

C语言 程序设计教程

Cyuyan Chengxu Sheji Jiaocheng

主 编 李德龙 副主编 户军茹



国防工业出版社

National Defense Industry Press

C 语言程序设计教程

主 编 李德龙

副主编 户军茹

参 编 丁 谊 张 震

国防工业出版社

· 北京 ·

内容提要

本书是为C语言程序设计课程编写的教材，也可以作为学习C语言程序设计的参考教材。

编程实践是学习程序设计的重要环节。多读经典程序，多分析程序，多编写程序，多上机实践，是掌握程序设计思想的关键。本书提供了大量典型的例题分析和用于自测的思考题与习题，便于读者巩固提高。

本书第一部分讨论C语言程序设计的基础知识，包括变量、运算符、输入输出及程序流程控制等；第二部分讨论C语言程序设计的提高知识，包括一维数组、函数及指针等；第三部分讨论C语言程序设计的高级知识，包括二维数组、结构体与链表、函数的递归调用及文件等。

本书概念表述严谨，逻辑推理严密，语言精练，既便于教学又便于自学。

本书可作为计算机类专业或信息类专业本科或专科教材，也可作为从事计算机工程与应用工作的科技工作者的参考书。

图书在版编目(CIP)数据

C语言程序设计教程/李德龙主编.—北京：国防工业出版社，2015.8

ISBN 978-7-118-10452-3

I. ①C… II. ①李… III. ①C语言—程序设计—教材
IV. ①TP312

中国版本图书馆CIP数据核字(2015)第204389号

※

国防工业出版社 出版发行

(北京市海淀区紫竹院南路23号 邮政编码100048)

国防工业出版社印刷厂印刷

新华书店经售

*

开本 787×1092 1/16 印张 11¼ 字数 283千字

2015年8月第1版第1次印刷 印数1—4000册 定价28.00元

(本书如有印装错误，我社负责调换)

国防书店：(010)88540777

发行邮购：(010)88540776

发行传真：(010)88540755

发行业务：(010)88540717

前 言

C语言兼具高级语言和汇编语言的特点，广受编程人员的喜爱。作为一种被广泛使用的编程语言，它已经在事实上成为我国各类院校进行计算机编程教育时主要使用的教学语言。但作为教学语言，它过于繁琐的语法细节和灵活的使用方法，往往使得初学编程的学生无法把注意力放在“如何把实际问题转化为程序”，而是放在“怎样让程序能够顺利的运行”，导致程序设计的入门课程变成了纯粹的C语言教学课程，造成了本末倒置的结果。为了让程序设计的入门课程回归本源，编者根据自己的实际学习和多年教学经历编写了本书，本书例题、习题既有经典问题，又增加了根据程序设计教学实践特点设计的问题。

本书的特点主要体现在以下几方面。

应用为王，快速入门。编者认为程序设计只有在编写程序的过程中才能够真正掌握。在本书的设计中，经过三次课的学习，读者就可以掌握基本编程所需要的语法，从而快速进入实践环节。

由浅入深，层层递进。本书打破了以语法为章节设计点的常规模式，根据编程的实际需要，设置了三个层次的学习内容，由浅入深地进行学习，不同层次或不同需要的读者都可以按照进度学习到自己所需要掌握的知识。

轻语法，重实践。本书的语法讲解都围绕着编程实践展开，重点在于引导读者学会如何把实际问题转化为计算机程序，着重培养读者的实际动手能力。

实战练习，步步提高。实践是学习编程的最好方法，本书通过三种方法来提高读者的实践能力：第一，本书的所有代码都在实际的环境中调试通过，提供给读者一个可以模仿的范本，便于读者的初步学习；第二，本书的实例问题在不同章节会反复出现，但是每次出现都会提高难度，帮助读者逐步掌握复杂程序的设计方法；第三，通过设置实战章节，集中讲解与实际问题的算法知识，帮助读者掌握解决实际问题的方法。

贴心提醒，处处引导。本书贯彻引导学生自主学习的思路，将各种编程时要注意的问题、语法陷阱及需要深入思考的问题等设置为不同的小栏目，让读者在学习过程中自然轻松地了解相关知识。

