全国高等教育自学考试指定教材 计算机及应用专业（专科）

高级语言程序设计

附：高级语言程序设计自学考试大纲

课程代码
0342
[2007年版]

组编／全国高等教育自学考试指导委员会
主编／迟成文

本教材附赠网络学习卡

经济科学出版社

全国高等教育自学考试指定教材
计 算 机 及 应 用 专 业（专 科）

高 级 语 言 程 序 设 计

（附：高级语言程序设计自学考试大纲）
（2007 年版）

全国高等教育自学考试指导委员会　组编
迟成文　主编

经 济 科 学 出 版 社

图书在版编目（CIP）数据

高级语言程序设计／全国高等教育自学考试指导委员会组编．
—北京：经济科学出版社，2007.2
全国高等教育自学考试指定教材．计算机及应用专业：
专科
ISBN 978-7-5058-6092-6

Ⅰ．高… Ⅱ．全… Ⅲ．高级语言-程序设计-高等教育-
自学考试-教材 Ⅳ．TP312

中国版本图书馆 CIP 数据核字（2007）第 011112 号

责任编辑：杨　梅
责任校对：杨　海
版式设计：代小卫
技术编辑：邱　天

高级语言程序设计
（附：高级语言程序设计自学考试大纲）
（2007 年版）
全国高等教育自学考试指导委员会　组编
迟成文　主编
经济科学出版社出版
社址：北京海淀区阜成路甲 28 号　邮编：100036
网址：www.esp.com.cn
电子邮件：esp@esp.com.cn
北京友谊印刷有限公司印装
787×1092　16 开　18.375 印张　430000 字
2007 年 3 月第一版　2007 年 3 月第 1 次印刷
印数：00001-20100 册
ISBN 978-7-5058-6092-6/F·5353　定价：28.00 元
本书如有质量问题，请与教材供应部门联系。

组编前言

21世纪是一个变幻莫测的世纪，是一个催人奋进的时代，科学技术飞速发展，知识更替日新月异。希望、困惑、机遇、挑战，随时随地都有可能出现在每一个社会成员的生活之中。抓住机遇、寻求发展、迎接挑战、适应变化的制胜法宝就是学习——依靠自己学习、终生学习。

作为我国高等教育组成部分的自学考试，其职责就是在高等教育这个水平上倡导自学、鼓励自学、帮助自学、推动自学，为每一位自学者铺就成才之路。组织编写供读者学习的教材就是履行这个职责的重要环节。毫无疑问，这种教材应当适合自学，应当有利于学习者掌握、了解新知识、新信息，有利于学习者增强创新意识，培养实践能力，形成自学能力，也有利于学习者学以致用，解决实际工作中所遇到的问题。具有如此特点的书，我们虽然沿用了“教材”这个概念，但它与那种仅供教师讲、学生听，教师不讲、学生不懂，以“教”为中心的教科书相比，已经在内容安排、编写体例、行文风格等方面都大不相同了。希望读者对此有所了解，以便从一开始就树立起依靠自己学习的坚定信念，不断探索适合自己的学习方法，充分利用已有的知识基础和实际工作经验，最大限度地发挥自己的潜能，以达到学习的目标。

欢迎读者提出意见和建议。

祝每一位读者自学成功！

全国高等教育自学考试指导委员会

2006年12月

编 者 的 话

本书是按照全国高等教育自学考试指导委员会最新颁布的计算机及应用专业（专科）《高级语言程序设计》课程自学考试大纲编写的自学教材。

C 语言是目前广泛用于软件开发的一种编译型程序设计高级语言。与其它高级语言相比，C 语言具有丰富的数据类型和运算符，极其简单的语句，源程序清单简洁清晰；具有汇编语言的能力，可以直接处理硬件系统和设备接口的控制；是一种结构化程序设计语言，支持自顶向下的结构化程序设计技术；具有完善的模块化结构，为中大型软件设计中采用模块化程序设计方法提供了基础。

本书以目前微型机上流行的 ANSI C 为版本，兼顾集成化环境 Turbo C 编译程序，系统地、循序渐进地介绍了 C 语言的数据类型和运算符、语句格式和功能、结构化程序设计思想和方法。全书共分为 9 章。第 1 章介绍程序、计算机语言、算法等基本概念。第 2 章介绍 C 语言的基本字符集、基本词类、基本句类和基本的程序结构，以及 Turbo C 编译程序的基本使用方法。第 3 章介绍基本数据类型（整型、实型、字符型）和字符串常量，以及运算符和表达式。第 4 章介绍三种基本控制结构（顺序、选择、循环）的相关语句和程序设计方法。第 5 章介绍数组的使用及其程序设计方法。第 6 章介绍指针的使用及其程序设计方法。第 7 章介绍函数的设计和调用方法。第 8 章介绍结构型数据的特点、定义及其程序设计方法。第 9 章介绍文件处理及其程序设计方法。

