十一五

C/C++程序设计教程

普通高等学校『十一五』规划教材

梁成升 编著

國防工業出版
National Defense Industry P

普通高等学校“十一五”规划教材

C/C++ 程序设计教程

梁成升　编著

国防工业出版社

·北京·

图书在版编目(CIP)数据

C/C++程序设计教程/梁成升编著.—北京:国防工业出版社,2009.5
普通高等学校"十一五"规划教材
ISBN 978-7-118-06238-0

Ⅰ.C... Ⅱ.梁... Ⅲ.C语言-程序设计-高等学校-教材 Ⅳ.TP312

中国版本图书馆CIP数据核字(2009)第031880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)
北京奥鑫印刷厂印刷
新华书店经售
*
开本 787×1092 1/16 **印张** 19¾ **字数** 486千字
2009年5月第1版第1次印刷 **印数** 1—4000册 **定价** 33.00元

(本书如有印装错误,我社负责调换)

国防书店:(010)68428422 发行邮购:(010)68414474
发行传真:(010)68411535 发行业务:(010)68472764

前　言

C语言是一种通用的程序设计语言，也是许多高校为学生开设的第一门程序设计课程。随着计算机技术的发展，面向对象程序设计方法逐步得到推广，C++则是当前被广泛使用的、支持面向对象的程序设计语言之一，许多学习C语言的读者希望了解C++语言。鉴于此种情况，特编写了这本《C/C++程序设计教程》。

本书充分考虑程序设计入门的教学特色，理论上做到必须、够用，注重联系实际，突出实用性；语言组织上通俗易懂，做到在内容的编排上尽量符合初学者的要求。本书主要特点可归纳如下：

(1) 本书是在介绍C语言相关知识的基础上讲解C++程序设计的，内容编排合理、知识讲解循序渐进。

(2) 本书是在作者教学讲义的基础上编写的，内容切合实际，语法、算法科学，语言叙述准确。

(3) 本书使用Visual C++ 6.0运行环境，有利于学生对后续课程(如VC++、C#等计算机语言课程)的学习。

努力提高读者的编程能力是本书的目的，加强实践是达此目的的重要途径。本书中的所有源程序均已通过上机调试，希望读者能够坚持纸上编程与上机运行相结合，学习C/C++语法与掌握解决问题算法相结合，全面提高自己的编程能力。

在编写此书的过程中，参阅了大量国内外相关资料，引用了许多专家的研究成果，得到了不少同行老师的热忱帮助，在此一并致谢！另外，由于编者水平和经验有限，加之计算机技术发展迅速，疏漏和错误在所难免，希望广大读者批评指正。

编　者

目　录

第1章　C语言概述

随着计算机技术的迅速发展，软件开发领域出现多种程序设计语言。C 语言作为生命力最强的高级程序设计语言之一，非常适合编写系统程序，曾被誉为是真正的程序设计者的语言。C 语言流行广、影响大。我国自 1994 年推出二级 C 语言以来，C 语言备受社会关注，已成为讲解程序设计的标准语言。

由于面向对象技术的发展和广泛应用，C 语言作为结构化程序设计语言已逐渐被面向对象的程序设计语言 C++替代。C++是一种更好的 C 语言，是 C 语言的一个超集，是 C 语言的改良版本。因此，想要学好 C++语言，就要首先学习和掌握好 C 语言。在讲解 C 语言之前，我们先熟悉程序设计的一般过程、算法和结构化程序设计思想。

1.1　程序设计方法

1.1.1　程序设计概述

程序是为解决某一问题而编写的一组有序指令的集合。通常，将解决一个实际问题的具体操作步骤用某种程序设计语言描述出来，就形成了程序。计算机程序设计语言可以归纳为机器语言、汇编语言和高级语言三类。

1. 机器语言

机器语言是计算机硬件系统可识别的二进制指令构成的程序设计语言。机器语言是面向机器的语言，与特定的计算机硬件设计密切相关，因机器而异，可移植性差。它的优点是机器能够直接识别，执行速度快。缺点是记忆、书写、编程困难，可读性差且容易出错，因此就产生了汇编语言。

