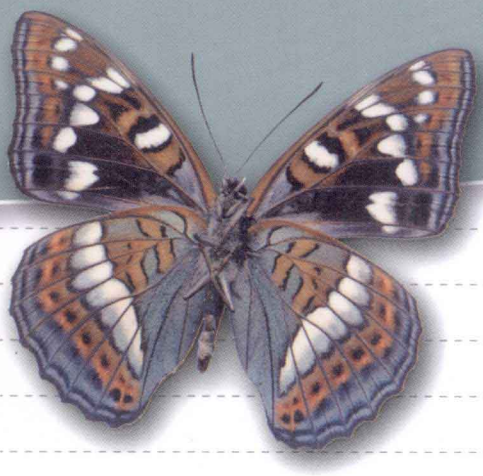
21世纪 高等学校计算机系列规划教材

C语言程序设计



张志强 周克兰 主编
杨季文 主审

清华大学出版社

21 世纪高等学校计算机系列规划教材

C 语言程序设计

张志强　周克兰　主编
杨季文　主审

清 华 大 学 出 版 社
北　京

内容简介

本书全面、系统地介绍了C语言程序设计的基本概念、语法和编程方法。全书共分为10章，每个章节都从实际应用出发，蕴含了作者丰富的教学经验和编程心得。第1章通过一个简单C语言程序的编写、编译和运行介绍了程序设计的基本概念；第2章从计算机内数据存储的角度介绍了C程序中包括指针在内的数据类型的概念及使用方法；第3章介绍了包括指针运算在内的C语言提供的各种运算功能；第4章讲述了结构化程序设计的方法；第5章结合循环与指针，讲述了使用数组处理大量数据的方法；第6章讲述了包括结构体在内的各种自定义数据类型的使用方法；第7章讲述了使用函数进行模块化程序设计的方法，并重点讲述了指针在函数参数中的作用；第8章讲述了使用指针操作动态内存的方法及链表基本应用方法；第9章讲述了在C语言中处理文件的基本方法；第10章介绍了C的预处理命令及简单应用。

全书内容由浅入深，例题经典、丰富，将指针的应用融合到全书的各章节之中。本书结构新颖、紧凑、内容通俗易懂，是学习C语言的合适教材。本书既可以作为普通本科院校、普通高等专科学校的计算机教材，也可以作为计算机培训和计算机等级考试辅导的教学用书。

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/张志强等主编.—北京：清华大学出版社，2011.1
(21世纪高等学校计算机系列规划教材)
ISBN 978-7-302-24601-5

Ⅰ.①C…　Ⅱ.①张…　Ⅲ.①C语言—程序设计　Ⅳ.①TP312

中国版本图书馆CIP数据核字(2011)第002692号

责任编辑：魏江江
责任校对：焦丽丽
责任印制：李红英
出版发行：清华大学出版社　　**地　址**：北京清华大学学研大厦A座
http://www.tup.com.cn　　**邮　编**：100084
社 总 机：010-62770175　　**邮　购**：010-62786544
投稿与读者服务：010-62795954，jsjjc@tup.tsinghua.edu.cn
质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 刷 者：北京密云胶印厂
装 订 者：三河市新茂装订有限公司
经　销：全国新华书店
开　本：185×260　**印　张**：16.5　**字　数**：400千字
版　次：2011年2月第1版　　**印　次**：2011年2月第1次印刷
印　数：1～3000
定　价：29.00元

产品编号：038852-01

编审委员会成员

（按地区排序）

清华大学	周立柱	教授
	覃　征	教授
	王建民	教授
	冯建华	教授
	刘　强	副教授
北京大学	杨冬青	教授
	陈　钟	教授
	陈立军	副教授
北京航空航天大学	马殿富	教授
	吴超英	副教授
	姚淑珍	教授
中国人民大学	王　珊	教授
	孟小峰	教授
	陈　红	教授
北京师范大学	周明全	教授
北京交通大学	阮秋琦	教授
	赵　宏	教授
北京信息工程学院	孟庆昌	教授
北京科技大学	杨炳儒	教授
石油大学	陈　明	教授
天津大学	艾德才	教授
复旦大学	吴立德	教授
	吴百锋	教授
	杨卫东	副教授
同济大学	苗夺谦	教授
	徐　安	教授
华东理工大学	邵志清	教授
华东师范大学	杨宗源	教授
	应吉康	教授
上海大学	陆　铭	副教授
东华大学	乐嘉锦	教授
	孙　莉	副教授

