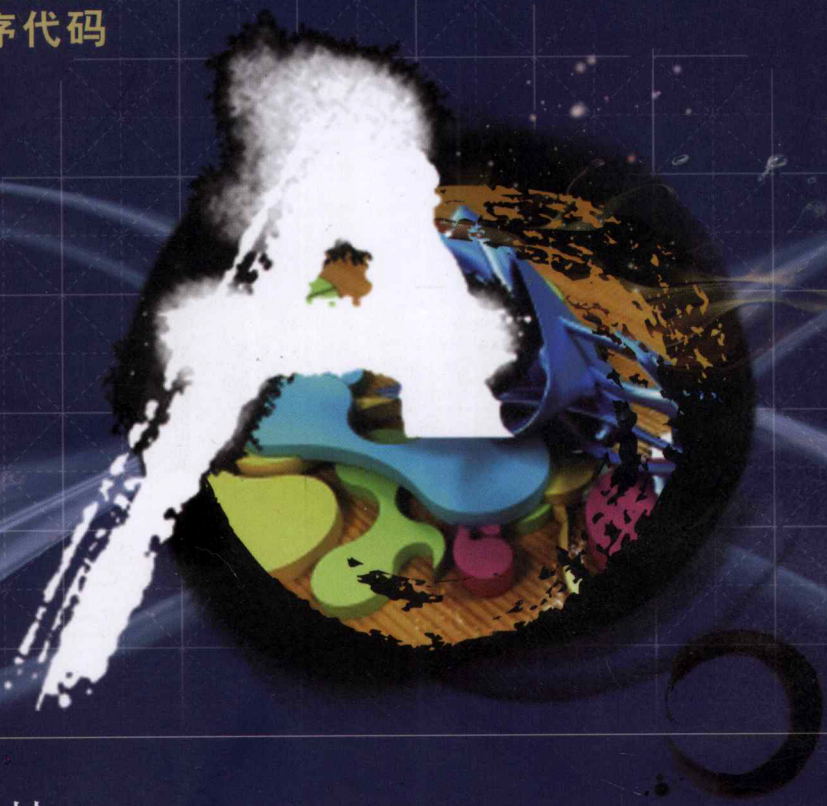
深入介绍C语言基础知识
完整演示程序设计开发实例
配电子课件和源程序代码



高等学校规划教材

C语言与程序设计

◎曹计昌　卢萍　李开　编著

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

高等学校规划教材

C语言与程序设计

曹计昌　卢　萍　李　开　编著

電子工業出版社
Publishing House of Electronics Industry
北京 · BEIJING

内 容 简 介

本书是华中科技大学“C 语言程序设计”精品课程主教材。本书力图完整、全面、清楚地介绍 C 语言的基本语法和语义，同时通过对一些精心提炼的算法实例进行分析和编程设计，培养学生算法设计、程序设计、调试程序的能力。全书分为上、下两篇，共 18 章，另外提供了 3 个附录。

上篇是 C 语言，包括第 1 章至第 10 章，内容主要有概论、基本词法语法规则与程序元素、基本的标准输入与输出、流程控制、函数与程序结构、编译预处理、数组、指针、结构与联合、文件的输入与输出。下篇是程序设计，包括第 11 章至第 18 章，深入介绍了复杂类型的指针、递归、排序、线性数据结构、非线性数据结构、参数数目可变的函数与库函数、图形图像处理程序设计、程序设计开发实例。

为方便教学，本书配有电子课件和源程序代码，任课教师可以登录华信教育资源网（www.hxedu.com.cn）免费注册下载。

本书适合作为创新实验班（ACM 班）和卓越工程师班程序设计课程教材，也适合作为高等院校计算机、通信、电子、自动化等专业和软件学院本科生的教材，或作为研究生入学考试 C 语言与程序设计方面的参考书，亦适合软件开发工程师和广大科技人员自学参考。

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。
版权所有，侵权必究。

图书在版编目(CIP)数据

C 语言与程序设计 / 曹计昌，卢萍，李开编著. —北京：电子工业出版社，2013.1
高等学校规划教材
ISBN 978-7-121-19039-1

I. ①C… II. ①曹… ②卢… ③李… 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 279952 号

策划编辑：索蓉霞
责任编辑：索蓉霞
印　　刷：三河市鑫金马印装有限公司
装　　订：三河市鑫金马印装有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　　邮编：100036
开　　本：787×1092　1/16　印张：28　　字数：806 千字
印　　次：2013 年 1 月第 1 次印刷
定　　价：49.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：(010) 88258888。

