 语言概述

## 1.1 C 语言简介

C 语言是目前世界上最为流行的计算机高级程序设计语言之一，它设计精巧，功能齐全，不仅是开发系统软件的理想工具，而且也是开发应用软件的理想程序设计语言。因此，它深受广大程序设计者的欢迎。

C 语言是在 1972 年至 1973 年间由美国贝尔实验室的 D.M.Ritchie 为描述和实现 UNIX 操作系统而设计的。在此以前像 UNIX 操作系统这样的系统软件，一般都是利用汇编语言这种低级语言来编写的。C 语言的开发成功，使得利用高级语言来编写系统软件成为可能。UNIX 操作系统源代码的 90% 以上都是由 C 语言编写的，UNIX 操作系统的一些主要特点，如便于理解、易于修改及具有良好的可移植性等，在一定程度上都受益于 C 语言，所以，UNIX 操作系统的成功与 C 语言是密不可分的。

最初的 C 语言是依赖于 UNIX 操作系统环境的，随着 C 语言的不断发展及应用的普及，目前，C 语言已经能够在多种操作系统如 UNIX、DOS 等环境下运行，而且，不断地推出新的 C 语言版本，其性能也越来越强，20 世纪 80 年代初又出现了 C++。实用的 C 语言编译系统种类繁多，适用于 IBM-PC 机运行的就有 Microsoft C(MSC)、Turbo C(TC)和 Lattice C(LC)等。

1988 年 Brian W.Kernighan 和 Dennis M.Ritchie 按照美国国家标准协会(ANSI)的要求重新编写了《The C Programming Language》一书，因此，目前将该书中介绍的 C(ANSI C)称为标准 C。

## 1.2 C 语言特点

C 语言能够成为目前应用最广泛的高级程序设计语言之一，完全是由其语言特点决定的。C 语言的特点可大致归纳如下：

C 语言短小精悍，基本组成部分紧凑、简洁。C 语言一共只有 32 个标准的关键字、45 个标准的运算符以及 9 种控制语句，不但语言的组成精练、简洁，而且使用方便、灵活。

C 语言运算符丰富，表达能力强。C 语言具有“高级语言”和“低级语言”的双重特点，其运算符包含的内容广泛，所生成的表达式简练、灵活，有利于提高编译效率和目标代码的质量。

C 语言数据结构丰富，结构化好。C 语言提供了编写结构化程序所需要的各种数据结构和控制结构，这些丰富的数据结构和控制结构以及以函数调用为主的程序设计风格，保证了利用 C 语言所编写的程序能够具有良好的结构化。

C 语言提供了某些接近汇编语言的功能，有利于编写系统软件。C 语言提供的一些

运算和操作，能够实现汇编语言的一些功能，如它可以直接访问物理地址，并能进行二进制位运算等，这为编写系统软件提供了方便。

C 语言程序所生成的目标代码质量高。C 语言程序所生成的目标代码的效率比用汇编语言描述同一个问题低 20% 左右，因此，用 C 语言编写的程序执行效率高。

C 语言程序的可移植性好。在 C 语言所提供的语句中，没有直接依赖于硬件的语句，与硬件有关的操作，如数据的输入、输出等都是通过调用系统提供的库函数来实现的，而这些库函数本身并不是 C 语言的组成部分。因此，用 C 语言编写的程序能够很容易地从一种计算机环境移植到另一种计算机环境中。

当然，C 语言本身也有其弱点：一是运算符的优先级较多，不容易记忆；二是由于 C 语言的语法限制不太严格，这在增强了程序设计的灵活性的同时，在一定程度上也降低了某些安全性，这对程序设计人员提出了更高的要求。

总之，由于 C 语言的上述特点，使得 C 语言越来越受到程序设计人员的重视，并且已经在广泛的领域里得到了应用。

## 1.3 C 语言程序结构和书写风格

### 1.3.1 简单的 C 语言程序

下面介绍几个简单的 C 语言程序，并对其基本语法规则和表现形式进行相应的说明，以使读者对 C 语言程序结构有一个概括的了解。

【例 1.1】 编写一个 C 语言程序，显示字符串“Welcome to use C.”。

```
#include "stdio.h"
main( )
{
    printf("Welcome to use C.\n");
}
```

这是一个简单而完整的 C 语言程序，经过编辑、编译和连接后，其执行结果是在屏幕的当前光标位置处显示如下字符串：

```
Welcome to use C.
```

下面对上述程序进行一下说明。

该程序由一个函数 `main()` (称作主函数) 构成。一个 C 语言程序可以由多个函数组成，但任何一个完整的 C 语言程序，都必须包含一个且只能包含一个名为 `main` 的函数，程序总是从 `main` 函数开始执行的。