本书是全国计算机及应用专业（专科）的自学考试指定教材，在本书后面配有本课程的自学考试大纲供读者参考。考虑到本书是以自学为主的教材，所以在撰写时，力求取材得当、通俗易懂、结构清晰、层次分明，通过精选的大量例题帮助考生加深对语法知识和程序设计方法的理解，章末配有大量与自学考试题型相同的习题，作为自我测试之用。

本书的第 1、2、8、9 章由牛犁编写，第 3、4、5、6、7 章由迟成文编写，各章例题和习题由迟琳和陈兵在计算机上调试，全书由迟成文统稿。

在本书的编写过程中，参考了大量有关 C 语言程序设计的书籍和资料，编者在此对这些参考资料的作者表示感谢。南京大学陈本林教授、中国科学技术大学刘振安教授、上海交通大学陈维钧副教授仔细审读了“自学考试大纲”和本书的原稿，提出了大量宝贵的修改意见，在此表示衷心的感谢。

由于编者水平有限，书中难免存在错误和不当之处，敬请广大读者不吝赐教，以便再版时修改。

迟成文

2006 年 9 月

内容提要

本书是全国高等教育自学考试指导委员会组编的计算机及应用专业（专科）自学考试指定教材，包括自学考试大纲和教材的所有内容。

本书全面介绍了高级语言 C 的基本语法知识和程序设计方法。内容循序渐进，语言通俗易懂，便于自学；例题丰富，习题数量和难易程度适中，便于练习和自我考查。

本书既是自学考试指定教材，也可作为各类院校计算机及应用专业（专科）学生的教材或参考书，对从事程序设计的技术人员也具有较好的参考价值。

目　录

高级语言程序设计

高级语言程序设计自学考试大纲

第1章 概 述

本章介绍计算机程序、程序设计、高级语言等基本概念，着重介绍算法及其描述方法，后者是学习和掌握C语言程序设计的基础。

1.1 程序与程序设计

1.1.1 程序

从自然语言角度来说，程序是对解决某个问题的方法步骤的描述；从计算机角度来说，程序是用某种计算机能理解并执行的计算机语言描述解决问题的方法步骤。

程序的特点是有始有终，每个步骤都能操作，所有步骤执行完后，问题便得到解决。

例如，某个会议的安排如下：

第一项 宣布会议开始。

第二项 全体起立唱国歌。

第三项 宣读嘉奖令。

第四项 颁发奖励证书。

第五项 宣布会议结束。

上述步骤就是一个解决嘉奖问题的程序。

又如，求一元一次方程 $ax+b=0$ 的根。解决这个问题的算法如下：

第1步 开始。

第2步 输入方程的系数a、b。

第3步 判断a的值：

若 $a\neq0$，计算并输出根（$\frac{-b}{a}$）；转第4步。

否则（即 $a=0$），输出“方程无根”，转第4步。

第4步 结束。

1.1.2 程序设计

程序设计就是分析解决问题的方法步骤，并将其记录下来的过程。从自然语言角度来说，就是用自然语言记录；从计算机角度来说，必须用计算机语言记录下来。上面的例子都

是用自然语言记录的程序，其中第二个程序还可以用计算机语言来记录，例如用 C 语言，记录结果如下：

```
#include  <math.h>                    /*程序中使用了求绝对值的系统函数*/
void main()
{ float a, b; x;                      /*说明存放数据的变量*/
 scanf("%f,%f", &a, &b);              /*输入 a、b 两个数据*/
 if (fabs(a) >0.00001)                /*判断 a 的绝对值是否 > 10^-5，是则认为 a 不是 0*/
    {x = -b/a;                        /*若 a 不是 0，计算根 x*/
     printf("x = %f\n", x);           /*输出根*/
    }
 else
     printf("Does not have root!\n"); /*若 a 是 0，输出没有根*/
}
```