前　言

教材建设是课程建设的基石和前提条件。要写好一本教材，需要排除各种干扰，博采众长，更要耐得住寂寞地对教材结构、内容组织、知识领域、重点难点、能力要求、例子习题等潜心揣摩，仔细推敲，精心组织。一本好的教材可以让学生在规定学时内获得更多的知识、培养更强的能力，甚至受益终身，这是本书作者的真实感受。正是本着对课程建设的责任感，虽然水平有限，但作者还是决心尽力写一本既能够深刻阐明C语言的语法语义，又能够涵盖本学科程序设计基础性问题和解决这些问题方法的教材。

《C语言与程序设计》与作者2008年出版的《C语言程序设计》教材相比多了一个“与”字，但教材内容的含义和重点都有很大变化。语言是程序设计的工具，而学习语言的根本目的就是要能够进行程序设计。因此，要使学生既具广度，更具深度地扎扎实实打好C语言基础，能够熟练掌握编程工具；同时要更加突出程序设计，培养学生用C语言编程解决本专业基础性、底层性、系统性问题的能力。

C语言是目前全球范围内流行和使用最为广泛的一种程序设计语言，选C语言作为程序设计的语言工具无疑是合适的。但是，学习C语言的根本目的是希望能够以C语言为工具进行程序设计，用C编程实现计算机的解题算法或解题任务。因此，作者将教材内容分为两大部分，第一部分是C语言篇，第二部分是程序设计篇。作者希望通过C语言篇使读者打下良好的C语言基础，然后通过程序设计篇使读者能够了解和掌握常用的数据结构和各种经典算法，并能够用C编程解决实际问题，尤其是本专业的、底层的、系统的基础性问题。

本书的编写贯彻了教育部高等学校计算机科学与技术教学指导委员会2009年编写的《高等学校计算机科学与技术专业核心课程教学实施方案》中关于程序设计基础课程所提出的教学要求。参考了ACM与IEEE于2012年2月推出的计算机学科教学计划CS2013（Computer Science Curricula 2013），以及C语言标准ISO-IEC 9899:1999（简称C99）和2011年12月发布的最新C语言标准ISO-IEC 9899:2011（简称C11）。虽然C11新标准中的某些语言成分一时得不到某些编译器的支持，但适当介绍会有助于拓展学生的视野。在2012年7月发布的开源Pelles C编译器能够支持C11的某些语言成分。作者试用了一下，觉得虽然该编译器调试等功能不强，但毕竟可以验证C11的某些新增语言成分。建议有兴趣的读者不妨下载试试，下载Pelles C编译器的网址是：http://www.smorgasbordet.com/pellesc/。

本书的编写坚持知识结构的完整性，没有受篇幅、学时限制。课程目标是使学生能够熟练掌握C语言，并具备用C编程解决实际问题（算法分析与程序设计）的能力。因此，**本书编写一方面完整、全面、清楚地介绍了C语言的基本语法和语义，同时，通过对一些精心提炼的算法实例进行分析和编程设计，培养学生算法设计、程序设计、调试程序的能力。**

教材编写按照两步法和ABC三层次法组织内容。

两步法是指：第一步通过第一篇打好C语言基础，第二步通过第二篇落实到程序设计和编程能力。要既注重“开车”，即清楚地介绍C语言的基本语法和语义，学好用好C；又强调“造车”，即语言成分、编译系统、标准库函数的设计实现问题。如果按照64学时考虑，可以用24～32学时初步处理C语言的语法规则问题，再用32～40学时处理程序设计问题。

ABC三层次法指在章节内容的组织上按照ABC三个层次进行组织；即：按照不加“*”号的A层，加一个“*”号的B层，以及加两个“**”号的C层，将教材内容分为基本、中等、提高三个层面并安排贯彻到全书的各个章节。

基本层面是书中不加“*”号的内容，属于学习C语言必须掌握的部分，它可以满足32学时的教学要求。中等层面指书中标识有一个“*”号部分的内容，基本层面和中等层面的合成内容属于计算机科学与技术专业本科教学必须讲授的内容，它可以满足52学时至64学时的教学要求。至于加两个“**”号的提高部分，则是供学有余力的学生进一步提高所用，教师在课堂上不必讲授或有选择性讲解部分例子的算法思想，让学有余力的学生自学提高。任课教师可以根据情况对教材内容进行裁剪。

