全国计算机等级考试（新大纲）应试用书

全国计算机等级考试

二级教程

——C语言程序设计

■ 本书编写组

计算机等级考试

人民邮电出版社
POSTS & TELECOM PRESS

全国计算机等级考试（新大纲）应试

全国计算机等级考试

二级教程
——C语言程序设计

本书编写组

人民邮电出版社

图书在版编目(CIP)数据

二级教程——C 语言程序设计 / 本书编写组编. —北京：人民邮电出版社，2005.2
全国计算机等级考试（2004 年大纲）应试用书
ISBN 7-115-12994-0

Ⅰ. 二… Ⅱ. 本… Ⅲ. C 语言—程序设计—水平考试—自学参考资料 Ⅳ. TP312

中国版本图书馆 CIP 数据核字（2004）第 143483 号

内 容 提 要

本书是根据教育部考试中心最新修订的《全国计算机等级考试大纲（2004 年版）》“二级 C 语言程序设计考试大纲”的要求编写而成。全书共分 10 章，主要内容包括：C 程序设计基础、顺序结构、选择结构、循环结构、数组、函数、指针、结构体和共用体、编译预处理和位运算以及文件。

本书结构合理、语言清晰简明，并以较多的实例讲解 C 语言的语法现象和规则，在每一章后专门列出一节，有目的地选取了一些典型题目进行分析，指出解题的要点，在每章末尾，收集了较多的练习题，使应试者能在短时间内把握主要内容，掌握解题要点并顺利地通过考试。

本书可作为全国计算机等级考试应试用书，也可作为 C 语言学习的参考书。

全国计算机等级考试（新大纲）应试用书
全国计算机等级考试
二级教程——C 语言程序设计

◆ 本书编写组

◆ 人民邮电出版社出版发行　北京市崇文区夕照寺街 14 号
邮编 100061　电子函件 315@ptpress.com.cn
网址 http://www.ptpress.com.cn
读者热线 010-67129259
北京隆昌伟业印刷有限公司印刷
新华书店总店北京发行所经销

◆ 开本：787×1092 1/16
印张：18.25
字数：440 千字　2005 年 2 月第 1 版
印数：1－4 000 册　2005 年 2 月北京第 1 次印刷

ISBN 7-115-12994-0/TP・4391

定价：24.00 元

本书如有印装质量问题，请与本社联系　电话：（010）67129223

编 者 的 话

全国计算机等级考试已经经过了近十年的发展，在2004年，教育部考试中心根据等级考试形势的发展和出现的新问题，对计算机等级考试大纲进行了修订，对二级考试开考的程序设计语言的语种进行了调整，停考了一些语种，增加了几门新的程序设计语言。

本书是全国计算机等级考试二级C语言程序设计的教程，全书根据2004年版考试大纲中对C语言的要求编写而成。全书共分10章，主要内容包括：C程序设计基础、顺序结构、选择结构、循环结构、数组、函数、指针、结构体和共用体、编译预处理和位运算、文件。

作为二级考试程序设计语言中从开考至今惟一保留的一个语种，C语言仍然具有广泛的应用价值，但是由于C语言语法现象比较复杂，程序编写比较自由，使得学生在学习中有些概念不容易理解，在初学编程时比起其他语言来说入门也困难一些，为此，本书力求使应试者比较容易地掌握编程的技巧。

本书结构合理、语言清晰简明，并以较多的实例讲解C语言的语法现象和规则，并在每一章后专门列出一节，有目的地选取了一些典型的题目进行分析，其中有些题目来自往年的考试中。对这些题目，指出解题的要点，在解题中，不但给出正确答案，同时也解释这样做的理由以及其他错误选项不对的原因。在有些题目中，对一些基本的程序设计算法也作了归纳和整理，例如求解一元方程的几种迭代解法、数据排序的基本算法和相应的程序设计，目的是帮助应试者全面地掌握解题的方法并能顺利地编写完整的程序。