本书是由多年从事软件设计与C语言教学的教师编写的，主要编写人员有李德龙、户军茹、丁谊、张震。在成书的过程中，得到了张敏情、郭敦陶、武光明等的帮助。我们始终本着科学、严谨的态度，力求精益求精，但书中错误、疏漏之处在所难免，敬请广大读者批评指正。

祝读者顺利进入计算机程序设计的领域！

编者

2015年6月

第一篇 基础篇

第 1 章 初识 C 语言	2
1.1 C 语言的诞生	2
1.1.1 机器语言	2
1.1.2 汇编语言	2
1.1.3 高级语言	3
1.1.4 C 语言	3
1.1.5 面向对象程序语言	4
1.1.6 下一代程序设计语言	4
1.2 C 语言的特点	5
1.2.1 优点	5
1.2.2 缺点	5
1.3 C 语言开发环境的搭建	6
1.3.1 编译软件 Visual C++ 6.0 的安装	6
1.3.2 Visual C++ 6.0 的使用	7
1.4 简单的 C 语言程序	7
1.5 C 语言程序的基本结构	8
1.5.1 基本程序结构	8
1.5.2 C 语言的关键字	10
1.6 算法及流程图	10
习题	12
第 2 章 变量、运算符及输入输出	13
2.1 数据类型	13
2.2 常量与变量	14
2.2.1 常量	14
2.2.2 变量	15
2.2.3 变量命名规则	15
2.3 整型数据（整数）	16
2.3.1 整型数据	16

2.3.2 整型变量	16
2.4 实型数据 (实数)	16
2.4.1 实型数据	16
2.4.2 实型变量	16
2.5 字符型数据	17
2.5.1 字符常量	17
2.5.2 转义字符	18
2.5.3 字符变量	18
2.6 算术运算符	19
2.6.1 算术运算符	19
2.6.2 算术运算符的优先级	20
2.6.3 强制类型转换	20
2.7 输入输出函数的使用及格式输出	21
2.7.1 scanf()函数及输入格式控制	21
2.7.2 printf()函数及输出格式控制	24
2.8 常用的数学函数	26
习题	27
第 3 章 程序流程控制	28
3.1 循环控制语句(for 语句)	28
3.2 关系和逻辑运算符	31
3.3 条件控制语句(if 语句)	32
3.3.1 if 语句	33
3.3.2 if...else if 语句	36
习题	38
第 4 章 调试程序	40
习题	43
第 5 章 实战练习—穷举法	44
5.1 穷举法—计算类问题	44
5.2 穷举法—排列组合类问题	47
5.3 穷举法—图形类问题	49
5.4 穷举法—逻辑推理类问题	53

第二篇 提高篇

第 6 章 选择结构和循环结构的其他形式	58
6.1 用 switch 语句实现多分支选择结构	58
6.2 循环的其他形式和循环控制语句	60
6.2.1 while 语句	60
6.2.2 do...while 语句	63

6.3 break 和 continue 语句.....	64
6.3.1 break 语句.....	65
6.3.2 continue 语句.....	65
习题.....	66
第 7 章 一维数组	67
7.1 一维整型数组.....	67
7.1.1 一维数组的定义.....	67
7.1.2 一维数组元素的引用.....	68
7.1.3 一维数组的初始化.....	69
7.1.4 程序举例.....	70
7.1.5 查找算法.....	71
7.1.6 插入算法.....	72
7.1.7 删除算法.....	72
7.1.8 排序算法(选择法与冒泡法).....	73
7.2 一维字符型数组.....	76
7.2.1 字符数组的定义.....	76
7.2.2 数组的初始化.....	76
7.2.3 数组的引用.....	76
7.2.4 字符串形式的字符数组.....	77
7.2.5 字符串的输入输出.....	77
7.2.6 字符串处理函数.....	78
7.2.7 字符数组在加解密算法中的应用.....	80
习题.....	83
第 8 章 函数	84
8.1 函数的概念及定义.....	84
8.1.1 函数的基本概念.....	84
8.1.2 函数定义的一般形式.....	86
8.2 函数的调用.....	86
8.2.1 函数的参数.....	86
8.2.2 函数的返回值.....	87
8.2.3 函数调用.....	88
8.2.4 函数声明.....	89
8.3 局部变量与全局变量.....	90
8.3.1 局部变量.....	90
8.3.2 全局变量.....	91
8.4 不同参数类型的程序举例.....	92
8.4.1 基本类型数据作函数的参数.....	92
8.4.2 数组元素作函数的参数.....	93
8.4.3 数组名作函数的参数.....	94

