

第 1 章 欢迎进入 C 语言世界

【引 例】

你可能偶尔玩过网络游戏，可能会使用 QQ 工具进行网上聊天，更是几乎每天都要使用手机与人进行沟通。那么，当这些内容与你的学习和生活密不可分，你是否曾经想过，是什么使电脑或智能终端等硬件设备拥有如此丰富多彩的功能？

你可能知道神舟十号载人航天器在距地面 343 公里的近圆轨道与天宫一号对接，你也可能知道石油地质勘探都要使用大型计算机进行模拟运算，你是否想过，是什么样的“大脑”能实现如此精确的控制和复杂模拟？对，是软件，是程序，是用某种计算机语言编写的程序，是通过程序实现了上述功能。

计算机编程语言有很多种，而 C 语言是世界上最流行的程序设计语言之一，也是最优秀的计算机编程语言之一。

【本章内容提要】

- ◆ 了解 C 语言程序宏观框架结构特点。
- ◆ 了解在 VC++ 6.0 软件开发环境下，C 语言程序开发过程。
- ◆ 简要了解 C 语言程序设计应掌握的知识脉络。

1.1 程序宏观框架结构及构成

1.1.1 程序框架结构

计算机的本质是“程序的机器”。程序设计是软件开发人员的基本能力，懂得程序设计，才会进一步懂得计算机，进而真正了解计算机是如何工作的。通过学习程序设计，可帮助大家进一步了解计算机的工作原理，培养分析问题和解决问题的能力。即使大家将来不直接从事计算机软件开发，但由于学过了程序设计，理解软件生产的特点和生产过程，就能与程序开发人员进行更好的沟通和合作，从而能更好地开展相关工作。

学习一门程序设计语言的唯一途径就是阅读程序并使用该语言编写程序。下面我们先通过几个简单的应用实例认识一下 C 语言程序。

【例 1-1】

在计算机或一些智能终端启动时，经常会出现欢迎界面或提示语，本例将实现在计算机屏幕上显示“欢迎进入 C 语言的世界！”提示信息。

代码清单 1-1

```
#include <stdio.h>           //包含标准输入输出头文件
void main()                 //主函数
{ printf("欢迎进入 C 语言的世界! \n"); //调用输出函数在屏幕上显示提示信息
}
```

该实例看上去很简单，却体现了 C 语言程序最基本的程序框架：

第一部分称之为“编译预处理”，形如示例中的程序段：

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

第二部分称之为“函数组”，形如示例中的程序段：

```
void main()
{ printf("欢迎进入 C 语言的世界! \n");
}
```

“编译预处理”以“#”开头，其作用是后面的函数做准备工作的。上述示例中的编译预处理部分只有一条头文件包含命令 `#include <stdio.h>`，其含义是在程序中包含标准输入输出头文件 `stdio.h`，该头文件中声明了输入和输出库函数及其他信息，这意味着在后面的程序中将用到该文件中的内容。

“函数组”由多个函数构成，函数是构成 C 语言程序的基本单位，多个函数共同协作完成程序要实现的功能。函数组中必须包括一个 `main()` 主函数，且 C 语言程序中有且仅有一个主函数，整个程序的执行从主函数开始，以主函数为核心展开。此外，函数组中除了主函数外还包括库函数和用户自定义的函数。

上述示例中的函数组部分只有一个主函数 `main()`，主函数调用库函数 `printf()` 在屏幕上输出“欢迎进入 C 语言的世界！”提示信息，库函数使用之前必须在编译预处理部分包含其所对应的头文件，所以在示例程序的开始就出现了 `#include <stdio.h>`。

除了主体框架的“编译预处理”和“函数组”以外，在程序中还允许为程序添加注释，以增强程序的可读性。上述示例中以“//”为起始的文字描述是程序中的注释。

【例 1-2】

从键盘输入矩形的长和宽，计算并在屏幕上显示输出该矩形的周长。

通过程序代码清单 1-2 可以看出，该示例中程序框架依然是编译预处理和函数组两部，只是稍复杂些。其中编译预处理部分只包括一条头文件包含命令，包含标准输入输出头文件 `stdio.h`，为主函数中用到的输出函数 `printf()`、输入函数 `scanf()` 做准备；函数组部分同样只包括一个主函数 `main()`，主函数通过数据准备、数据输入、数据计算、输出结果等语句