作者在编写此书过程中提炼了自己在长期教学过程中的实践经验、例子程序，并参考了国内外不少的教科书，希望此书能够具备以下特点：

1．在教材内容方面，能够突出计算机专业的学科特点，要与非计算机专业的教材有明显区别。同时，也希望使非计算机专业的学生感到有必要学习此书以提高自己的C语言和程序设计的能力与水平。

2．完整且有一定深度地介绍了C语言的各类数据类型。除基本数据类型，数组、结构、联合、字段结构等构造类型外，突出介绍了各类指针、指针与数组的关系、类型表达式、复杂说明等难度较高的数据类型。不回避难点，让学生掌握克服难点的方法，才能给学生打下坚实的语言基础。

3．从编程使用的角度比较系统地介绍了各种数据结构。如动态数组、单向链表、双向链表、十字交叉链表、堆栈、队列、散列函数（哈希函数）与查找表、值栈、逆波兰表达式生成与求值、二叉树、堆与优先队列、图等。因为这些是程序设计必需的基本数据结构，程序设计说到底就是数据结构加算法。因此学习通过C的实例来描述和实现这些数据结构，可以给学生打下扎实的程序设计所需的数据结构基础。

4．通过ABC法既有广度又有深度地介绍了各种经典算法。如暴力求解法（枚举法）、筛法、递推、排序、递归、回溯、分治、动态规划，蒙特卡罗法、高精度计算、贪心、深度优先搜索、广度优先搜索、最小生成树、最短路径算法等，目的是让本课程教学覆盖程序设计涉及的基本算法，给学生打下扎实的程序设计所需的基本算法基础。

5．本书适当强调“造车”，力求解决本专业的基础性问题。通俗地说，学好C语言，使用好C这个编程工具属于“开车”问题；例如，能够用scanf、printf函数进行格式化输入/输出属于“开车”问题。而了解语言本身的构造、编译方面的实现过程、标准库函数的设计方法等则属于“造车”问题，如能够进行scanf、printf函数的设计就属于“造车”问题。

因此，本书加强了如C的字符串库函数设计、数字串与数之间的转换函数设计、printf和scanf等I/O库函数设计、简单宏替换的实现、模拟串行进位的任意位数超长数据的加法运算、以函数指针为成员的结构设计等内容。希望使读者不仅能“开好车”，而且能够具备一定的“造车”基础和能力，即用C编程解决本专业的基础性问题的程序设计能力，为培养学生今后写底层程序、系统程序奠定扎实基础。

6．适当强调了形式化和推理。如BNF范式、语法图，算法的伪码描述，通过标准中E1[E2]与*((E1)+(E2))等价的规则，推导多维数组元素的指针表示，通过类型说明符的优先级和结合性及类型表达式推导解释各种复杂声明等。目的是培养学生的抽象思维能力、形式化的分析推理能力，以及精确的理解C的语法语义的能力。

7．适应网络和多媒体发展的趋势，介绍了图形和图像处理。包括位图文件格式、位图图像的灰度变换、二值变换等操作，以及OPENCV图形图像库；通过例子和习题介绍了统一资源定位符URL（也称为网页地址）的散列函数实现快速查找的方法；而教材中介绍的最短路径算法就是学生今后学习计算机网络时进行路由表设计的基础。

8．有意识地将经典算法和实用问题引入教材的正文或习题。如八皇后问题、0-1背包问题、装载问题、挖地雷问题、填字游戏、骑士游历问题、球钟问题、农夫过河问题、回文构造问题、迷宫

试读结束：需要全本请在线购买： www.ertongbook.com

问题、小球下落问题、格尼斯堡七桥问题、地铁换乘问题等，目的是训练学生算法分析和实现的能力，同时也为人工智能、近似推理、机器学习等后续课程打下初步的基础。

9．体现了最新 C 语言标准 ISO/IEC 9899:2011 中新增语言成分。如新增多线程环境下原子类型修饰符_Atomic，对象的对齐（alignment of objects），支持 Unicode 字符集的 char16_t、char32_t 的类型等；还有静态断言(static assertions)，无返回函数（no-return functions），新的独占模式的 fopen 函数，因为安全原因删除了 gets() 函数，用新的更安全的函数 gets_s()替代，以及函数的边界检查接口等。同时，在教材中尽可能直接引用最新 C 语言标准规定的语法和语义。