8.5 文件包含	98
8.6 C++中的函数重载	100
习题	101
第9章 指针	103
9.1 指针变量	103
9.1.1 指针变量的定义与初始化	103
9.1.2 指针赋值	104
9.1.3 指针变量的引用	105
9.2 指针与数组	106
9.2.1 指向数组元素的指针	106
9.2.2 通过指针引用数组元素	107
9.3 指向字符串的指针变量	109
9.3.1 字符串的表示形式	109
9.3.2 使用字符串指针变量与字符数组的区别	111
9.4 指针与函数	112
9.4.1 指向基本类型的指针变量作函数参数	112
9.4.2 指向数组的指针变量作函数参数	113
9.4.3 字符串指针作函数的参数	115
习题	116

第三篇 高级篇

第10章 二维数组定义与应用	118
10.1 二维数组的定义	118
10.2 二维数组元素的引用	119
10.3 二维数组的初始化	120
10.4 二维数组程序举例	121
习题	123
第11章 结构体与链表	125
11.1 为什么需要结构体	125
11.2 结构体的创建	126
11.3 结构体类型变量的说明	127
11.4 结构体变量成员的表示方法	129
11.5 结构体变量的赋值	129
11.6 结构体变量的初始化	129
11.7 结构体数组的定义	130
11.8 结构体指针变量的说明和使用	132
11.8.1 指向结构体变量的指针	132
11.8.2 指向结构体数组的指针	134

11.8.3 结构体指针变量作函数参数.....	135
11.9 动态存储分配.....	136
11.10 链表的概念.....	137
习题.....	140
第 12 章 函数的递归调用.....	141
第 13 章 文件.....	143
13.1 C 文件概述.....	143
13.2 文件指针.....	144
13.3 文件的打开与关闭.....	144
13.3.1 文件的打开(fopen 函数).....	144
13.3.2 文件关闭函数 (fclose 函数)	146
13.4 文件的读写	146
13.4.1 字符读写函数 fgetc 和 fputc.....	146
13.4.2 字符串读写函数 fgets 和 fputs	149
13.4.3 数据块读写函数 fread 和 fwrite.....	151
13.4.4 格式化读写函数 fscanf 和 fprintf.....	152
13.5 文件的随机读写	153
13.5.1 文件定位	153
13.5.2 文件的随机读写	154
13.6 文件检测函数.....	154
13.6.1 文件结束检测函数 feof 函数	154
13.6.2 读写文件出错检测函数.....	155
13.6.3 文件出错标志和文件结束标志置 0 函数.....	155
习题.....	155
附录 1 ASCII 代码对照表.....	156
附录 2 运算符和结合性.....	157
附录 3 C 库函数.....	159
附录 4 全国计算机等级考试二级 C 语言笔试真题.....	166
参考答案.....	176
参考文献.....	177

第一篇 基础篇

680-221964

自 1946 年世界上第一台电子计算机问世以来, 计算机科学及其应用的发展十分迅猛, 计算机被广泛地应用于人类生产、生活的各个领域, 推动了社会的进步与发展。计算机已将人类带入了一个新的时代——信息时代。新的时代对于我们的基本要求之一便是自觉、主动地学习和掌握计算机的基本知识和基本技能, 并把它作为自己应该具备的基本素质。要充分认识到, 缺乏计算机知识, 就是信息时代的“文盲”。对于理工科的大学生而言, 掌握一门高级程序设计语言及其基本的编程技能是必需的。

计算机是由硬件系统和软件系统两大部分构成的。硬件是物质基础, 而软件可以说是计算机的灵魂, 没有软件, 计算机就是一台“裸机”, 什么也不能干; 有了软件, 才能灵动起来, 成为一台真正的“电脑”。所有的软件, 都是用计算机语言编写的。