在每章末尾，收集了较多的练习题，题型包括了选择题、填空题和编程题，前两类是笔试的题型，编程题则是上机考试必须掌握的，因此认真完成这些题目是掌握所学内容的关键。

书末附有一套模拟试题供应试者自行测试。

需要说明的是，考试大纲规定，二级考试的内容为基础知识和某门程序设计语言两部分，由于基础知识部分另有单独教材，本教程中仅包括C语言程序设计的内容，因此，作为应试者要全面准备这两个方面的内容。

本书对于参加等级考试的应试者，是一本实用的教材。同时，对于从事C语言程序设计人员，本书也有一定的参考价值。

由于编写时间仓促以及本人的水平有限，书中难免出现疏漏或错误，恳请读者不吝赐教。

编　者

2004年10月

目　录

第 1 章　C 程序设计基础

本章介绍 C 语言的程序组成，程序中的基本数据类型、常量、变量和表达式的概念及使用，为结构化程序设计做好准备。

1.1　C程序概述

1.1.1　C 语言简介

C 语言是结构化程序设计语言之一，和其他高级程序设计语言相比，C 语言自身具有以下显著的特点。

（1）语言简练

C 语言中共有 32 个关键字，其中和数据类型有关的有 15 个，和存储类型有关的有 4 个，用于流程控制的有 12 个，以及 1 个运算符 sizeof。

（2）程序设计灵活

主要是指程序书写形式自由和编程的自由度大，例如，对于数据类型，允许整型数据、字符型数据和逻辑型数据通用；又如，语法检查比较宽轻，尤其是不检查数组下标的越界，这就要求程序员自己保证程序的正确性。

（3）运算符丰富

C 语言中提供了 34 种运算符，可以构成多种表达式，完成复杂的运算。

（4）数据类型丰富

除了整型、实型和字符型等基本数据类型外，还提供了像数组、指针和结构体等构造数据类型，因此可以处理复杂的数据结构。

（5）直接访问物理地址和位运算

这使得 C 语言具有汇编语言的一些功能，是其他高级语言不具备的。

（6）函数式的语言

C 语言以函数作为程序的模块，实现模块化、结构化。

正因为 C 语言具有上述的特点，使其既可以编写系统软件，也可以编写应用软件。

1.1.2　C 语言的程序

1. C 语言程序中函数的组成

由于 C 语言中是以函数作为程序的模块，模块之间的关系通过函数调用实现，这里先通过例子说明 C 语言中一个完整的函数的组成。

【例 1-1】 计算两个整数的和，源程序如下：

```
main()
{
int a,b,sum;
a=3;
b=4;
sum=a+b;
printf("%d+%d=%d\n",a,b,sum);
}
```

这是一个完整的程序，程序仅由一个函数组成，组成该函数的各行含义如下：

第 1 行 main()是函数的说明部分，表示函数名为 main；

第 2 行和第 8 行的一对花括号表示由它括起来的整个部分是函数的函数体，注意到函数体内的每一行都以分号结束；

第 3 行 int a,b,sum;是变量的类型说明语句，作用是定义了在该函数中要用到的 3 个整型变量 a、b 和 sum；

第 4 行的 a=3;和第 5 行的 b=4;是两个赋值语句，作用是将两个数 3 和 4 分别赋给变量 a 和 b；

第 6 行的 sum=a+b;也是赋值语句，先计算 a 与 b 之和，再将结果赋给变量 sum；

第 7 行是函数调用语句，用来调用 C 语言的库函数 printf，作用是按指定格式输出变量 a、b 和 sum 的结果。

在 printf 的双引号中，3 个"%d"指定数据以十进制整数输出，"\n"是回车换行符，其他的字符称为普通字符，照原样输出。

这样，这个程序的运行结果是：

```
3+4=7
```

2. C 源程序的组成

【例 1-2】输出两个整数中较大的一个数，程序如下：

```
int max(int x,int y)
{
  int z;
  if(x>y)
    z=x;
  else
    z=y;
  return (z);
}
main()
{
  int a,b,c;
  a=3; b=4;
  c=max(a,b);                    /*调用函数 max，将最大值返回给变量 c*/
  printf("max=%d\n",c);
```

```
}
```