全书分为上、下两篇，共 18 章，另外提供了 3 个附录。

上篇是 C 语言，包括第 1 章至第 10 章，内容主要有概论、基本词法语法规则与程序元素、基本的标准输入与输出、流程控制、函数与程序结构、编译预处理、数组、指针、结构与联合、文件的输入与输出。

下篇是程序设计，包括第 11 章至第 18 章，深入介绍了复杂类型的指针、递归、排序、线性数据结构、非线性数据结构、参数数目可变的函数与库函数、图形图像处理程序设计、程序设计开发实例。

对 32 学时课程教学，建议略去全部加 “*”和“**”的内容及第 10 章和第 11 章；学时安排方面建议第 1、6、12、13 章各用 1 学时，第 3 章用 2 学时，第 4、5、7、8、9 章各用 4 学时，第 2 章用 6 学时。

对 56～64 学时课程教学，除讲授上面 32 学时包括的内容外，“*”部分建议讲授，可略去加“**”内容；学时安排方面建议第 6 章用 1 学时，第 3、11、16、17、18 章各用 2 学时，第 1、12、13 章各用 3 学时，第 5、7、9、10、15 章各用 4 学时，第 2、4、8、14 章各用 6 学时。如果专门安排有课程设计教学环节，则可以将第 10 章的部分或全部内容的讲授安排到课程设计环节中，节余下来的学时可以充实到其他章节的讲授。

为方便教学，本书配有电子课件和源程序代码，任课教师可以登录华信教育资源网（www.hxedu.com.cn）免费注册下载。

本书适合作为创新实验班（ACM 班）和卓越工程师班程序设计课程教材，也适合作为高等院校计算机、通信、电子、自动化等专业和软件学院本科教材使用，或作为研究生入学考试 C 语言与程序设计方面的参考书，亦适合软件开发工程师和广大科技人员自学参考。

本书作者都是长期从事创新实验班（ACM 班）、卓越工程师班和院内其他本科班级 C 语言程序设计课程教学，以及以 C 语言为工具从事科学研究和项目开发的教师，作者的学生不少都是全国软件大赛（C 语言本科组）特等奖、一等奖等奖项的得主。本书第 1、7、8、9、10、11、14、15 章由曹计昌老师编写；第 2、5、6、12、13、17 章，以及附录 1、2、3 由卢萍老师编写；第 3、4、16、18 章由李开老师编写；曹计昌老师制订了本书的编写大纲，并对全书进行了统稿。

本书在编写过程中，参考了国内外不少的优秀教材，得到了华中科技大学、计算机科学与技术学院有关领导与专家，以及电子工业出版社的领导和索蓉霞编辑的大力支持与帮助，并得到了华中科技大学教材基金的支持，在此一并致谢。

由于编者水平有限，书中缺点与错误在所难免，敬请有关专家和读者批评指正。

编　者

于华中科技大学

目　录

上篇　C 语言

下篇 程序设计

试读结束：需要全本请在线购买： www.ertongbook.com

上篇 C 语言

第 1 章 概 论

本章首先介绍程序设计语言、程序及程序设计的概念，接着介绍学习 C 语言程序设计的第一个例子。然后介绍 C 语言的产生与发展、C 语言的语言特征、C 语言的标准、计算机系统及内存编址、数和字符的编码表示，以及数在内存中的表现形式、算法的概念及其表示方法。最后介绍原码、补码、反码及其相关运算。

1.1 程序设计语言与程序设计

了解什么是程序设计语言，了解什么是程序，了解什么是程序设计，就是了解本门课程的基本研究对象。这些对于学习语言和程序设计来讲是非常必要的。

严格来说，计算机语言包括机器语言、汇编语言和高级语言这三类语言。如果不涉及汇编语言，程序设计语言往往就是指高级语言。高级语言将面向问题的数据类型的概念引入程序设计，通过将数据分类成为字符型、整型、浮点型等不同的类型，来刻画、描述不同类型的数据。高级语言产生、发展、演变，各种各样高级语言的兴起，实质上就是高级语言数据类型的不断完善、不断扩充、不断复杂多样，以及对客观实体描述能力不断增强的一个过程。