由左右花括号括起来的部分是函数体，函数体中的语句将实现程序的预定功能。在本例中，`main` 函数的函数体中只有 `printf` 一个语句，它是调用由系统提供的标准库函数 `printf()`，功能是进行格式化输出(显示)，即将字符串“Welcome to use C.\n”显示在终端屏幕上。其中，字符串中的字符“\”和“n”合起来，表示一个“回车-换行”字符，在“回车-换行”字符后输出的任何字符，将被显示在屏幕的下一行上。

`printf()` 后的分号是语句结束符。C 语言中的每个基本语句，都是以“;”结束的，若

无“;”则视为语法错误。

#include 是编译预处理命令,由双引号括起来的 stdio.h 被称作“头文件”,“stdio.h”文件中定义了 I/O 库所用到的某些宏和变量,其作用是将由双引号(或尖括号)括起来的文件中的内容,读入到此命令的位置处。在使用 C 语言输入、输出库函数时,一般需要使用#include 命令将“stdio.h”文件包含到源文件中。有关#include 命令的作用及其使用方法,将在“编译预处理”一章中作详细介绍。

【例 1.2】 从键盘输入两个整数,并将这两个整数之和显示出来。

```
#include "stdio.h"
main()                /* 计算两个整数之和 */
{
    int x, y, z;
    scanf("%d%d", &x, &y);    /* 读入两个整数,存入变量 x 和 y 中 */
    z=x+y;
    printf("%d + %d = %d", x, y, z);
}
```

上述程序经过编辑、编译和连接后,当程序执行到 scanf 语句时,将等待用户输入两个整型数据后再继续执行,假如用户输入 128 和 375(此时, x 的值为 128, y 的值为 375),则程序将显示如下信息:

128 + 375 = 503

下面,对上述程序进行一下说明。

程序中由/\*和\*/括起来的内容是程序的注释部分,它是为增加程序的可读性而设置的。注释部分对程序的编译过程和执行结果没有任何影响。

C 语言中的所有变量都不能直接引用,必须先定义(或说明)为某种数据类型,然后方可使用,同时必须遵循“先定义(或说明)后使用”的原则,如语句:

```
int x, y, z;
```

定义了 x、y、z 是三个整型变量,以后就可以使用这三个变量来存放整型数据。

scanf("%d%d", &x, &y)语句的作用是进行格式化输入,它是由系统提供的标准库函数,其后的圆括号内为参数表,不同的函数需要不同的参数。scanf 函数中的参数主要包括两部分内容:一是“格式控制”部分,如本程序中“%d%d”表示输入的两个数据均为十进制整型格式,指明给 x、y 输入整型数;二是“地址表”部分(本书中出现的表的概念,如地址表、输出表等,是指用逗号分隔的有限个元素序列),它使用的是存放输入数据的变量的地址,如“&x, &y”是 x 和 y 的地址。

“z=x+y;”是赋值语句,等号“=”是赋值运算符,表示把右边表达式 x+y 的值赋给变量 z,即将 503 赋给 z。

程序中“printf("%d+%d=%d", x, y, z);”是进行格式化输出,其参数也包括两部分内容:一是“格式控制”部分,用于对输出数据的格式进行说明;二是“输出表”部分,它使用的是存放输出数据的变量本身。本程序中执行此语句,首先在当前光标位置上按输出十进制整型格式(%d)输出变量 x 的值,再输出字符“+”,然后按整型格式输出变量 y 的值,再输出字符“=”,最后按整型格式输出变量 z 的值。

### 1.3.2 C 语言程序结构

通过上面两个简单程序的实例，我们可以看出 C 语言程序的结构有以下特点：

一个 C 语言程序可以由多个函数组成，但任何一个完整的 C 语言程序，都必须包含一个且只能包含一个名为 main 的函数，此函数称为“主函数”。C 程序总是从 main 函数开始执行，而不论 main 函数在整个程序中的位置如何(main 函数可以放在程序最前头，也可放在程序最后，或在一些函数之前且在另一些函数之后)，主函数的结束就是整个程序的结束。

C 语言程序是由一个或多个函数组成，所以我们说函数是 C 语言程序的基本单位。函数的功能是通过函数之间的相互调用来实现的。程序中所使用的函数，既可以是系统提供的库函数，也可以是用户根据需要自己定义的函数。如上述 main 函数中调用的 scanf 函数和 printf 函数，就是系统提供的库函数，这些函数不需要用户自己定义，在需要时，只要按照指定的格式进行调用即可。C 语言编译系统提供的库函数非常丰富，特别是与硬件打交道的部分，一般都是通过调用库函数来实现的。

函数体是用左右花括号括起来的部分，它包括变量的定义和一组执行语句，C 语言所用到的所有变量都必须遵循“先定义后使用”的原则，如例 1.2 中“int x, y, z;”语句。每个语句都由分号“;”结束。