2. 汇编语言

汇编语言是一种用助记符号代表等价的二进制机器指令的程序设计语言。汇编语言也是一种直接面向计算机所有硬件的低级语言，但计算机不能直接执行汇编语言程序，必须经过汇编程序翻译成机器语言程序后才能在计算机上执行。可以看出，从机器语言到汇编语言是计算机语言发展史上里程碑式的进步。

3. 高级语言

高级语言是一种用接近自然语言和数学语言的语法、符号描述基本操作的程序设计语言。它符合人的逻辑思维方式，简单易学。目前，常见的高级语言有 Visual Basic、Java、C、C++、C#、Delphi 等。用高级语言编写的程序通常称为“源程序”，而由二进制的“0”、“1”代码构成的程序称为“目标程序”。用高级语言编写的程序计算机同样不能直接执行，要用翻译程序将其转换成机器语言目标程序后才能执行。例如用 C 语言编写的程序，必须先经 C 编译系统翻译成目标程序，再连接成可执行文件后才能执行。

1.1.2 程序设计的一般过程

一个程序应包括数据的输入、数据的加工处理和数据的输出三大部分。要设计出一个好的程序，必须了解计算机程序设计的过程，如图 1.1 所示。

图 1.1 程序设计的一般过程

1. 分析问题

按照任务所提出的要求，对要处理的任务进行调查分析，明确要实现的功能，并选择合适的解决方案。要分析哪些是原始数据，如何得到这些数据，怎样处理这些数据，结果如何输出等，从而确定和设计数据的组织方式，即数据结构。

2. 确定算法

算法是解决问题的方法和步骤。世界著名的计算机科学家沃思（Wirth）曾提出一个用来表达程序设计实质的著名公式：程序=算法+数据结构。就是说："程序是在数据的特定的组织方式的基础上，对抽象算法的具体描述"。因此，在分析问题时，必须认真考虑数据结构，然后针对具体的数据结构设计相应的解决问题的方法和步骤，即算法。

描述算法的方法有多种，常用的有自然语言、结构化流程图、N-S 图和伪代码等。在这里主要介绍用自然语言描述和用流程图描述算法的方法。

1) 用自然语言描述

用自然语言描述算法通俗易懂，但容易出现歧义，且表达的算法与计算机高级语言形式差距较大，通常只在很简单的问题中使用。

例 1.1 求整数 1 到 100 的和。

首先我们分析这个问题，明确它有 3 个功能：输入 1 到 100 的整数，求和，输出结果；要处理的原始数据为：1 到 100 的整数，它可以根据表达式的规律自动生成，即 n 的初值为 1，它的后一项比前一项多 1，一直加到最后项为 100 止；要进行的处理为：累加求和，并显示。这里用 sum 代表和，用 n 代表 1 到 100 的某项值。因此，该问题的数据组织形式为：sum=sum+n,，n=n+1，其中 n≤100。

用自然语言描述如下所示：

(1) sum 置 0、n 置 1。

(2) sum 进行累加→sum=sum+n。

(3) n 增加 1→n++。

(4) 如果 n≤100，表示数据还未处理结束，返回第(2)步，然后继续重复第(2)、(3)步；否则，计算结束，转下一步。

(5) 输出结果 sum。

2) 用流程图描述

流程图是用图形、箭头和文字说明来表示算法的框图。用流程图的优点是形象直观、通俗易懂，是描述算法的一种很好的工具，尤其是对于较复杂的问题，能将设计者的思路清楚地表达出来。流程图一般由以下规定使用的基本框图组成，如图 1.2 所示。