实现了题目的要求。

代码清单 1-2

```
#include <stdio.h>           //包含标准输入输出头文件
void main()                 //主函数
{ int a,b,c;                //数据准备, 定义整型变量
printf("请输入矩形的长和宽: "); //调用输出函数, 显示提示语
scanf("%d%d",&a,&b);        //数据输入, 从键盘输入 a、b 值
c=2*(a+b);                 //数据计算, 将计算的周长赋值给 c
printf("该矩形周长为: %d.\n",c); //输出结果, 调用输出函数输出结果
}
```

通过上述两个示例的描述和解读, 相信读者对 C 语言程序宏观框架有了一个基本的了解, 下面总结如下:

- ① C 语言程序基本框架包括编译预处理和函数组两部分。
- ② 编译预处理是程序编译之前的准备工作, 以“#”开头。
- ③ 函数组包括主函数、库函数和用户自定义函数。函数是构成 C 语言程序的基本单位, main()称为主函数, C 语言程序中有且仅有一个主函数, 整个程序的执行从主函数开始, 以主函数为核心展开; C 语言标准函数库中提供了大量功能丰富的库函数, 当使用到相应的库函数时, 需要在编译预处理中包含相应的头文件, 具体可参见附录 D; 用户也可以根据需要编写具有特定功能的函数, 称为用户自定义函数。

④ 在 C 语言中的任何适当位置可添加注释, 以增强程序的可读性。在 Visual C++ 编程环境中, 可用“//”作为程序单行注释的起始符号, 也可以使用“/*”和“*/”作为单行或多行注释的起始和终止符号。

【随堂练习 1-1】

根据上述示例模仿编程, 输入正方形的边长, 计算其面积。

1.1.2 程序的构成

如同格式规范的文章由字词、句子、段落逐级构成一样, 在程序宏观框架结构下, C 语言程序由标识符、语句、函数等表述形式构成, 最终形成完整的 C 语言程序代码。

1. 标识符

在上述程序代码中, 我们看到由 void、main、int、printf、scanf、a、b、c 等一系列符号构成了程序中的语句和函数, 这些符号统称为标识符, 标识符是用来标识程序中某个对象名字的字符序列, 这些对象可以是语句、数据类型、函数、变量、常量等。C 语言把标识符分为三类, 即关键字、预定义标识符、用户自定义标识符。

(1) 关键字

在 C 语言中，为了定义变量、表达语句功能，或说明某些信息的存储类型，必须用到一些具有特殊意义的标识符，这些标识符就是关键字。如上述程序代码中的 `void`、`int`。C 语言中常用的关键字及其解释可参见附录 C。

C 语言中关键字主要有以下两类：

- ① 类型说明符：用来说明变量、函数的类型，如 `int`、`float`、`char`、`void` 等。
- ② 语句定义符：用来表示一个语句的功能，如 `if`、`for`、`while`、`return` 等。

(2) 预定义标识符

预定义标识符是指已经被 C 语言系统预先定义好了的具有特定含义的标识符，如上述程序代码中的函数名 `printf`、`scanf`，以及预处理命令 `define`、`include` 等等。

(3) 用户自定义标识符

在编写程序过程中，用户需要给自定义的符号常量、变量、函数、数组、类型等起名字，这就是用户自定义标识符。C 语言规定，所有的用户标识符必须先定义或说明后再使用。

用户自定义标识符的命名有其规则：C 语言规定，用户标识符由字母（A~Z，a~z）、数字（0~9）、下划线“_”组成，并且首字符不能是数字，通常以下划线开头的标识符是编译系统专用的，所以在编写 C 语言程序时，最好不要使用以下划线开头的标识符。

在用户自定义标识符时还应注意以下几点：

- ① C 语言对大小写字符敏感，所以在编写程序时要注意大小写字符的区分。例如：对于 `max` 和 `Max` 这两个标识符来说，C 语言会认为这是两个完全不同的标识符。
- ② 不能把 C 语言关键字作为用户自定义标识符。
- ③ 通常不使用预定义标识符作为用户标识符，因为这样会失去系统规定的原意，同时造成二义性。
- ④ 用户自定义标识符的命名应做到简洁明了，尽量做到“见名知意”，从而便于程序的阅读和维护。例如，用 `length` 表示长度，`sum` 表示求和，`max` 表示最大值。