这个程序由两个函数组成，分别是 main 和 max，下面解释和上例不同的地方。

（1）函数 max 的说明部分为：

```
int max(int x,int y)
```

函数名 max 前的 int 表示该函数的结果是整型，max 括号内的 int x,int y 表示该函数有两个形式参数（简称形参）x 和 y 并且都是整型。

（2）函数 max 的函数体中，先定义了整型变量 z；下面的 if 是条件选择语句，作用是如果 x>y，那么 z=x 否则 z=y，这样，变量 z 中存放的是 x 和 y 中的较大的值。

（3）函数体中的 return (z)是返回语句，表示将 z 的值作为函数值返回给主调函数 main()。

在 main 函数中，有以下几点要说明。

（4）两条赋值语句 a=3; b=4;写在了一行，这是允许的，因为 C 语言中以分号作为语句的结束，因此，一行可以书写多个语句。

（5）赋值语句 c=max(a,b);中，右边的表达式是函数调用，即调用函数 max，同时将实际参数 a 和 b 的值分别传递给 max 的形式参数 x 和 y，最后将函数 max 的返回值赋给变量 c。

（6）赋值语句 c=max(a,b);后面有形如/*...*/ 的形式，这是程序的注释部分，注释可以加在程序的任何位置，既可以单独占一行，也可以放在一条语句的后边，注释部分不参与程序的编译和运行。

在编写程序时，应养成这样的习惯，即在程序适当的位置加上注释以提高程序的可读性，尤其是编写的程序规模较大时更应如此。

这个程序的运行结果是：

```
max=4
```

由上所述，对于 C 语言源程序的组成，有如下几个要点：

（1）一个 C 语言源程序由一个或多个函数组成；

（2）每个源程序必须有而且也只能有一个 main()函数；

（3）除了 main()函数外，程序中还可以有若干个其他的函数。

一个完整的函数由函数说明和函数体两部分组成，如图 1-1 所示，其中的函数说明中包含 4 个部分，即函数类型、函数名、参数类型和参数名。

而在函数说明下面最外层的一对花括号中被包围的部分是函数体，它包括变量说明和执行语句两部分，其中变量说明部分用来定义变量的存储类型、数据类型和初值。

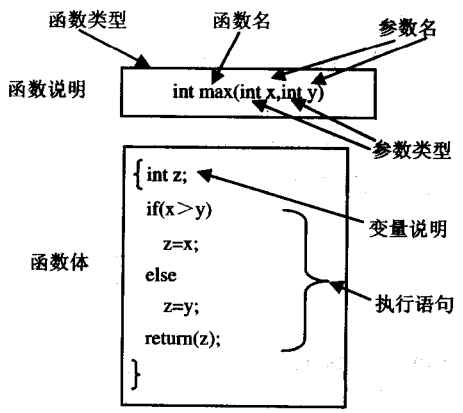

图 1-1　函数的组成部分

关于程序的执行，有如下几点：

（1）程序从 main()函数开始执行；

（2）其他函数通过调用的方式被执行；

（3）程序最后在 main()函数中结束。

3. C 程序的书写格式

从上面的介绍可知，C 程序的书写格式比较自由，主要表现为下面几点：

（1）一行可以写多个语句；

（2）一个语句可以分写在几行；

（3）可以在程序的任何位置用/*　　*/对程序或语句加注释。

1.1.3 程序的基本结构

程序结构是指函数中语句执行的流程结构，程序结构有顺序、选择和循环三种基本结构，这三种结构的共同特点是单进和单出，即一个入口和一个出口，由这些基本结构可以构成任何复杂的程序。