如果认为高级语言就是我们所要讨论的程序设计语言，那么什么是程序设计语言？正如将物体向不同平面投影可以得到不同的平面图形一样，不同的人从不同的角度对程序设计语言有不同的理解。计算机的使用者认为程序设计语言是操纵计算机的工具；程序员则认为它是程序员之间的相互通信和交流的方法；喜欢数学和算法的人则认为它是算法的符号表示。按照 Ravi Sethi 的观点，一门通用的程序设计语言应该是能够为各种各样的用户都能提供服务的语言。尽管对程序设计语言的理解和定义多种多样，但是按照一般比较流行的观点，可以认为：**程序设计语言由一些符号构成，这些符号被用于定义、组织并完成各种各样的计算任务。**

人类所使用的语言称为自然语言，它是以语音为物质外壳、以词汇为建筑材料、以语法为结构规律而构成的体系。与此类似，可以将程序设计语言定义为：**程序设计语言是以具有特定语义的符号为基本构成单位、以语法为程序构成规律，专门用于定义、组织并完成各种各样的计算任务而形成的体系。**

程序是用程序设计语言表示的计算机解题算法或计算机解题任务。程序设计是将解题任务转变成程序的过程。Nell Dale 等人指出：程序就是要求计算机执行的指令序列。程序设计就是如何计划、安排计算机必须遵循的操作步骤及顺序的过程。

1.2 学习 C 语言程序设计的第一个例子

C 语言是目前全球流传最广、使用最多的程序设计语言。C 语言并不神秘，学习 C 语言的最好办法就是在掌握 C 语言的一般语法和几种基本语句的语义基础之上，自己动手编写程序。先从编写

简单的程序入手，然后逐步深入，这样就可以自然地学会 C 语言，并且运用 C 语言进行程序设计。另外，阅读他人编写的程序，也是提高自己 C 语言编程能力的一个好办法。

1.2.1 创建并运行第一个 C 程序

下面先看一个用 C 语言编写的程序。

【例 1.1】 用 C 语言编写的第一个程序示例。输入自己的名字的汉语拼音，要计算机问候自己，并且输出这是自己学习 C 语言的第一个程序的句子。程序如下：

```
#include "stdio.h"                    /*文件包含编译预处理命令，要求将 stdio.h 包含到程序中*/
void show(char str[]);                /*函数原型说明语句，show 是函数名，str 是字符数组形参*/
int main(void)                        /*主函数，无参，返回值类型为整型*/
{//main 函数函数体开始
    char name[20];                    /*声明语句，声明一个以 name 为数组名有 20 个字节的
                                        字符数组*/
    printf("Input your name please!\n");/*输出一个提示信息，\n 用于控制换行*/
    gets(name);                       /*读取用户从键盘输入一个字符串（即用户名）到 name
                                        字符数组中*/
    printf("Hello %s!\n",name); /*先输出 Hello，再输出 name 中字符（即用户名）*/
    show(name);                       /*以 name 为实参调用 show 函数，程序转 show 执行*/
    return 0;                         /*返回语句，从 main 函数返回系统，0 是返回值*/
}//main 函数函数体结束
void show(char str[])                 /*show 函数的头部，进入 show 函数后 str 可代替
                                        name 数组*/
{//函数体开始，下面调用 printf，双引号内字符按原样输出，但%s 将输出 str 内容
    printf("This is the first program for %s to learn C programming!\n",str);
}// show 函数函数体结束
```