2. 函数和语句

函数是构成 C 语言程序的基本单位，而语句则是函数的重要组成元素。函数结构形式如下所示：

```
函数返回值类型 函数名（形式参数） //函数首部
{ 说明语句； //函数体
  可执行语句；
}
```

函数的第一行称为函数首部，大括号“{}”括起来的部分称为函数体。函数体由若干语句组成，函数体使用一对大括号“{”和“}”作为定界符。

根据语句在程序中所起的作用不同,语句可分为说明语句和可执行语句,通过这些语句可实现对数据的描述和操作。通常情况下,为了增强程序的可读性,每行只写一条语句,每个语句以分号“;”结束,如果没有写“;”,在程序编译时会出现“missing ;”(忘记;)的错误提示。

【随堂练习 1-2】

1. 下面哪些标识符属于合法的用户自定义标识符:

Main、void、_num、my\$、a*、N4、3m、a-2

2. 结合【例 1-2】指出程序代码中所用到的标识符哪些是关键字,哪些是预定义标识符,哪些是用户自定义标识符。

3. 分析【例 1-2】中函数的结构,包括函数首部(函数返回值类型、函数名、形式参数)、函数体语句(说明语句、可执行语句)。

1.2 程序开发过程及环境

1.2.1 程序开发过程

用 C 语言编写的程序称为源程序,它不能被计算机直接识别和执行,为此,还需要一种担任翻译工作的程序,即编译程序,它可以把 C 语言程序代码转换为计算机能够直接识别和执行的二进制目标代码。

从编写 C 语言源程序到运行程序需要经过以下四个步骤。

1. 编辑源程序

编辑是指在文本编辑工具软件中输入和修改 C 语言源程序,最后以文本文件的形式存放在磁盘上。虽然可使用记事本等字处理软件直接编写 C 语言源程序,但通常都在专用的软件开发工具中进行,如 Turbo C 和 Visual C++等,用 Visual C++编辑的源程序存入磁盘后,系统默认的文件扩展名为“.cpp”。

2. 编译源程序,生成目标程序

编译是将已编辑好的源程序翻译成二进制目标程序。编译是由系统本身的编译程序来完成的,编译过程将对源程序进行语法检查,当发现错误时,会提示错误的类型和出错的程序位置,以使用户修改。直至未发现语法错误时,会自动形成扩展名为“.obj”的目标程序。