浙江大学	吴朝晖	教授
	李善平	教授
扬州大学	李　云	教授
南京大学	骆　斌	教授
	黄　强	副教授
南京航空航天大学	黄志球	教授
	秦小麟	教授
南京理工大学	张功萱	教授
南京邮电学院	朱秀昌	教授
苏州大学	王宜怀	教授
	陈建明	副教授
江苏大学	鲍可进	教授
中国矿业大学	张　艳	副教授
武汉大学	何炎祥	教授
华中科技大学	刘乐善	教授
中南财经政法大学	刘腾红	教授
华中师范大学	叶俊民	教授
	郑世珏	教授
	陈　利	教授
江汉大学	颜　彬	教授
国防科技大学	赵克佳	教授
	邹北骥	教授
中南大学	刘卫国	教授
湖南大学	林亚平	教授
西安交通大学	沈钧毅	教授
	齐　勇	教授
长安大学	巨永锋	教授
哈尔滨工业大学	郭茂祖	教授
吉林大学	徐一平	教授
	毕　强	教授
山东大学	孟祥旭	教授
	郝兴伟	教授
中山大学	潘小轰	教授
厦门大学	冯少荣	教授
仰恩大学	张思民	教授
云南大学	刘惟一	教授
电子科技大学	刘乃琦	教授
	罗　蕾	教授
成都理工大学	蔡　淮	教授
	于　春	讲师
西南交通大学	曾华燊	教授

此为试读，需要完整PDF请访问：www.ertongbook.com

出版说明

随着我国改革开放的进一步深化，高等教育也得到了快速发展，各地高校紧密结合地方经济建设发展需要，科学运用市场调节机制，加大了使用信息科学等现代科学技术提升、改造传统学科专业的投入力度，通过教育改革合理调整和配置了教育资源，优化了传统学科专业，积极为地方经济建设输送人才，为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是，高等教育质量还需要进一步提高以适应经济社会发展的需要，不少高校的专业设置和结构不尽合理，教师队伍整体素质亟待提高，人才培养模式、教学内容和方法需要进一步转变，学生的实践能力和创新精神亟待加强。

教育部一直十分重视高等教育质量工作。2007 年 1 月，教育部下发了《关于实施高等学校本科教学质量与教学改革工程的意见》，计划实施“高等学校本科教学质量与教学改革工程（简称‘质量工程’）”，通过专业结构调整、课程教材建设、实践教学改革、教学团队建设等多项内容，进一步深化高等学校教学改革，提高人才培养的能力和水平，更好地满足经济社会发展对高素质人才的需要。在贯彻和落实教育部“质量工程”的过程中，各地高校发挥师资力量强、办学经验丰富、教学资源充裕等优势，对其特色专业及特色课程（群）加以规划、整理和总结，更新教学内容、改革课程体系，建设了一大批内容新、体系新、方法新、手段新的特色课程。在此基础上，经教育部相关教学指导委员会专家的指导和建议，清华大学出版社在多个领域精选各高校的特色课程，分别规划出版系列教材，以配合“质量工程”的实施，满足各高校教学质量和教学改革的需要。

本系列教材立足于计算机公共课程领域，以公共基础课为主、专业基础课为辅，横向满足高校多层次教学的需要。在规划过程中体现了如下一些基本原则和特点。

（1）面向多层次、多学科专业，强调计算机在各专业中的应用。教材内容坚持基本理论适度，反映各层次对基本理论和原理的需求，同时加强实践和应用环节。

（2）反映教学需要，促进教学发展。教材要适应多样化的教学需要，正确把握教学内容和课程体系的改革方向，在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养，为学生的知识、能力、素质协调发展创造条件。