1.1 C 语言的诞生

计算机程序设计语言的发展, 经历了从机器语言、汇编语言到高级语言的历程。

1.1.1 机器语言

电子计算机所使用的是由“0”和“1”组成的二进制数, 二进制是计算机语言的基础。计算机发明之初, 人们只能降贵纡尊, 用计算机的语言去命令计算机干这干那。一句话, 就是写出一串串由“0”和“1”组成的指令序列交由计算机执行, 这种语言就是机器语言。

使用机器语言是十分痛苦的, 特别是在程序有错需要修改时, 更是如此。而且, 由于每台计算机的指令系统往往各不相同, 所以在一台计算机上执行的程序, 要想在另一台计算机上执行, 必须另编程序, 造成了重复工作。但由于使用的是针对特定型号计算机的语言, 故而运算效率是所有语言中最高的。机器语言, 是第一代计算机语言。

1.1.2 汇编语言

为了减轻使用机器语言编程的痛苦, 人们进行了一种有益的改进: 用一些简洁的英文字母、符号串替代一个特定指令的二进制串。例如, 用“ADD”代表加法, “MOV”代表数据传递等。这样一来, 人们很容易读懂并理解程序在干什么, 纠错及维护都变得方便了, 这种程序设计语言称为汇编语言, 即第二代计算机语言。然而计算机是不认识这些符号的, 这就需要一个

专门的程序，负责将这些符号翻译成二进制数的机器语言，这种翻译程序被称为汇编程序。

汇编语言同样十分依赖于机器硬件，移植性不好，但效率仍十分高。针对计算机特定硬件而编制的汇编语言程序，能准确发挥计算机硬件的功能和特长，程序精炼而质量高，所以至今仍是一种常用而强有力的软件开发工具。

1.1.3 高级语言

从最初与计算机交流的痛苦经历中，人们意识到，应该设计一种这样的语言，这种语言接近于数学语言或人的自然语言，同时又不依赖于计算机硬件，编出的程序能在所有机器上通用。经过努力，1954年，第一个完全脱离机器硬件的高级语言——FORTRAN问世了，60多年来，共有几百种高级语言出现，有重要意义的有几十种，影响较大、使用较普遍的有FORTRAN、ALGOL、COBOL、BASIC、LISP、NOBOL、PL/1、Pascal、C、PROLOG、Ada、C++、VC、VB、Delphi、JAVA等。

高级语言的发展也经历了从早期语言到结构化程序设计语言，从面向过程到非过程化程序语言的过程。相应地，软件的开发也由最初的个体手工作坊式的封闭式生产，发展为产业化、流水线式的工业化生产。60年代中后期，软件越来越多，规模越来越大，而软件的生产基本上是个自为战，缺乏科学规范的系统规划与测试、评估标准，其恶果是大批耗费巨资建立起来的软件系统，由于含有错误而无法使用，甚至带来巨大损失，软件给人的感觉是越来越不可靠，以致几乎没有不出错的软件。这一切，极大地震动了计算机界，史称“软件危机”。人们认识到：大型程序的编制不同于写小程序，它应该是一项新的技术，应该像处理工程一样处理软件研制的全过程。程序的设计应易于保证正确性，也便于验证正确性。1969年，提出了结构化程序设计方法，1970年，第一个结构化程序设计语言——Pascal语言出现，标志着结构化程序设计时期的开始。

1.1.4 C语言

C语言之所以命名为C，是因为C语言源自Ken Thompson（肯·汤普逊）发明的B语言，而B语言则源自BCPL语言。

1967年，剑桥大学的Martin Richards对CPL语言进行了简化，于是产生了BCPL（Basic Combined Programming Language）语言。

1970年，美国贝尔实验室的Ken Thompson以BCPL语言为基础，设计出很简单且很接近硬件的B语言（取BCPL的首字母）。并且他用B语言写了第一个UNIX操作系统。

1972年，美国贝尔实验室的D.M.Ritchie（丹尼斯·里奇）在B语言的基础上最终设计出了一种新的语言，他取了BCPL的第二个字母作为这种语言的名字，这就是C语言。他与Ken Thompson同为1983年图灵奖得主。

1977年，Dennis M.Ritchie发表了不依赖于具体机器系统的C语言编译文本《可移植的C语言编译程序》。



C语言创始人D.M. Ritchie

● K&R C

1978年由美国电话电报公司(AT&T)贝尔实验室正式发表了C语言。Brian Kernighan和

Dennis Ritchie 出版了一本书，名叫《The C Programming Language》。这本书被 C 语言开发者们称为“K&R”，很多年来被当作 C 语言的非正式的标准说明。人们称这个版本的 C 语言为“K&R C”。