3. 连接目标程序及其相关模块,生成可执行程序

一个 C 语言应用程序,可能包含有 C 语言标准库函数和许多模块,而各个模块往往是单独编译的,因此,经编译后得到的目标程序还不能直接执行,需要把编译好的各个模

块的目标程序与系统提供的标准库函数进行连接，生成扩展名为“.exe”的可执行程序。

连接过程由系统提供的连接程序完成，如果连接过程中出现错误信息，则需要修改错误后重新进行编译和连接，直到生成可执行程序。

4. 运行可执行程序

运行程序，并检查运行结果。如果是算法错误，只能回到第一步修改源程序，再重新编译、连接和运行，直到得到正确的结果。

总之，C 语言程序开发主要经过编辑、编译、连接和执行四个步骤，其完整过程描述如图 1-1 所示。

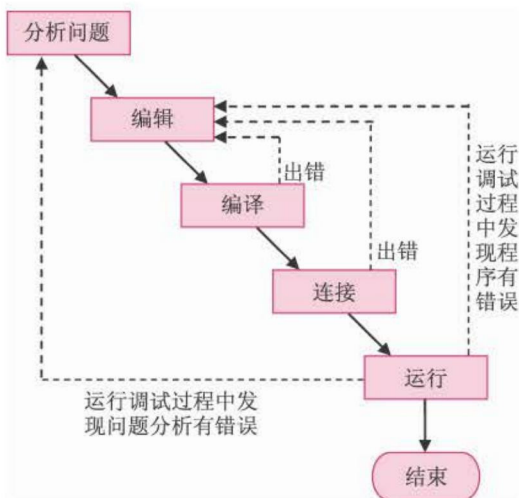


图 1-1 C 语言程序开发过程

1.2.2 程序开发环境

C 语言程序开发工具有很多，本书以常用的 Visual C++ 6.0（简称为 VC++ 6.0）作为程序开发环境。VC++ 6.0 是 Microsoft 公司推出的一个基于 Windows 系统平台、可视化的软件开发工具，提供了集编辑、编译、连接和运行于一身的集成开发环境。目前，VC++ 6.0 已成为专业程序员使用 C 语言进行软件开发的首选工具。

使用 VC++ 6.0 开发应用程序的步骤如图 1-2 所示。

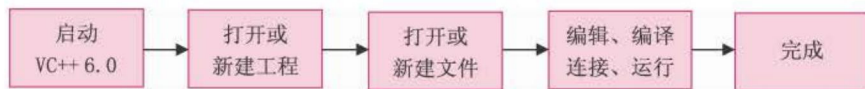


图 1-2 使用 VC++ 6.0 开发应用程序的步骤

下面以【例 1-1】为例，详细描述在 VC++ 6.0 环境下进行程序开发的步骤。

1. 启动 VC++ 6.0

选择“开始→程序→Microsoft Visual Studio 6.0→Microsoft Visual C++ 6.0”菜单命令或双击桌面上的 Visual C++ 6.0 快捷方式图标，如图 1-3 所示，即可打开 VC++ 6.0 用户界面。VC++ 6.0 初始界面如图 1-4 所示。

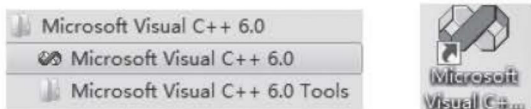


图 1-3 启动 VC++ 6.0 菜单及 VC++ 6.0 快捷方式图标



图 1-4 VC++ 6.0 初始界面

2. 新建工程

选择“文件→新建”菜单命令，打开“新建”对话框，如图 1-5 所示，在工程选项卡的列表框中选择“Win32 Console Application (Win32 控制台应用程序)”选项，在“工程名称”文本框中输入工程名称，如“test”，在“位置”文本框中输入或选择工程存放的位置，如“D:”，然后单击“确定”按钮。



图 1-5 新建工程对话框

单击“确定”按钮后显示如图 1-6 所示的询问对话框，选择“一个空工程”选项，单击“完成”按钮。



图 1-6 询问对话框

单击“完成”按钮后显示如图 1-7 所示的“新建工程信息”对话框，单击“确定”按钮，工程建立完成，此时可看到 D 盘中将出现新建的工程文件夹“test”，文件夹中有工程初始文件。



图 1-7 “新建工程信息”对话框

3. 新建源程序文件

选择“文件→新建”菜单命令，打开“新建”对话框，选择“文件”选项卡，如图 1-8 所示，在其列表框中选择“C++ Source File (C++源文件)”选项，在“文件名”文本框中输入文件名称，如“c1-1.cpp”（扩展名可以省略），在“位置”文本框中输入或选择文件存放的文件夹，单击“确定”按钮后进入程序编辑窗口。

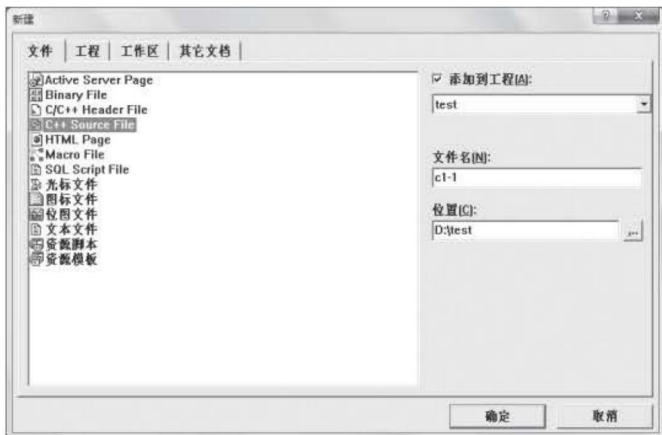


图 1-8 新建文件对话框

程序编辑窗口如图 1-9 所示。程序编辑完成后，可单击工具栏中的“保存”按钮，按【Ctrl+S】组合键，或选择“文件→保存”菜单命令保存文件。此时可看到“D:\test”文件夹中增加了“c1-1.cpp”文件。