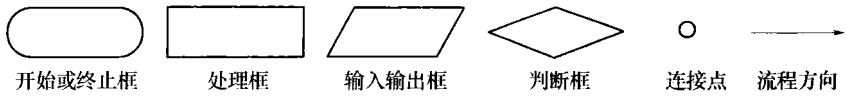

图 1.2　流程图基本框图

一般流程图由顺序、选择和循环基本结构组成，这 3 种基本结构可以相互嵌套，形成结构化的流程图。由结构化的流程图编出的程序也是结构化的程序。

用结构化的流程图的方式描述例 1.1 的算法，如图 1.3 所示。

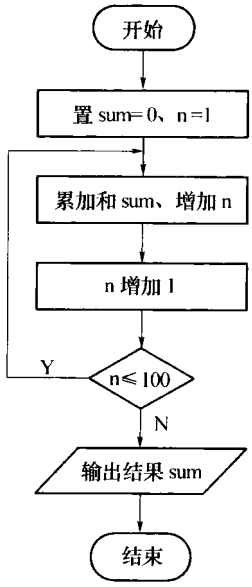

图 1.3　求 1 到 100 整数和的流程图

3) 用 N-S 描述还可采用流程图表示算法

1973 年美国学者提出了一种新的流程图形式。在这种流程图中，完全去掉了带箭头的流程线。全部算法写在一个矩形框内，在该框内还可以包含其他的从属于它的框，或者说，由一些基本的框组成一个大的框。这种流程图又称 N-S 结构化流程图（N、S 是两位美国学者的英文姓氏的首字符）。这种流程图适应于结构化程序设计，而且作图简单，占空间小，一目了然，因而很受欢迎。

N-S 流程图采用以下的流程图符号：

(1) 顺序结构。顺序结构用图 1.4（a）表示，A 和 B 两个框组成一个顺序结构。

(2) 选择结构。选择结构用图 1.4（b）表示。当 p 成立时执行 A 操作，当 p 不成立时执行 B 操作。注意，此图是一个整体，代表一个基本结构。

(3) 循环结构。循环结构用图 1.4（c）与（d）表示。其中图（c）称当型循环，即首先判断条件是否成立，当 p1 成立时反复执行 A 操作，直到条件 p1 不成立为止。图（d）是先执行 A 操作，再判断条件 p1，若不成立反复执行 A 操作，直到条件 p1 成立为止。

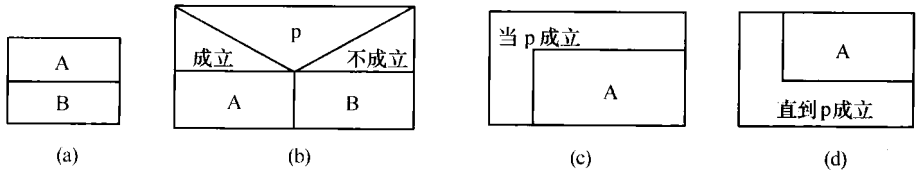

图 1.4　N-S 流程图符号

3. 编写程序

根据以上确定的算法，用某种计算机语言实现这种算法，就是编写源程序。例如，用 C 语言编写源程序。编写源程序时要正确地使用语言，准确地描述算法，这是一项细致的工作，要经过不断的练习才能成为一个熟练的编程人员。

源程序：

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

```
void main()
{int n,sum=0;
for (n=1;n<=100;n++)
sum=sum+n;
printf("sum=1+2+3+…+100=%d\n",sum);}
```

运行结果：sum=1+2+3+…+100=5050

其中的语句及函数在以后将陆续学到。

4. 调试程序

对于复杂的问题并不是编出来就能用的，要经过多次排错、调试及试运行，才可能得到正确运行的程序。

5. 整理文档

程序调试通过后，应该整理资料，编写程序使用说明书及程序所要求的软硬件环境等技术性文档。

1.1.3 结构化程序设计思想

结构程序设计就是一种进行程序设计的原则与方法，按照这种原则和方法可设计出结构清晰、容易理解、容易修改、容易验证的程序。即结构化程序设计是按照一定的原则与原理，组织和编写正确且易读的程序的软件技术。结构化程序设计的目标在于使程序具有一个合理的结构，以保证和验证程序的正确性，从而开发出正确的、合理的程序。那么在编写结构化的程序时，应该采用结构化的程序设计思想。它包括两个方面的内容：模块化设计和结构化编码。

1. 模块化设计

程序设计的第一步是分析问题。人脑的思维能力的局限性与处理问题的复杂性之间存在如下矛盾：当我们分析一个较小的问题时，可以考虑到问题的每一个细节，而当我们分析一个庞大而又复杂的问题时，要想直接考虑到问题的每一步具体操作几乎是不可能的。但是，通常一个完整的系统，往往由成千上万个具体的操作组成，即要实现几百行、几千行乃至几十万行的程序，该如何处理呢？