- ANSI C

1970 到 80 年代，C 语言被广泛应用，从大型主机到小型微机，也衍生了 C 语言的很多不同版本。

1983 年美国国家标准局（American National Standards Institute，简称 ANSI）成立了一个委员会，来制定 C 语言标准。

1989 年 C 语言标准被批准，被称为 ANSI X3.159-1989 “Programming Language C”。这个版本的 C 语言标准通常被称为 ANSI C。

- C99

1990 年，国际标准化组织 ISO (International Organization for Standards) 接受了 89 ANSI C 为 ISOC 的标准 (ISO 9899—1990)。1994 年，ISO 修订了 C 语言的标准。

1995 年，ISO 对 C90 做了一些修订，即“1995 基准增补 1 (ISO/IEC 9899/AMD1:1995)”。

1999 年，ISO 又对 C 语言标准进行修订，在基本保留原来 C 语言特征的基础上，针对应该的需要，增加了一些功能，命名为 ISO/IEC 9899:1999。

在 ANSI 标准化后，C 语言的标准在一段相当的时间内都保持不变，尽管 C 语言继续在改进。（实际上，Normative Amendment 1 在 1995 年已经开发了一个新的 C 语言版本。但是这个版本很少为人所知。）它被 ANSI 于 2000 年 3 月采用。

- C11

2001 年和 2004 年先后进行了两次技术修正。

2011 年 12 月 8 日，ISO 正式公布 C 语言新的国际标准草案：ISO/IEC 9899:2011，即 C11。

1.1.5 面向对象程序语言

80 年代初开始，在软件设计思想上，又产生了一次革命，其成果就是面向对象的程序设计。在此之前的高级语言，几乎都是面向过程的，程序的执行是流水线似的，在一个模块被执行完成前，人们不能干别的事，也无法动态地改变程序的执行方向。这和人们日常处理事物的方式是不一致的，对人而言是希望发生一件事就处理一件事，也就是说，不能面向过程，而应是面向具体的应用功能，也就是对象（object）。其方法就是软件的集成化，如同硬件的集成电路一样，生产一些通用的、封装紧密的功能模块，称之为软件集成块，它与具体应用无关，但能相互组合，完成具体的应用功能，同时又能重复使用。对使用者来说，只关心它的接口（输入量、输出量）及能实现的功能，至于如何实现的，那是它内部的事，使用者完全不用关心，C++、VB、Delphi 就是典型代表。

1.1.6 下一代程序设计语言

高级语言的下一个发展目标是面向应用，也就是说：只需要告诉程序你要干什么，程序就能自动生成算法，自动进行处理，这就是非过程化的程序语言。例如数据库查询语言 SQL (Structured Query Language)。

1.2 C语言的特点

C语言是一种计算机程序设计语言，它既具有高级语言的特点，又具有汇编语言的特点。1978年后，C语言已先后被移植到大、中、小及微型机上，它可以作为工作系统设计语言，编写系统应用程序，也可以作为应用程序设计语言，编写不依赖计算机硬件的应用程序。它的应用范围广泛，具备很强的数据处理能力，不仅仅是在软件开发上，而且各类科研都需要用到C语言，同时结合各种C语言支持的函数库，可以用于编写系统软件，三维、二维图形和动画，具体应用比如单片机以及嵌入式系统开发。

1.2.1 优点

(1) 语言简洁、紧凑、使用灵活、方便。C语言共有32个关键字，9种控制语句，程序书写形式自由，主要用小写字母表示。

C语言	PASCAL	语言含义
{...}	BEGIN ...END	复合语句
if (e) S;	IF (e) THEN S	条件语句
int i;	VAR i : INTEGER	定义i为整型变量
int f ();	FUNCTION f (): INTEGER	定义f为整型的函数
int *p; VAR	P: INTEGER	定义P为指针变量
i++, ++i	i=i+1	i自增值1, i=i+1