（3）实施精品战略，突出重点，保证质量。规划教材把重点放在公共基础课和专业基础课的教材建设上；特别注意选择并安排一部分原来基础比较好的优秀教材或讲义修订再版，逐步形成精品教材；提倡并鼓励编写体现教学质量和教学改革成果的教材。

（4）主张一纲多本，合理配套。基础课和专业基础课教材配套，同一门课程可以有针对不同层次、面向不同专业的多本具有各自内容特点的教材。处理好教材统一性与多样化，基本教材与辅助教材、教学参考书，文字教材与软件教材的关系，实现教材系列资源配套。

(5) 依靠专家，择优选用。在制定教材规划时依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时，要引入竞争机制，通过申报、评审确定主题。书稿完成后要认真实行审稿程序，确保出书质量。

繁荣教材出版事业，提高教材质量的关键是教师。建立一支高水平教材编写梯队才能保证教材的编写质量和建设力度，希望有志于教材建设的教师能够加入到我们的编写队伍中来。

21世纪高等学校计算机系列规划教材

联系人：魏江江 weijj@tup.tsinghua.edu.cn

几十年来，C语言从其诞生之日起，就作为应用最为广泛的程序设计语言长盛不衰，从家用电器中的单片机到企业生产设备中的工业控制系统，从汽车中的车载电脑到轮船、宇宙飞船中的控制系统，从家用电脑到巨型机，可以说只要有计算机的地方，C语言都是最重要的程序设计语言。

本教材主要作者为一直工作于教学一线的大学教师，承担“C语言程序设计”课程的教学任务十余年，有着丰富的教学经验，同时长期从事C语言编程工作，有几十万行程序代码的开发经验，开发的软件多次获得省级、市级的奖励。在教学实践中，作者感受最深的就是，学生普遍反映C语言难学难懂，尤其是指针千变万化，难以捉摸，而事实上，只要遵照一定的学习规律，C语言并不难掌握。

部分C语言教材过细的内容组织让学生迷失了方向。看着满篇的烦琐内容，读者根本不明白该学什么，更不明白C语言的重点是什么，唯一的感觉就是C语言难而烦。本书力求做到去繁就简，以弄懂基本的、主要的、核心的内容为重点，教材也紧紧围绕循环、函数、指针等核心内容进行组织。

另外，本教材也特别强调实践能力的培养。上机是学习C语言的最好方法，读者在学完第1章后就可以开始上机练习。书中每章的内容都包含大量的实例，课后练习也以编程为主，课后练习的内容由浅入深，如果能做到最后一道题目，本章的内容即可基本掌握。

本书第1章阐述了计算机及程序的基本工作原理，C语言程序是怎样运行的，并通过一个完整的C程序例子介绍了C程序的各部分组成、功能及C程序编辑、编译、运行的方法。通过本章的学习，使读者能够对C语言程序及程序设计的过程和方法有一个基本的认识。

第2章用了大量的篇幅介绍计算机中数据的存储方式，从而引入了整数、浮点数、指针等数据类型的概念，并使读者理解这些数据类型的作用和意义。在本章最后介绍了C语言输入、输出的基本方法，使读者马上可以通过这些输入、输出功能，对刚学到的各种不同类型数据进行比较和分析。

第3章讲解了C语言中可以使用的各种运算符，并通过大量的实例来展示这些运算符的功能。本章还详细讲解了指针的各种运算方法，为以后指针的使用做好准备。

通过前面3章的学习，读者已经掌握了C语言中各种基本数据类型数据的处理方法。

第4章的内容通过讲解顺序结构、选择结构、循环结构这三种程序结构，使读者可以编写出具有一定实用功能的程序。本章的难点是循环，尤其是多重循环，这也是很多学生在学习C语言过程中第一次开始掉队的地方，解决这个难点的唯一方法就练习，反复地编程实践。

第 5 章讲述数组，使 C 语言程序可以处理大量的数据，数组的处理离不开循环，所以本章的内容还包括了第 4 章内容循环的强化。另外，由于数组的元素在内存中是连续存储的，这在本书中第一次给了指针大展身手的空间，完成本章内容的学习，C 语言已经入门一半了。