解决办法是采用“自顶向下，逐步细化和精化”的模块化设计方法。这种模块化设计方法的基本思想是，将一个复杂的大任务逐层分解出许多易于理解和实现的、逻辑上相对独立的子任务，形成一棵功能树。然后分别实现各个子任务。这样，对一个大任务的处理就变成了对一个个小的子任务的实现，使问题变得简单了。

功能树上的每一个子任务在 C 语言中对应一个函数，也就是说每个函数是一个功能模块。一般的划分原则是每个功能模块完成一项相对独立的功能，模块之间的联系尽量少，以保证各模块的独立性。

2. 结构化编码

在设计好结构化的算法之后，还需进行结构化的编码，才能得到结构化的程序。用结构化的程序设计语言正确地描述算法，就是结构化编码。在 C 语言中有直接描述三种基本结构和基本操作的语句，用这些语句将算法表示出来，就是 C 的源程序。

学习 C 语言，不仅仅是学习 C 语言的语法，更重要的是学会设计算法和用 C 语言编码，从而掌握程序设计的一般方法。

1.2 C语言简介

1.2.1 C语言的发展

C语言是在B语言的基础上发展起来的，最早可追溯到ALGOL 60语言。1960年出现的ALGOL 60语言是一种面向问题的高级语言，离计算机硬件太远，太抽象，不宜用来编写系统程序。1963年剑桥大学和伦敦大学推出了CPL（combined programming language）语言。CPL语言在ALGOL 60的基础上接近硬件一些，但规模比较大，不易实现。1967年剑桥大学推出BCPL(basic combined programming language)语言，它具有结构化好，具有指针处理方式和直接访问内存地址等功能。1970年贝尔实验室对BCPL进行简化并设计出B(取BCPL的第一个字母)语言，用B语言编写了第一个UNIX操作系统。但B语言过于简单，功能有限。1972年贝尔实验室又在B语言的基础上设计出C语言。C语言既保持了B语言精练和接近计算机硬件的优点，又克服了B语言过于简单和数据无类型的缺点。

最初的C语言只是为描述和实现UNIX操作系统而设计开发的。但随着C语言的不断发展和应用的普及，C语言可以在多种操作系统下运行，并且产生了不同版本的C语言系统。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语言进行发展和扩充，制定了新的标准，称为ANSI C。1987年，ANSI又公布了新标准87 ANSI C。目前流行的C编译系统都是以它为基础的。现在计算机上常使用的C语言系统有Microsoft C、Turbo C、Quick C等。

随着面向对象技术的发展，在C语言的基础上增加了面向对象的程序设计功能，于1983年由贝尔实验室设计了C++。C++语言的主要特点是全面兼容C语言和支持面向对象的编程方法。目前流行的软件开发工具以Visual C++6.0和C++ Builder 5.0为代表。由于Windows软件的广泛使用，Visual C++6.0是程序员选择的主要编程工具，本书C++部分的实例和练习题都是在Visual C++6.0集成环境下运行的。

1.2.2 C语言的特点

多年来，C语言经久不衰并不断发展，主要是由于它具有以下特点。

1. 结构化程序设计语言

C语言支持结构化程序设计的3类基本控制结构：顺序、选择和循环结构，如if…else语句、switch语句、while语句、do…while语句、for语句等。用函数作为程序模块以实现程序的模块化。而且C语言源程序是由多个函数模块组成的，每个函数功能独立，可以单独进行编译，生成的目标代码质量高，而且可以与其他语言连接生成可执行文件，因而它的可读性及可移植性好，调试和维护方便。

2. 语言简洁，使用方便、灵活

C语言只有32个关键字、45个标准运算符和9种控制语句。程序书写形式自由，区分大小写，主要用小写字母表示。语言的许多成分都是通过调用函数来实现的，使其程序简练而功能强大。

3. 运算符丰富

C语言把括号、赋值、强制类型转换等都作为运算符处理，使C的运算类型极其丰富，