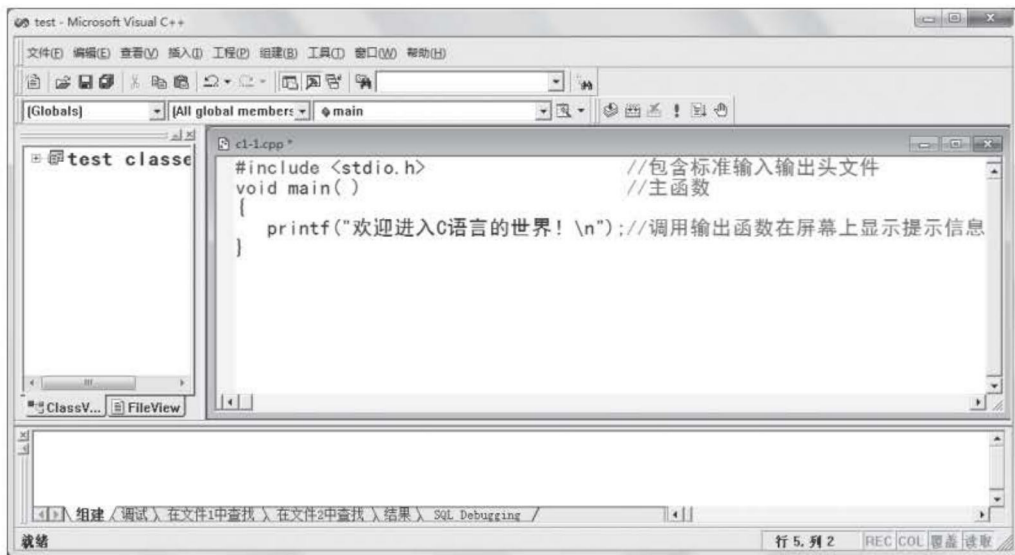


图 1-9 程序编辑界面

4. 编译源程序

选择“组建→编译”菜单命令或单击工具栏中的编译按钮，开始对源程序进行编译，若未出现编译错误，则生成扩展名为“.obj”的目标程序；若出现编译错误，则需要根据编辑窗口下方的信息提示栏中的“错误信息”对源程序继续编辑修改，直到编译通过为止。此时可看到“D:\test\debug”文件夹中增加了“c1-1.obj”文件。

5. 生成可执行程序

选择“组建→组建”菜单命令或单击工具栏中的组建按钮，即可生成扩展名为“.exe”的可执行程序。此时可看到“D:\test\debug”文件夹中增加了“test.exe”文件。

6. 执行程序

选择“组建→执行”菜单命令或单击工具栏中的组建执行按钮，可执行前面创建的可执行程序。此时将显示程序执行的输出窗口，如图 1-10 所示。



图 1-10 程序运行结果输出窗口

在程序开发过程中，可通过工作空间视图“ClassView（类视图）”和“FileView（文件视图）”查看工程信息，在工程中删除或添加文件。如图 1-11 所示。

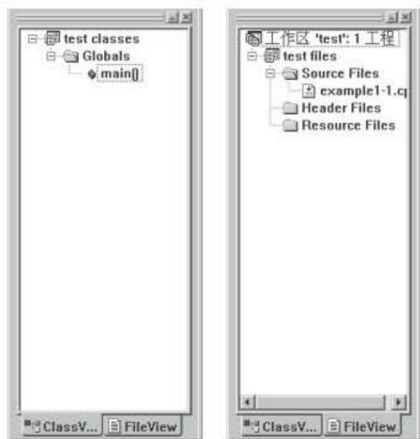


图 1-11 工作空间视图

【随堂练习 1-3】

1. 在 VC++6.0 环境中完成【例 1-2】程序开发过程。
2. 查阅【例 1-2】所对应的工程文件夹，了解相关文件的含义。

1.3 C 语言特点及知识脉络

C 语言是目前世界上最流行、使用最广泛的高级程序设计语言。对于操作系统、系统应用程序以及需要对硬件进行操作的场合，用 C 语言编程明显优于其他高级语言。C 语言具有绘图能力强、可移植性好、生成的目标代码质量高、程序执行效率高、数据处理能力强等特点，特别适合编写系统软件、二维三维图形和动画软件、数值计算软件等。