1. 顺序结构

顺序结构是最简单的程序结构，它按语句在函数中出现的次序依次执行。

2. 选择结构

选择结构又称为分支结构，是根据条件在程序的若干个流向中选择一个。

在 C 语言中，条件的成立与否是通过表达式的值来确定的，表达式可以是任何类型的，而程序根据表达式的值是 0 还是非 0 来判断表达式的成立与否。

构成分支的语句有 if 语句和 switch 语句。

3. 循环结构

循环结构是指程序在执行过程中，其中的某一段程序在满足条件时被重复执行若干次，被重复执行的程序段称为循环体。

在 C 语言中，有 3 种方法可构成循环结构，即 while 语句、do-while 语句和 for 语句，也可以用 goto 语句和 if 语句配合构成循环结构。

1.1.4 C 程序的运行

一个 C 语言的程序从编辑开始到得到运行结果要经过以下的步骤。

（1）编辑

编辑源程序是利用编辑程序建立一个新的源程序文件或修改已存在的源文件的过程，源程序文件以.C 作为扩展名。

（2）编译

编译是通过编译程序将指定的源程序文件进行编译形成目标文件.OBJ。

（3）连接

连接是将编译形成的目标文件.OBJ 和库函数、其他目录文件连接形成统一的可执行的二进制文件.EXE。

（4）执行

运行可执行的二进制文件.EXE，可以得到程序的运行结果。

我们平常使用的软件 Turboc 2.0 是一个集程序编辑、编译、连接和运行为一体的集成环境，详细使用方法见附录 4。

1.2 C语言中的常量、变量和标识符

1.2.1 常数

常数的值在程序运行过程中保持不变，按表示形式，常数有以下两种。

1. 直接常数

直接常数是直接按数据的类型书写，例如 123、4.54、'a'、"abc"等，它们分别表示 C 语言中的整型常数、实型常数、字符常数和字符串常数。

2. 符号常数

符号常数是指使用标识符来代表常量，就像数学上使用符号π代表圆周率一样，习惯上用大写字母表示符号常数。

在使用符号常数之前要用#define 进行定义，#define 是后面要介绍的编译预处理命令之一的宏替换，用#define 定义符号常数的格式如下：

格式：#define 符号常数　常量

【例 1-3】以下程序定义并使用符号常数。

```
#define PRICE 30
main()
{
    int num,total;
    num=10;
    total=num*PRICE;
    printf("total=%d",total);
}
```

程序中定义了符号常数 PRICE，在编译系统对程序编译之前将程序中所有的符号常数 PRICE 都用 30 来代替。

1.2.2 变量

变量是其值可以改变的量，C 语言中规定，在程序中要使用的变量应该先定义后使用，变量名的命名规则与下面要介绍的标识符的命名规则相同。

当一个变量被定义后，这个变量就与内存中的若干个连续的存储单元相对应，这样，对变量进行的操作实际上就是对该存储单元进行的操作。

例如，下面的说明语句：

```
int  a;
```

表示定义了一个变量，变量名为 a，类型是整型，它与内存中的两个连续单元对应。

一个完整的变量定义包括 3 个部分，分别是变量的存储类型、数据类型和对变量赋初值。本书中在前几章主要定义的是数据类型和初值，关于变量的存储类型，将在第 6 章中介绍。

变量赋值是指从向变量所代表的存储单元传送数据的操作，变量赋初值则是指在定义变量的同时给变量赋值，也称为变量的初始化。

例如：int　a=3;　相当于　int a; a=3;

1.2.3 标识符

C 语言中的标识符分为 3 类，分别为关键字、预定义标识符和用户标识符。

1. 关键字

这类标识符由C语言规定，在程序中表示固定的含义，例如，用来说明变量数据类型的标识符int、char、float等，用来说明变量存储类型的标识符static、auto、register等，以及控制语句中保留的if、for、goto等。