在函数定义之外还可包含一个说明部分，该说明部分叫外部说明，它可包括预编译命令(如例题中的#include "stdio.h")、外部变量的说明等。

可以在程序中任何可以加空格符的地方(除一个语句内部)加上注释，若在函数名或保留字中间插入注释内容是错误的。注释的格式为：

```
/* 注释内容 */
```

一个注释可写成一，也可写成若干行。注释内容可以用英文，也可以用汉字。

### 1.3.3 C 程序的书写风格

C 语言程序的书写风格如下：

C 语言程序的书写格式比较自由，没有固定的格式要求。为了提高程序的可读性，往往根据语句的从属关系，以缩进书写的形式来体现语句的层次性，即在语句之前加上适当数量的空格字符，使处于同一层次的语句从同一列开始。

C 程序在一行内，既可以写一个语句，也可以写多个语句。我们推荐一行一句的风格。

C 程序对大小写没有要求，但习惯上用小写字母来书写，大写字母一般用做符号常数名或其他特殊用途。

标识符的命名应当尽量使其“见其名而知其义”。

适当地使用注释，可增加程序的可读性。

## 1.4 C 语言程序的编译和执行

我们把编好的 C 语言程序叫 C 源程序，从 C 源程序到计算机上得到运行结果，其基本过程如图 1.1 所示。

图 1.1 C 语言程序开发过程

(1) 编辑

选择适当的编辑程序，将 C 语言源程序通过键盘输入到计算机中，并以文件的形式存入到磁盘中。在 DOS 系统下，经过编辑后得到的源程序文件都是以.C 为其文件扩展名。

(2) 编译

通过编辑程序将源程序输入到计算机后，需要经过 C 语言编译器将其生成目标程序。在对源程序的编译过程中，可能会发现程序中的一些语法错误，这时就需要重新利用编辑程序来修改源程序，然后再重新编译。在 DOS 系统下，经过编译后得到的目标文件都是以.OBJ 为其文件扩展名。

(3) 连接

经过编译后生成的目标文件是不能直接执行的，它需要经过连接之后才能生成可执行的代码。在 DOS 系统下，连接后所得到的可执行文件都是以.EXE 为其文件扩展名。

(4) 执行

经过对源程序编译、连接之后，就可以生成可执行的文件，这时就可以执行该可执行文件了。另外，在 DOS 命令提示符下，只要键入可执行的文件名，并按“回车”键，就可运行可执行文件。

现在有许多厂家都推出一种集成环境来处理 C 语言程序，如 Turbo C，在这种集成环境下，对程序的编辑、编译和连接等操作，都可以集成在一起完成，使用起来非常方便。

## 习 题

- 1.1 请简述 C 语言的主要特点及其用途。
- 1.2 请参照本章例题，编写一个 C 语言程序，用于显示以下信息：

Happy New Year!

1.3 请参照本章例题，分析下面的C语言程序，并给出运行结果。

```
#include "stdio.h"
main()
{
    int a , b , c ;
    a=100 ;
    b=20 ;
    c=a-b ;
    printf("%d" , c) ;
}
```

1.4 构成C语言程序的基本单位是( )，请从下列选择项中选出正确答案。

- (A) 过程 (B) 函数  
(C) 语句 (D) 命令

1.5 一个C语言程序至少包含一个( )函数，请从下列选择项中选出正确答案。

- (A) printf (B) scanf  
(C) main (D) 自定义

1.6 C语言程序中每个语句和数据定义的最后必须有一个( )，请从下列选择项中选出正确答案。

- (A) 分号 (B) 逗号  
(C) 空格 (D) 冒号

1.7 请从下列选择项中选出错误的选项( )。

- (A) #include <stdio.h>  
void main(/\* 主函数 \*/)
{
}
- (B) #include <stdio.h>  
void main()
{ prin /\* 打印 \*/ tf("name") ;
}
- (C) #include <stdio.h>  
void main()
{ int x /\* 初始化 \*/ = 10 ;
printf("%d" , x) ;
}
- (D) #include <stdio.h>  
void main()
{ int x=10 ;
/\* 打印 printf("%d" , x) ; \*/
}

1.8 一个完整的 C 语言程序包含一个或多个函数。对于函数 main( )，哪种说法是不正确的？请从下列选择项中选出正确答案。

- (A) 它是程序开始运行的第一个被调用的函数。
- (B) 没有它程序就无法运行。
- (C) 它是唯一不可缺少的函数。
- (D) 有没有它都行。

1.9 请从下列选择项中选出程序注释用法不正确的答案。

- (A) #inc /\* 包含 \*/ lude <stdio.h>

```
void main()  
{  
}
```

- (B) #include <stdio.h>  
void main(/\* 无参数 \*/)

```
{  
}
```

- (C) #include <stdio.h>

```
void main()  
{  
    float x =2.0 ;  
    printf("%f/* 实型 */" , x);
```

```
}
```

- (D) #include <stdio.h>

```
void main()  
{ float x /* 变量 */=2.0 ;  
    printf("%f /*显示 f /*f=2.0  
    */的值*/" , x);  
}
```