读者通过双击 Visual C++ 6.0（简称 VC 6.0）图标打开 VC 6.0 界面进行操作，如图 1.1 所示。

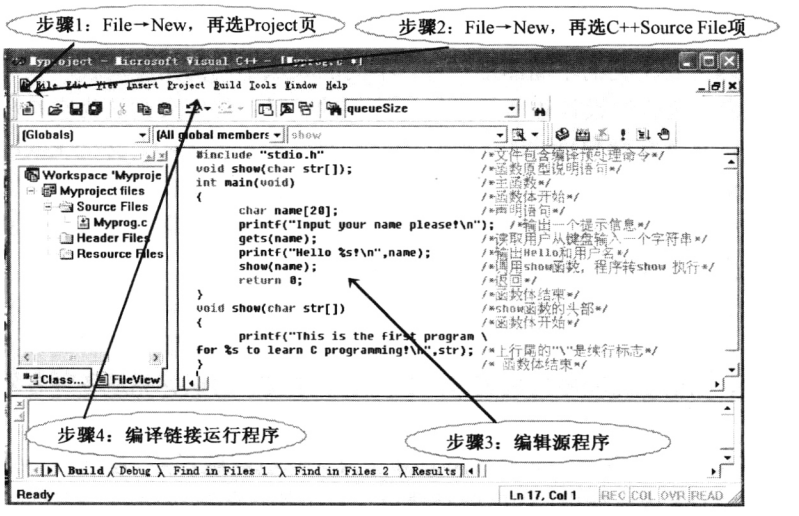


图 1.1 使用 Visual C++ 6.0 编程的示意图

步骤 1：选择 File 菜单，再选下拉菜单中的 New，在 New 窗口中的 Project 页（默认）中选择 Win32 Console Application（若选中颜色变蓝），并且在 Project Name 编辑框中输入工程名，如 Myproject，此时 Location 编辑框中将会显示 D:\Myproject，它表示在 D 盘上将创建 Myproject 工程的目录，相关的所有文件将都会存放在该目录下；单击右边带 3 个点的按钮可以改变 Myproject 工

程的存放路径。单击 OK 按钮，在打开的窗口中单击 An empty project（默认），再单击 Finish 按钮，在弹出的窗口中单击 OK 按钮。此时 VC 6.0 将创建一个空的工程。

步骤 2：单击 FileView，选中 Source Files（颜色变成蓝色），选择 File 菜单，再选下拉菜单中的 New，单击 C++ Source File（颜色变成蓝色），在 File 编辑框中输入 Myprog.c（也可以输入其他名字，但是一定以.c 为扩展名），单击 OK。

步骤 3：打开 Source Files 目录，可以看见 Myprog.c 文件，双击 Myprog.c，在右侧文本编辑区域输入例 1.1 中的源程序，进行源程序编辑。

步骤 4：选择 Build 菜单，再选下拉菜单中的 Rebuild All。此时 VC 6.0 编译器将对源程序进行编译、链接，并且生成名字为 Myproject.exe 的可执行文件。再选 Build，再选下拉菜单中的 Execute Myproject.exe 运行 Myproject.exe 程序，此时在弹出的窗口中出现屏幕提示：

```
Input your name please!
```

输入自己姓名的汉语拼音并回车

Jichang Cao

计算机就会接受你输入的姓名并且继续运行，得到运行结果。可以看到屏幕上显示的程序运行中的人机交互与运行结果如下（带下画线者是输入的信息）：

Input your name please!

Jichang Cao

Hello Jichang Cao!

This is the first program for Jichang Cao to learn C programming!

1.2.2　解释分析第一个 C 程序

在上面的 C 程序中，“/*”和“*/”之间的文字，以及“//”后的文字表示注释，是对程序的解释说明，以提高程序的可读性。例 1.1 程序中的第 1 行和第 2 行都是与 C 编译器打交道的语句。其中#include "stdio.h"称为编译预处理语句。#include 是文件包含编译预处理命令，双引号中的 stdio.h 是 C 编译系统中标准的输入和输出头文件的文件名。该语句的含义是要求将标准的输入和输出头文件 stdio.h 包含到本程序中。

而 void show(char str[]);称为函数原型说明语句，该语句通知编译器，本程序有一个名字为 show 的函数。同时说明该函数有一个由 char str[]说明的字符数组作为形式参数，该函数的返回值的类型为 void（无值型）。编译器在处理其后关于 show 函数的定义与调用时，都将以 show 的函数原型作为标准形式来判断关于 show 函数的定义与调用是否合法。

例 1.1 中另一个函数是 main 函数，它是程序中的主函数。C 语言规定，一个 C 程序可以由一个或多个函数组成，但有且仅有一个 main 函数。程序的执行总是从 main 函数开始。C 语言标准建议 main 函数可以返回一个整型值（shall be return an integer value）。