第 6 章讲述了包括结构体在内的 C 语言各种自定义数据类型的使用方法。本章以概念性内容居多，虽然繁琐但并不难掌握，书中通过各种示例对它们的定义方法和用法一一进行了展示，是经过第 5 章艰苦学习过程后的一次小的休整。

第 7 章讲述了 C 语言中函数的使用及模块化程序设计的基本思想，通过将一个复杂程序划分成若干个函数来实现，从而降低了程序的编写难度。在函数的调用过程中，指针作为函数参数可以起到双向传值的作用，这些都是本章的重点和难点。

第 8 章讲述动态内存的使用方法，操作内存离不开指针，本章首先讲述了如何获取动态内存，然后讲述了通过链表来组织、使用动态内存的方法。链表是结构体、指针的结合，由于操作的复杂性使得使用函数成为必然，所以说本章内容是书上前 7 章内容的综合。完成本章内容的学习，C 语言的掌握可以算入门了。

第 9 章讲述在 C 语言中操作文件的方法，主要是一些文件操作函数的应用。

第 10 章讲述在 C 语言中一些编译预处理命令的使用方法，在本章结尾处介绍了在组织多文件的 C 语言源程序时条件编译的应用，为读者以后编写大型 C 程序提供了方便。

本书在编写过程中参考了许多同行的著作，作者在此一并表达感谢之情。

感谢丹尼斯·里奇(Dennis MacAlistair Ritchie)和肯尼斯·汤姆逊(Kenneth Lane Thompson)，没有他们就没有 C 语言。

感谢杨季文教授和陈建明副教授以及为本书提供直接或间接帮助的每一位朋友，你们的帮助和鼓励促成了本书的顺利完成。如果您能够愉快地读完本书，并告之身边的朋友，原来 C 语言并不难学，那么作者编写本书的目的就达到了。

尽管作者尽了最大努力，但是由于时间关系及作者学识所限，肯定存在缺点和错误，从而影响写作目的，因此，恳请各位读者批评指正，以便再版时修订。

编　者

2010 年 12 月

CONTENTS 目录

第 1 章　C 语言导论 …… 1

1.1　C 语言概述 …… 1

1.1.1　C 语言的功能 …… 1

1.1.2　C 语言的起源 …… 2

1.1.3　C 语言的学习阶段与学习方法 …… 3

1.2　第一个 C 程序 …… 4

1.2.1　程序代码 …… 4

1.2.2　空白和注释 …… 4

1.2.3　预处理指令 …… 5

1.2.4　main 函数 …… 5

1.2.5　程序输出 …… 6

1.3　C 语言程序的运行 …… 6

1.3.1　程序的编译 …… 6

1.3.2　程序的运行和调试 …… 7

1.4　习题 …… 8

1.5　阅读材料——UNIX 和 C 的故事 …… 9

第 2 章　数据 …… 13

2.1　基本数据类型 …… 13

2.1.1　整型数据 …… 13

2.1.2　浮点型数据 …… 16

2.1.3　指针型数据 …… 16

2.2　常量 …… 17

2.2.1　字面常量 …… 17

2.2.2　符号常量 …… 20

2.3　变量 …… 22

2.3.1　标识符 …… 22

2.3.2　变量的定义 …… 23

2.3.3　变量的初始化 …… 24

2.3.4　变量的引用 …… 25

2.4　输出与输入 …… 26

2.4.1　基本输出 …… 26

2.4.2 基本输入 …… 29
2.5 习题 …… 31
2.6 阅读材料——二进制与计算机 …… 33
2.6.1 二进制起源 …… 33
2.6.2 计算机与二进制 …… 34
2.6.3 进制转换 …… 34
2.6.4 计算机中的补码 …… 35