(2) 运算符丰富。C语言共有34种运算符，运算类型丰富，表达式类型多样，使用灵活方便。

(3) 数据结构丰富。C语言具有现代化语言的各种数据结构，如：整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等，能实现各种复杂的数据结构的运算。

(4) 具有结构化的控制语句。C语言用函数作为程序模块以实现程序的模块化，是结构化的理想语言，符合现代编程风格要求。如：

if...else 语句	while 语句	do...while 语句
switch 语句	for 语句	break 语句

(5) 程序设计自由度大。一般的高级语言语法检查比较严，而C语言允许程序员有较大的自由度，放宽了语法检查。C语言程序员要求对程序设计更为熟练。

(6) 可直接对硬件进行操作。C语言允许直接访问物理地址，进行位操作，能实现汇编语言的大部分功能。

(7) 程序执行效率高。C语言描述问题比汇编语言迅速，工作量小、可读性好，易于调试、修改和移植，而代码质量与汇编语言相当。C语言一般只比汇编程序生成的目标代码效率低10%~20%。

(8) 可移植性好。C语言在不同机器上的C编译程序，86%的代码是公共的，所以C语言的编译程序便于移植。在一个环境上用C语言编写的程序，不改动或稍加改动，就可移植到另一个完全不同的环境中运行。

1.2.2 缺点

(1) C语言的缺点主要表现在数据的封装性上，这一点使得C语言在数据的安全性上有很大的缺陷，这也是C语言和C++的一大区别。

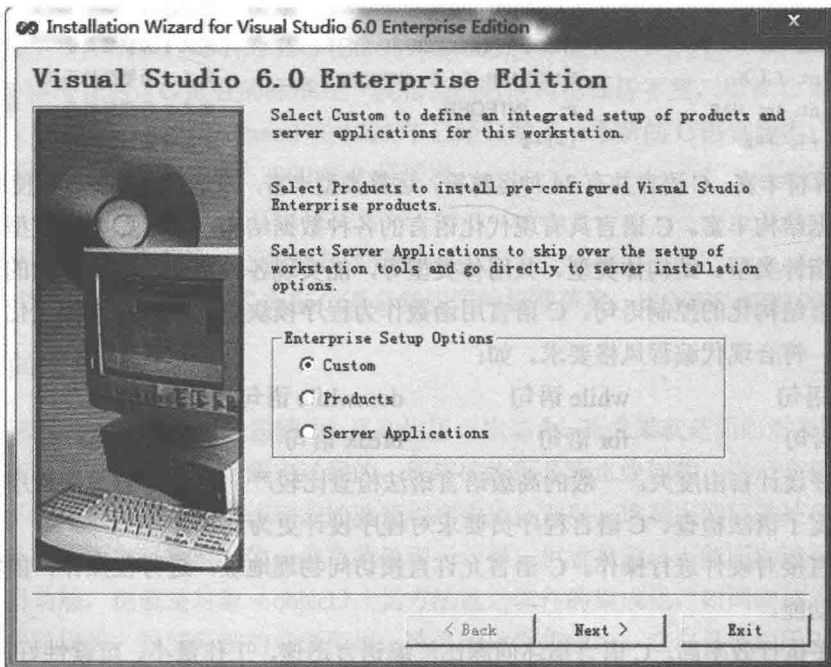
(2) C 语言的语法限制不太严格，对变量的类型约束不严格，影响程序的安全性，对数组下标越界不作检查等。从应用的角度，C 语言比其他高级语言较难掌握。也就是说，对用 C 语言的人，要求对程序设计更熟练一些。

1.3 C 语言开发环境的搭建

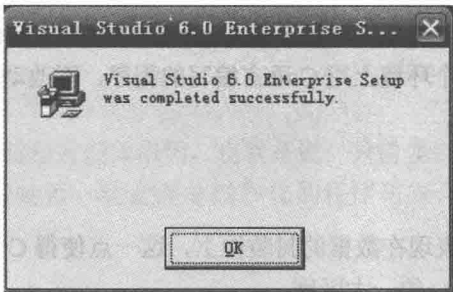
常用的编译软件有 Microsoft Visual C++，Borland C++，gcc(Linux 系统下最常用的编译器)，Watcom C++，Borland C++ Builder，Borland C++ 3.1 for DOS，Watcom C++ 11.0 for DOS，GNU DJGPP C++，Lccwin32 C Compiler 3.1，Microsoft C，High C 等。我们上课使用 Borland C++ 的编译器，为了使用方便，采用编译软件 Visual C++ 6.0 作为开发环境。它比 DOS 环境下的 Turbo C 环境使用更加方便。