从 main 函数后面的左花括号开始，到对应的右括号为止的部分称为 main 函数的函数体。main 函数体内的 char name[20];语句是一个说明语句，它说明 name 标识了一个占有 20 个字节的字符数组，name 是该数组的数组名。接着，printf("Input yore name please!\n");语句输出一个提示信息，要求用户从键盘上输入自己的名字。\n 用于控制换行。gets(name);语句读取用户从键盘输入的一个字符串（即用户自己的名字），并且将该字符串存放在字符数组 name 中。printf("Hello %s!\n",name);语句先输出 Hello，然后输出 name 字符数组中的各个字符。%s 是格式说明，意思是输出 name 标识的字符串。在 show(name);语句中发生了对函数 show 的调用。在调用过程中，name 首先将其值复制到 str 中，使 str 也对应着用户输入的名字，然后程序转移到 show 函数中执行。从 void show(char

str[])后面的左花括号开始，到对应的右括号为止的部分称为 show 函数的函数体。show 函数中通过调用 printf 函数在屏幕上先输出“This is the first program for”，然后输出所输入的名字，再输出“to learn C programming!”。最后的 return 0;表示向操作系统返回一个 0 值。

1.3 C 语言的产生、发展与语言特征

C 语言是目前全球范围内流行和使用最为广泛的一种程序设计语言。C 语言的产生有其深刻的技术与应用需求背景。

1.3.1 C 语言的产生与发展

在 20 世纪 60 年代，FORTRAN、ALGOL、COBOL、LISP 这样一些面向过程的语言较好地解决了科学计算、商用程序、表处理方面的应用问题。但是，由于这些语言都是面向特定应用的，缺乏对机器硬件的直接操作与控制能力。另外，汇编语言虽然具有良好的硬件操纵能力，然而其面向机器的基本特性，对硬件指令系统的依赖性，仅依赖助记符来描述机器操作使之缺乏面向问题的数据抽象能力和算法描述能力，这样一些不足，使得用汇编语言编程的难度大、效率（劳动生产率）低。

因此，能否设计出一种既具有面向过程的特性，又具有汇编语言特点的新程序设计语言，就是当时计算机界迫切需要解决的问题。它要求这种新的程序设计语言既应具有一些高级语言的特点，如较好的面向问题的数据抽象能力和算法描述能力；同时也应具有一些汇编语言的特征，如能够较好地对机器硬件进行直接操作与控制。C 语言就是在这种背景下产生的。

程序设计语言的发展与演变体现出继承与创新。C 语言源于 B 语言，B 语言源于 BCPL 语言，BCPL 语言源于 CPL 语言，CPL 语言源于 ALGOL 60 语言，而 ALGOL 60 语言源于 FORTRAN 语言。语言之间的这种继承与丰富、发展与完善的关系可以由图 1.2 来描述。

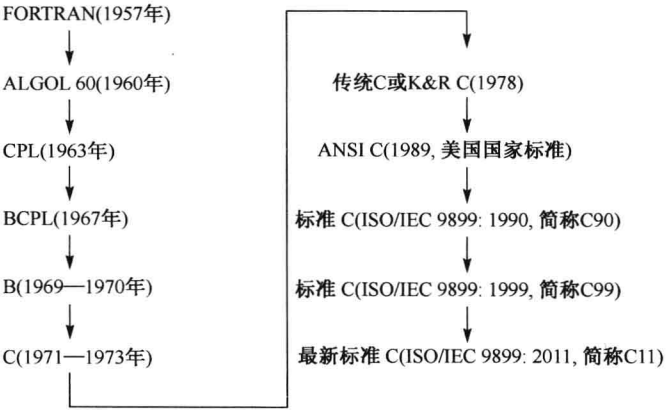


图 1.2 C 语言的继承、产生与发展历程

C 语言的最早起源可以追溯到 1957 年产生的 FORTRAN 语言。FORTRAN 语言成功的两个最重要的特征是：程序的编写接近人类使用的自然语言和数学公式；同时，编译后产生的目标代码的执行速度可以与手工用汇编语言编写的程序所产生的目标代码的执行速度相抗衡。打破了 20 世纪 50 年代人们广泛认为的“高级语言源程序编译后产生的目标代码的执行效率不及手工编程的代码执行效率”的观点。

FORTRAN 语言成功后，人们又想设计一种通用语言（Commom Language），目标是实现程序

共享，使公众（一般人员）能够用该语言编程进行数值处理。而 1960 年产生的 ALGOL 60 就是这种设想的产物。

由于 ALGOL 60 缺乏对计算机硬件的操作能力，因此不宜用来编写系统程序。英国剑桥大学（Cambridge University）于 1963 年基于 ALGOL 60，设计了 CPL（Combined Programming Language）语言，目的是希望继承 ALGOL 60 的高级语言特征，但又能够对机器硬件进行操作。但 CPL 语言过于复杂，语言规模过大，学习和使用 CPL 比较困难，因此 CPL 并没有完全实现。