第3章 运算与表达式 …… 38

3.1 算术运算 …… 38
3.1.1 基本算术运算符 …… 38
3.1.2 优先级与结合性 …… 40
3.1.3 数据类型转换 …… 41
3.1.4 自增、自减运算 …… 42
3.2 关系运算 …… 43
3.3 逻辑运算 …… 44
3.4 位运算 …… 45
3.4.1 按位逻辑运算 …… 46
3.4.2 移位运算 …… 48
3.4.3 程序例子 …… 49
3.5 指针运算 …… 50
3.5.1 取地址运算 …… 50
3.5.2 操作指针变量 …… 52
3.5.3 移动指针 …… 53
3.5.4 比较指针 …… 54
3.6 其他运算 …… 55
3.6.1 sizeof 运算 …… 55
3.6.2 逗号运算 …… 56
3.6.3 条件运算 …… 57
3.7 赋值运算 …… 58
3.7.1 赋值运算符和赋值表达式 …… 58
3.7.2 复合赋值运算 …… 58
3.8 习题 …… 59

第4章 程序控制结构 …… 62

4.1 程序语句 …… 62
4.2 顺序结构 …… 63
4.3 选择结构 …… 65
4.3.1 if 语句 …… 65

此为试读，需要完整PDF请访问：www.ertongbook.com

4.3.2 if 语句的嵌套 …… 68
4.3.3 switch 语句 …… 74
4.4 循环结构 …… 77
4.4.1 while 循环 …… 78
4.4.2 do while 循环 …… 80
4.4.3 for 循环 …… 82
4.4.4 其他控制语句 …… 84
4.4.5 循环的嵌套 …… 86
4.5 习题 …… 89

第 5 章 数组 …… 91

5.1 一维数组 …… 91
5.1.1 一维数组的定义 …… 91
5.1.2 一维数组元素的引用 …… 93
5.1.3 一维数组的初始化 …… 95
5.1.4 程序举例 …… 96
5.2 多维数组 …… 97
5.2.1 多维数组的定义 …… 98
5.2.2 多维数组元素的引用 …… 100
5.2.3 多维数组的初始化 …… 103
5.2.4 程序举例 …… 104
5.3 字符数组与字符串 …… 106
5.3.1 字符数组与字符串的关系 …… 106
5.3.2 字符串的输入输出 …… 107
5.3.3 字符串处理函数 …… 109
5.3.4 程序举例 …… 112
5.4 指针数组 …… 115
5.4.1 指针数组的定义与应用 …… 115
5.4.2 指向指针的指针 …… 119
5.5 习题 …… 120

第 6 章 其他数据类型 …… 122

6.1 结构体类型的定义 …… 122
6.2 结构体类型变量 …… 123
6.2.1 结构体变量的定义 …… 123
6.2.2 结构体变量的引用 …… 125
6.2.3 结构体变量的初始化 …… 128
6.3 结构体类型数组 …… 129
6.3.1 结构体数组的定义 …… 130

6.3.2 结构体数组的初始化 …… 130
6.3.3 结构体数组的引用 …… 132
6.4 位段类型 …… 133
6.4.1 位段成员的定义 …… 134
6.4.2 位段成员的使用 …… 135
6.5 共用体类型 …… 137
6.5.1 共用体类型的定义 …… 137
6.5.2 共用体变量的定义 …… 139
6.5.3 共用体变量的引用 …… 140
6.5.4 共用体变量的初始化 …… 142
6.5.5 共用体变量的应用 …… 143
6.6 枚举类型 …… 145
6.6.1 枚举类型的定义 …… 145
6.6.2 枚举类型变量的定义与引用 …… 146
6.7 typedef 自定义类型 …… 147
6.7.1 typedef 定义类型 …… 148
6.7.2 typedef 应用举例 …… 149
6.8 习题 …… 151

第 7 章 函数 …… 152

7.1 函数的定义和调用 …… 152
7.1.1 函数概述 …… 152
7.1.2 函数的定义 …… 153
7.1.3 函数的调用 …… 153
7.2 函数的返回值、参数及函数声明 …… 154
7.2.1 函数的返回值 …… 154
7.2.2 函数的参数 …… 157
7.2.3 函数的声明 …… 159
7.3 函数的嵌套和递归调用 …… 160
7.3.1 函数的嵌套调用 …… 160
7.3.2 函数的递归调用 …… 163
7.4 函数与指针 …… 167
7.4.1 指针变量作为函数参数 …… 167
7.4.2 数组与函数 …… 169
7.4.3 返回指针值的函数 …… 171
7.4.4 指向函数的指针 …… 172
7.5 作用域 …… 175
7.5.1 局部作用域 …… 175
7.5.2 全局作用域 …… 177