这是一个计算机程序，编写程序（包括程序输入、调试直到正确）的过程称为程序设计。

1.2 高级语言与 C 语言

1.2.1 计算机程序设计语言

在计算机的发展过程中，出现了各种各样的计算机程序设计语言。

最早期的语言是二进制语言，程序设计人员用计算机能直接识别和执行的二进制代码来编写程序。为了减轻程序设计人员的负担，很快又出现了汇编语言，这种语言是用符号来代表二进制代码的，所以称为符号语言。用这种语言编写的程序需要通过一种软件（称汇编程序）翻译后才能执行，所以又称汇编语言。不同的计算机上提供不同的二进制语言或汇编语言，所编写的程序一般只能在同类型的计算机上运行，所以这种语言又称为“面向机器的语言”。

程序设计的关键是将解决问题的方法步骤（称为算法）描述出来，设计人员很快就提供了一种描述算法过程很方便，同时脱离了对机型的要求，能在任何计算机上运行的计算机语言。程序设计人员可以利用这种计算机语言直接写出各种表达式来描述简单的计算过程，这种语言提供的各种控制语句可以描述复杂的加工处理过程。专家们将这种语言称为“高级语言”，而将二进制语言和汇编语言统称为“低级语言”。由于高级语言是面向算法过程进行描述的，所以又将高级语言称为“面向过程的语言”。

高级语言编写的程序称为“源程序”。源程序是不能在计算机上直接运行的，必须将其翻译成二进制程序后才能执行。翻译过程有两种方式：一种是翻译一句执行一句，称为“解释执行”方式，完成翻译工作的程序就称为“解释程序”；另一种是全部翻译成二进制程序后再执行，承担翻译工作的程序就称为“编译程序”，编译后的二进制程序称为“目标程序”。

世界上第一个高级语言是“ALGOL（算法语言的缩写）”，第二个高级语言是“FORTRAN（公式翻译的缩写）”语言。以后陆续出现了很多种高级语言，使用较广的有 BASIC 语言、COBOL 语言、PASCAL 语言和 C 语言等。

1.2.2 C 语言

C 语言的前身是 ALGOL 语言。1960 年推出 ALGOL60 版本后，很受程序设计人员的欢迎。用 ALGOL60 来描述算法很方便，但是用它编写硬件设备的处理程序比较困难，不宜用来编写系统程序。1963 年英国剑桥大学在 ALGOL 语言基础上增添了处理硬件的能力，并命名为“CPL（复合程序设计语言的缩写）”。CPL 由于规模大，学习和掌握困难，没有流行开来。1967 年剑桥大学的马丁·理查德对 CPL 语言进行了简化，推出名为“BCPL（基本复合程序设计语言）”。1970 年美国贝尔实验室的肯·汤普逊对 BCPL 进行了进一步的简化，突出了硬件处理能力，并取了“BCPL”的第一个字母“B”作为新语言的名称。同时用 B 语言编写了 UNIX 操作系统。1972 年贝尔实验室的布朗·W·卡尼汉和丹尼斯·M·利奇对 B 语言进行了完善和扩充，在保留 B 语言强大的硬件处理能力的基础上，扩充了数据类型，恢复了通用性，并取了“BCPL”的第二个字母 C 作为新语言的名称。此后，两人合作，重写了 UNIX 操作系统。C 语言伴随着 UNIX 操作系统成为一种很受欢迎的计算机高级语言。

1977 年，为了让 C 语言脱离 UNIX 操作系统，成为在任何计算机上都能运行的通用计算机语言，卡尼汉和利奇（K&R）撰写了“C 程序设计语言”一书，对 C 语言的语法进行了规范化的描述，成为当时的标准。随着微型机的普及，出现了不同的 C 语言版本，为了统一标准，美国标准化协会（ANSI）于 1987 年制定了 C 语言标准，称为“ANSI C”。通常将 K&R 的标准称为旧标准，将“ANSI C”称为新标准。

目前在微机上使用的 C 编译程序有：Turbo C、Microsoft C、Quick C。本书将以“ANSI C”为标准介绍 C 语言语法知识，以“Turbo C2.0”为编译程序介绍 C 程序调试方法。