1.3.1 编译软件 Visual C++ 6.0 的安装

(1) 安装 Visual C++ 6.0。双击安装文件，在下面安装界面上单击“Next”。



(2) 一直选择下一步，直到安装结束，单击“OK”按钮，自动运行 Visual Studio 6.0 程序。



1.3.2 Visual C++ 6.0 的使用

(1) 文件的创建与打开。如果是创建新的程序,选择“文件”→“新建”命令,系统默认新创建的文件后缀名为“.cpp”。如果要打开已有的程序文件,选择“文件”→“打开”命令。

(2) 文件的编辑与保存。创建好新的程序文件或打开已有的程序文件后,可以在编辑区内对该文件进行编辑,编辑好以后选择“文件”→“保存”命令,保存编辑好的程序文件。如果想修改文件的扩展名,比如要保存为“.c”文件,可以选择“文件”→“另存为”命令,保存时将扩展名选为“.c”就可以了。

(3) 文件的调试与运行。该部分内容在讲授具体程序时讲解。

1.4 简单的C语言程序

C语言是国际上广泛流行的计算机高级语言,它是一种用途广泛、功能强大、使用灵活的过程性编程语言,既可用于编写应用程序,又能用于编写系统软件。同时C语言又接近自然语言,更易理解和使用,下面先通过几个简单的C程序了解一下C语言的编程过程。

[例 1-1] 求 1~100 的和。

```
#include <stdio.h>
int main()
{
    int i,sum=0;
    for(i=1;i<=100;i=i+1)
        sum=sum+i;
    printf("%d",sum);
}
```

从以上程序可以看出,C语言很接近自然语言,我们在学习C语言的语法之前,就可以基本上理解C程序的功能,并能在此功能上进行改进。例如,可以将上面的程序改为求1到100的奇数的和。

[例 1-2] 求 1~100 奇数的和。

```
#include <stdio.h>
int main()
{
    int i,sum=0;
    for(i=1;i<=100;i=i+2)
        sum=sum+i;
    printf("%d",sum);
}
```



思考题

求 1 到 1000 偶数的和。

**提 示****■ 如何执行示例程序？**

第一步：打开 Visual C++ 6.0 程序，菜单“文件→新建”，新建一个 CPP 文件。

第二步：创建好 C 程序文件后，在编辑区内对该程序文件进行编辑。

第三步：编辑好程序文件后，按 F5 键运行程序，查看运行结果。

■ 示例程序无法执行怎么办？

示例程序无法执行的原因都是由于输入错误造成的，主要有以下几种情况：

1. 输入程序时没有区分大小写字母；
2. 输入程序时没有区分半角和全角（字母、数字、空格）；
3. 程序的语句后面没有加“；”；
4. 变量类型与变量之间没有空格。
5. 除了双引号之间，其他地方不能出现汉字和全角字符。

1.5 C 语言程序的基本结构

1.5.1 基本程序结构

任何一种程序设计语言都具有特定的语法规则和规定的表达方法。一个程序只有严格按照语言规定的语法和表达方式编写，才能保证编写的程序在计算机中能正确地执行，同时也便于阅读和理解。

为了了解 C 语言的基本程序结构，我们先介绍几个简单的 C 程序。

[例 1-3]

```
#include <stdio.h>           //标准函数库
int main( )                  //主函数
{
    printf("This is a sample of c program. "); /*调用标准库函数，
                                                显示引号中的内容*/
}
```

这是一个最简单的 C 程序，其执行结果是在屏幕上显示一行信息：

This is a sample of c program.

**提 示****■ 在 C 语言程序中如何添加注释？**

注释：表示对程序的说明（称为注释），不参与程序的运行。注释文字可以是任意字符如汉字、拼音、英文等。

//是 C99 标准及 C++ 中的注释风格，用于单行注释，从//符号开始到行结束的为注释部分。

/* */ 是 C 旧标准中的注释风格，在新的 C99 标准及 C++ 中也可以使用，一般用于多行注释。