7.5.3 多文件下的全局作用域 …… 179
7.6 变量的存储类别 …… 183
7.6.1 动态存储方式 …… 183
7.6.2 静态存储方式 …… 185
7.7 习题 …… 186

第 8 章 内存的使用 …… 188

8.1 动态使用内存 …… 188
8.1.1 分配内存 …… 188
8.1.2 释放内存 …… 189
8.1.3 应用举例 …… 190
8.2 链表 …… 192
8.2.1 链表概述 …… 192
8.2.2 创建链表 …… 196
8.2.3 释放链表 …… 204
8.2.4 链表操作 …… 207
8.3 习题 …… 209

第 9 章 文件 …… 210

9.1 文件概述 …… 210
9.1.1 数据文件 …… 210
9.1.2 文件的读写 …… 210
9.1.3 文件指针 …… 211
9.1.4 文件操作的步骤 …… 211
9.2 文件的打开与关闭 …… 211
9.2.1 打开文件 …… 211
9.2.2 关闭文件 …… 213
9.3 文件的读写 …… 214
9.3.1 字符读写文件 …… 214
9.3.2 字符串读写 …… 216
9.3.3 格式化读写文件 …… 218
9.3.4 非格式化读写文件 …… 219
9.4 文件的随机读写 …… 221
9.4.1 文件定位 …… 221
9.4.2 应用举例 …… 222
9.5 习题 …… 223

第 10 章 编译预处理 …… 225

10.1 宏定义 …… 225

10.1.1 无参宏定义 …… 225
10.1.2 带参宏定义 …… 227
10.2 文件包含 …… 230
10.3 条件编译 …… 231
10.3.1 条件编译的形式 …… 231
10.3.2 条件编译与多文件组织 …… 233
10.4 习题 …… 234

附录A 常用字符与ASCII码对照表 …… 236

附录B 运算符和结合性 …… 237

附录C 常用标准库函数 …… 239

参考文献 …… 248

第1章

C语言导论

1.1 C语言概述

1.1.1 C语言的功能

目前，计算机的应用已经深入到社会的各个领域，成为人们日常工作、生活、学习的必备工具。计算机是一种具有存储程序、执行程序能力的电子设备，计算机所有能力都是通过执行程序来实现的。程序就是人们把需要做的工作写成一定形式的指令序列，并把它存储在计算机内部存储器中，当人们给出命令之后，计算机就按照指令的执行顺序自动进行相应操作，从而完成相应的工作。人们把这种可以连续执行的一条条指令的序列称为“程序”。编写程序的过程就称为“程序设计”。人们编写的指令序列为了使计算机能够正确识别和执行，不能随意编写，必须有一定的规则。这些规则的定义包含了一系列的文法和语法的要求，按照这些规则编写的程序能够被计算机理解执行，所以它是人和计算机之间的交流语言。这种语言类似于人与人之间交流的语言，虽然没有人类语言那么复杂，但逻辑上要求更加严格，符合这些规则的“语言”也被称为“程序设计语言”。

机器语言或称为二进制代码语言，计算机可以直接识别。每台机器的指令，其格式和代码所代表的含义都是硬性规定的，如某种计算机的指令为 1011011000000000，它表示让计算机进行一次加法操作；而指令 1011010100000000 则表示进行一次减法操作。它们的前 8 位表示操作码，而后 8 位表示地址码。因为硬件设计不同，机器语言对不同型号的计算机来说一般是不同的。用机器语言编程，就是从实用的 CPU 的指令系统中挑选合适的指令，组成一个指令系列的过程。

由于“机器语言”与人们日常生活中使用的语言差距过大，而且大量的规则都和具体的计算机硬件设计和实现相关，所以使用“机器语言”编写程序难度很大。为了降低编写程序的难度，人们发明了一些更加接近人类日常语言的程序设计语言，但这些语言编写的程序不能被计算机直接识别、执行，必须翻译成“机器语言程序”才能被计算机执行。