C 语言的主要特点可以概括如下：

（1）比其它高级语言更接近硬件，比低级语言更容易描述算法，程序易编、易读、易查错、易修改。可以说兼有高级语言和低级语言的优点。

（2）数据类型和运算符十分丰富，程序设计和算法描述更为简单和方便。

（3）语法结构简单，语句数目少，简单易学。

（4）它是一种结构化程序设计语言，提供了完整的程序控制语句（选择语句和循环语句），很适合结构化的程序设计方法。

（5）它是一种模块化程序设计语言，适合大型软件的研制和调试。

（6）它提供了大量的库函数供调用，简化了程序设计工作。

1.3 算法及其描述

1.3.1 算法

广义地说，算法就是解决问题的方法。

例如，某个会议的具体安排就是完成“开会”这个问题的算法。又如，一元二次方程的求根公式就是解决一元二次方程求根的算法。这些都是广义上的算法。

从计算机角度来说，算法是计算机程序中解决问题的方法步骤。在计算机的程序设计中，算法应该能够分解成具体的若干个操作步骤，而且，每一个步骤都是能用某种计算机语言提供的语句或者语句串来完成的。

下面我们来看两个算法的例子。

【例 1-1】 求 $1+2+3+\cdots+99$。

解决这个问题的算法有很多种。

算法一：第 1 步：开始。

第 2 步：计算 $s=\frac{1+99}{2}\times 99$。

第 3 步：输出 s 中的结果。

第 4 步：结束。

注意，算法必须是有始有终的，所以算法步骤中必须有“开始”和“结束”两个步骤。有时为了简化算法步骤，通常第 1 步可以看成是隐含着“开始”，但“结束”应在算法步骤中明确给出。

算法二：第 1 步：将 1 放在 s 中；

第 2 步：计算 s+2，并保存在 s 中；

第 3 步：计算 s+3，并保存在 s 中；

第 4 步：计算 s+4，并保存在 s 中；

…

第 99 步：计算 s+99，并保存在 s 中；

第 100 步：输出 s 中的结果；

第 101 步：结束。

算法三：第 1 步：将 0 放在 s 中；

第 2 步：将 1 放在 n 中；

第 3 步：将 n 加到 s 中；

第 4 步：将 1 加到 n 中；

第 5 步：判断。若 n<100，转第 3 步；

否则，转第 6 步；

第 6 步：输出 s 中的结果，结束。

【例 1-2】 求一元二次方程 $ax^2+bx+c=0$（设 $a\neq0$）的实数根。

对应算法的步骤如下：

第 1 步　获得系数 a、b、c。

第 2 步　计算 $d=b^2-4ac$。

第 3 步　若 $d>0$　计算：$x1=\frac{-b+\sqrt{d}}{2a}$，$x2=\frac{-b-\sqrt{d}}{2a}$

输出：有两个实数根，分别为 x1 和 x2，转第 6 步；

否则，转第 4 步。

第 4 步　若 $d<0$，则输出：没有实数根，转第 6 步；

否则转第 5 步。

第 5 步　计算：$x1=x2=\frac{-b}{2a}$，输出：有两个相同的实数根，为 x1，转第 6 步。

第 6 步　结束。

算法是程序的灵魂，是程序设计的第一步。对初学程序设计的人来说，正确的选择算法，分解出算法步骤是完成程序设计的关键。

1.3.2　算法的描述

算法可以用自然语言来描述，例如上面介绍的“求 1+2+3+…+99”的算法、“求一元二次方程的实根”的算法就是用中文来描述的。

对于复杂算法，用自然语言很难清楚地描述算法步骤的流向和结构。这里，我们介绍一种称为“流程图”（又称为“框图”）的描述方法。它用标准的图形元素来描述算法步骤，结构一目了然。

组成流程图的常用图形元素如图 1-1 所示。

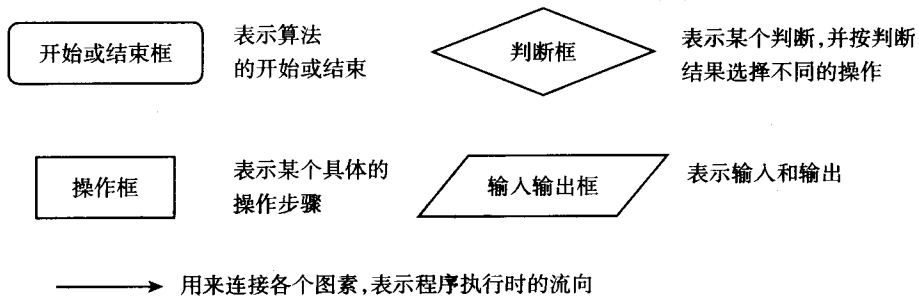

图 1-1　组成流程图的常用图素

图 1-2 和图 1-3 分别是求 1+2+…+99 的算法一和算法三对应的“流程图”，从中可以清晰地看出算法执行的流程。

图 1-4 是求一元二次方程实根算法对应的“流程图”，从图中可以清晰地看出算法执行的流程。