针对 CPL 的弱点，英国剑桥大学的 Martin Richards 在美国麻省理工学院访问工作时，对 CPL 语言进行了简化。于 1967 年提出了简化的 CPL 语言 BCPL（Basic Combined Programming Language）语言。BCPL 简洁、紧凑，除了诸如 word（字）、cell（单元）这样一些简单数据类型的描述之外，没有其他数据类型。因此，BCPL 属于无数据类型语言。BCPL 于 20 世纪 70 年代初在牛津（Oxford）的 OS6 操作系统，以及 MIT 和贝尔实验室的一些项目中得到了应用。

1969—1970 年，美国贝尔实验室的 Ken Thompson 为了使 DEC 公司的 PDP-7 在每个字占 18bit 且只有 8KB 内存空间的硬件环境下，实现 NUIX 操作系统，对 BCPL 进行了进一步的简化和改进，设计出了 B 语言，使其能够在 8KB 的内存空间中运行。同时，对 BCPL 语言成分的描述进行了改进。例如，用 V[i]表示内存中 V 标识的单元后面的第 i 个单元，而在 BCPL 中，V 后面的第 i 个内存单元用 V!i 表示。显然 V[i]较 V!i 更接近数学上数组 V 中第 i 个元素的描述，并且 B 语言中开始引入了一些新的操作。又如，++和– –操作，并且有前缀++和– –操作与后缀++和– –操作之分。在数据类型方面，B 语言和 BCPL 一样，仍然属于无数据类型语言。Dennis M. Ritchie 称 B 语言为无类型的 C 语言。

1969—1973 年，贝尔实验室的 Dennis M. Ritchie 为了在 DEC 公司的 PDP-11 上实现 UNIX 操作系统，在 B 语言的基础上设计并实现了 C 语言。Dennis M. Ritchie 首先引入了类型系统，接着定义了相关语义，并为这种新语言编写了编译器，并将改进后的语言命名为 C 语言，而 C 取自 BCPL 的第二个字母，用以表示 C 源于 B，而 B 源于 BCPL。此后，Mike Lesk 和 John Reiser 进一步完善了 C 编译预处理中的宏定义和条件编译。

1973 年，Dennis M. Ritchie 和 Ken Thompson 合作，用 C 语言重写了 UNIX 操作系统的内核，先后推出了 UNIX V5 和 UNIX V6，此时的 C 语言还是附属于 UNIX 操作系统的。从 1973—1977 年，用 C 编写的 UNIX 操作系统被先后移植到 IBM 360/370、DEC VAX 11/780、AT&T 等机器上。1978 年，Brian Kernigha 和 D. M. Ritchie 以 UNIX V7 的 C 编译程序为基础，写出了被称为“C 语言的圣经”且影响深远的名著“The C Programming Language”，书中介绍的 C 语言被称为 K&R C，或传统 C。这对 C 语言的普及和推广起到了不可估量的作用。自 1978 年以后，C 语言被移植到各种类型的计算机上，独立于 UNIX 和其他操作系统。

1.3.2　C 语言的标准化

1978 年 UNIX V7 和 UNIX System V 相继推出，使 UNIX 操作系统得到了广泛的使用。伴随 UNIX 取得的巨大成功，C 语言的突出优点也被人们广泛认同，这些都极大推动了 C 语言的普及和推广，同时也导致各种各样 C 语言版本的出现。以 1978 年 K&R C 为代表的 C 语言被称为传统的 C 语言，实事求是地说，它在许多语言细节方面尚显得不够精确。于是 ANSI 在 1983 年成立了 C 语言标准化委员会 X3J11，负责 C 语言的标准化。X3J11 于 1989 年底公布美国第一个 C 语言国家标准 ANSI 89，简称 C89。

次年，即 1990 年，国际标准化组织 ISO 将其接受为 C 语言的国际标准，称为 ISO/IEC 9899-1990。它是 C 语言的第一个国际标准，也称为标准 C，简称 C90。

在 20 世纪 90 年代，随着对宽字符集支持需求的出现，对整型数据和浮点数据表示范围扩大需

试读结束：需要全本请在线购买：　www.ertongbook.com