根据程序设计语言与人类语言的接近程度，基本上可把这些程序设计语言分为高级语言、中级语言、低级语言。低级语言最接近机器语言，学习和使用难度都比较大；高级语言最接近人类语言，学习和使用难度相对于低级语言要容易得多，应用最为广泛。目前常见的高级语言有 C、Java、C++、C#、Basic、Pascal 等。由程序设计语言编写的程序称为“源程

序”，高级语言编写的程序不能被计算机硬件直接识别、执行，高级语言源程序编译(Compile)成机器语言程序的过程如图 1.1 所示。

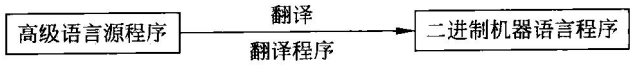


图 1.1　程序编译、连接过程示意图

几十年来，人们发明了很多种计算机程序设计语言，目前还在不断有新的程序设计语言被发明出来，这些语言往往具有不同的特点。C 语言是目前世界上应用最为广泛的高级程序设计语言，它是一种通用的高级程序设计语言，可以用来完成各种类型的应用软件设计。C 语言的通用性和无限制性使得它对于程序设计者来说都显得更加方便、更加有效。从微型计算机(包括我们日常使用的 PC)到小型机、中型机、大型机、巨型机，都离不开 C 语言编写的程序；从家用冰箱、电视、洗衣机、空调到手机，它们能够有效的工作大部分都依赖于内部运行的 C 语言编写的程序；现代化的智能车床、工业控制设备、汽车、火箭、宇宙飞船，其内部运行程序大部分也都是用 C 语言编写的。可以说，有计算机的地方就有 C 语言编写的程序在运行。

1.1.2　C 语言的起源

在学习 C 语言之前，我们有必要了解一下 C 语言的发展历史。

目前，计算机技术发展速度之快让人目不暇接，作为计算机软件技术的基础，新的程序设计语言和新的操作系统也在不断涌现，这些新技术、新产品往往都是由国际著名企业或国家重要部门投入巨大的人力、物力所开发出来的。然而，提到目前最优秀、最有价值的程序设计语言和操作系统，却要到四十年前去找，那就是强大的 C 语言和用 C 语言编写的 UNIX 操作系统，更加让人惊奇的是它们竟然只是基于个人兴趣由几个人所缔造的，花费的人力物力代价更是少得可怜。

从历史发展的角度看，C 语言起源于 1968 年发表的 CPL 语言(Combined Programming Language)，它的许多重要思想来自于 Martin Richards 在 1969 年研制的 BCPL 语言，以及以 BCPL 语言为基础的 B 语言。Dennis M. Ritchie 在 B 语言的基础上，于 1972 年研制了 C 语言，并用 C 语言写成了第一个在 PDP-11 计算机上实现的 UNIX 操作系统(主要在贝尔实验室内部使用)。以后，C 语言又经过多次改进，直到 1975 年用 C 语言编写的 UNIX 操作系统第 6 版公诸于世后，C 语言才举世瞩目。1977 年出现了独立于机器的 C 语言编译文本《可移植 C 语言编译程序》，从而大大简化了把 C 语言编译程序移植到新环境所需做的工作，这本身也使得 UNIX 操作系统迅速地在众多的机器上普及。随着 UNIX 的日益广泛使用，C 语言也迅速得到推广。1978 年以后，C 语言先后移植到大、中、小、微型计算机上，它的应用领域已不再限于系统软件的开发，而成为当今最流行的程序设计语言。

以 1978 年发布的 UNIX 第 7 版的 C 语言编译程序为基础，Brian W. Kernighan 和 Dennis M. Ritchie 合著了影响深远的名著 *The C Programming Language*，这本书中介绍的 C 语言成为后来广泛使用的 C 语言版本的基础，它被称为标准 C。

1983 年美国国家标准化协会(ANSI)根据 C 语言问世以来的各种版本，对 C 语言的发展和扩充制定了新的标准，称为 ANSI C。1989 年 ISO 根据 ANSI C 公布了 C 标准，称

此为试读，需要完整PDF请访问：www.ertongbook.com