大家平时看到的 C 语言程序主要是各种硬件驱动程序、嵌入式程序（例如，一些车载导航系统程序、智能手机系统及应用程序、POS 机系统程序等）。此外，大部分操作系统也主要是由 C 语言写成的。总体来说，需要与硬件打交道的地方大多采用 C 语言进行编程，所以 C 语言的运用价值不是通常的 Windows 可视化桌面应用软件能体现的。

C 语言程序设计是面向过程的程序设计，其学习过程可以按照“宏观知识框架”→“微观知识细节”→“模块化综合性程序设计”的自然认知过程进行，下面就按照这种思路给出 C 语言知识脉络结构图，使读者对 C 语言的知识要点和学习过程有一个基本的了解，如图 1-12 所示。

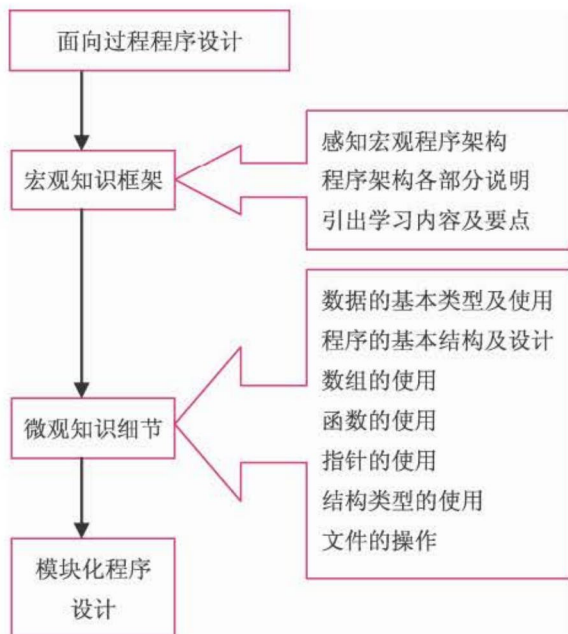
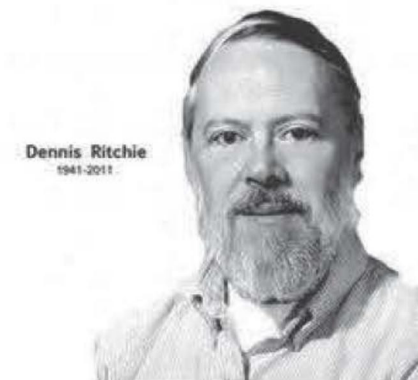


图 1-12 C 语言知识脉络

知识拓展

C 语言与丹尼斯·里奇



丹尼斯·里奇，全名丹尼斯·麦卡利斯泰尔·里奇，美国计算机科学家，C 语言之父，对 C 语言、其他编程语言和 Unix 操作系统的发展做出了巨大贡献。里奇在哈佛大学学习物理学和应用数学，1967 年进入贝尔实验室，曾担任朗讯科技公司贝尔实验室下属的计算机科学研究中心系统软件研究部主任一职。1983 年，他与肯·汤普逊一起获得了图灵

奖，获奖理由是他们“研究发展了通用的操作系统理论，尤其是实现了 UNIX 操作系统”。1999 年，两人因为发展 C 语言和 Unix 操作系统，一起获得了美国国家技术奖章。

里奇身上有很多可贵的品格：首先，他对所做的事十分感兴趣。比如创造出 Unix 的初衷并非为了挣钱，事实上刚开始是为了省钱，或者将他们的游戏装到一个更省钱的计算机中；第二，敢于进入陌生的领域。里奇原本是一个物理学家和数学家，但是，他却成了最具传奇的程序员。很显然，他的专业背景为他研发出 C 语言或者 Unix 起了很大帮助，正如里奇所言：“要不惧工作在一个陌生的领域里。”如果里奇花了数十年的时间在晦涩的数学上，或许 Unix 就会胎死腹中；第三，有创新思维。里奇利用贝尔实验室的资金、技术、员工等条件，用自己的创新性思维，与他的朋友按照自己的时间安排来研发他们想要的东西；最后，要懂得分享。现在许多企业都喜欢保密，将自己的核心技术藏匿起来，这在里奇看来都是不成功的。

著名的计算机科学家 N.Wirth 评价他说：里奇先生的专业精神令人感动，40 年如一日，在他所从事的领域辛勤耕耘，他的多项发明，包括 C 语言、Unix，也包括 Plan9，无论哪一项，在软件发展史上都有着举足轻重的地位。和他的伟大成就形成对照的是他的行事，态度低调，他的表达就像他的软件一样，简洁生动而准确。

麻省理工大学计算机系的马丁教授评价说：如果说，乔布斯是可视化产品中的国王，那么里奇就是不可见王国中的君主。乔布斯的贡献在于，他如此了解用户的需求和渴望，以至于创造出了让当代人乐不思蜀的科技产品。然而，却是里奇先生为这些产品提供了最核心的部件，人们看不到这些部件，却每天都在使用着。

里奇先生的同事布莱恩·克尼汉评价说：牛顿说他是站在巨人的肩膀上，如今，我们都站在里奇的肩膀上。这句话应该是对丹尼斯·里奇先生的一生最有力也是最中肯的评价。

2011 年 10 月 9 日，丹尼斯·里奇去世，享年 70 岁。

2011 年 10 月 13 日，在众多的国际互动论坛上，计算机爱好者们以特有的方式纪念这位编程语言的重要奠基人。许多网友的发帖中没有片言只字，仅仅留下一个分号“;”。在 C 语言中，分号标志着一行指令语句的结束，网友们以此来悼念“C 语言之父”，美国著名计算机专家丹尼斯·里奇（Dennis Ritchie）。

C 语言是使用最广泛的语言之一，可以说，C 语言的诞生是现代程序语言革命的起点，是程序设计语言发展史中的一个里程碑。自 C 语言出现后，以 C 语言为根基的 C++、Java 和 C# 等面向对象语言相继诞生，并在各自领域大获成功。但今天 C 语言依旧在系统编程、嵌入式编程等领域占据着统治地位。

TIOBE 公司每月发布一次世界编程语言排行榜，该排行榜是根据世界范围内的资深软件工程师、课程和第三方厂商使用某种编程语言的数量，并使用搜索引擎（如 Google、Bing、Yahoo!、百度）以及 Wikipedia、Amazon、YouTube 统计出排名数据，其结果反映

了各种编程语言的流行情况。从排行榜发布之初到现在，C 语言因其突出的优点和广泛的应用一直处于领先地位。

本章自测

本章节中，核心内容包括 C 语言程序框架结构、程序的构成和程序开发过程。通过本单元的学习，我们应该知道：

1. C 语言程序最基本的程序框架由两部分构成，分别是：

① _____

② _____

2. C 程序最大的特点就是所有的程序都是用函数来装配的，函数是构成 C 语言程序的基本单位，函数包括主函数、库函数和自定义函数。函数的一般结构形式为：

3. 标识符是用来标识程序中某个对象名字的字符序列。C 语言把标识符分为三类，即关键字、预定义标识符及用户自定义标识符。对于用户自定义标识符的命名，C 语言规定：

① 所有的用户标识符必须先定义或说明后再使用；

② 用户标识符由_____、_____、_____组成，并且首字符不能是数字；

③ C 语言中区分英文字母大小写；

④ 不能用_____作为用户自定义标识符，通常不使用预定义标识符作为用户自定义标识符。

4. 理论上讲，程序开发过程分为四个步骤，分别为：

① _____

② _____

③ _____

④ _____



5. 使用 VC++ 6.0 开发应用程序的步骤大致为:

总之, 通过本章的学习, 大家应该掌握 C 语言程序框架结构和程序开发过程, 还要对 C 语言程序设计的知识脉络有一定的了解。