由于每个关键字都有特定的含义，因此，不能作为用户程序中的变量名和函数名等。

2. 预定义标识符

预定义标识符在C语言中也有特定的含义，如预处理命令的define、include，库函数的名称printf、scanf等。

在C语言语法中允许程序中用户标识符与这类标识符重名，但这样会使这些标识符失去系统规定的原意，因此，应避免将这些标识符另作它用。

3. 用户标识符

这一类是由用户根据程序需要自行定义的标识符，用来标识变量、符号常量、数组名和函数等，本书中以后将用户标识符简称为标识符。

用户在定义标识符时应遵守以下的规则：

（1）标识符中只能由字母、数字和下划线组成；

（2）标识符以字母或下划线开始；

（3）在C语言中标识符是区分大小写的，例如，abc、ABC和aBc在C语言中是不同的变量名；

（4）关于标识符的长度，不同系统中的规定不同，在使用时应了解所用系统的规定。

当用户标识符与关键字相同时，程序在编译时会出错；而当用户标识符与预定义标识符相同时，编译系统不会报错，此时预定义标识符失去原意，并可能出现运行错误。

1.3 C语言的基本数据类型

1.3.1 数据类型

在程序设计时针对不同的处理要使用不同的数据类型，区分不同数据类型的原因是：

（1）不同类型的数据在内存中占用的单元个数不同、表示方式也不同；

（2）不同类型的数据对应着不同的数值范围；

（3）一种数据类型对应着一组允许进行的操作。

C语言中提供了4类数据类型，分别是基本类型、构造类型、地址类型和空类型。

1. 基本类型

基本类型用来表示单个的数据，又可以分为以下几类：

（1）整型：表示整数，分为长整型、基本整型和短整型；

（2）实型：表示实数，按精度不同分为单精度（又称为浮点型）和双精度；

（3）字符型：表示单个字符。

2. 构造类型

构造类型又称为复合类型，是指由基本类型组合而成的数据类型，可以分为以下几类：

（1）数组：是指由相同类型的数据元素构成的类型；

（2）结构体类型：由不同类型数据成员构成，每个成员各占不同的内存单元；

（3）共用体类型：由不同类型数据成员构成，所有成员共占同一内存单元；

（4）枚举类型：由可能取的若干个常量值构成。

3. 地址类型

地址类型又称指针类型，可以直接表示内存中的地址，在第 7 章专门介绍。

4. 空类型

空类型由 void 进行定义，用来表示没有返回值的函数或没有确定指向的指针。

C 语言中的数据有常量和变量，分别属于以上的各种类型。

1.3.2 整型数据

1. 整型常量

整型常量书写时可以使用十进制、八进制或十六进制的形式，按类型可以有长整型、整型等区别，在表示方法上可以在数据的前面或后面加上特殊的标记。

（1）当常量以 0 开头时，表示该常量是八进制数，例如 0123。

（2）当常量以 0x 开头时，表示该常量是十六进制数，例如 0x123。

（3）常数以 L 结束时，表示该常量是长整型数，例如 123L。

（4）常数以 u 结束时，表示该常量是无符号整数，例如 0x123uL 表示十六进制的无符号的长整型数。

数据前面不加任何标记时，表示是十进制数。

2. 整型变量

整型变量可以分为基本型、短整型、长整型和无符号型，这些类型的标识符、所占内存字节数和表示的数值范围如表 1-1 所示。

表 1-1　　整型变量的分类

	类　型	标 识 符	占 字 节 数	数 值 范 围
有符号	整型(基本型)	[signed] int	2	-2^{15}～$2^{15}-1$
	短整型	[signed] short [int]	2	-2^{15}～$2^{15}-1$
	长整型	[signed] long [int]	4	-2^{31}～$2^{31}-1$
无符号	整型(基本型)	unsigned [int]	2	0～$2^{16}-1$
	短整型	unsigned short [int]	2	0～$2^{16}-1$
	长整型	unsigned long [int]	4	0～$2^{32}-1$

表 1-1 中，方括号中的名称可以省略，例如，[signed] int 可以简写为 int，即不写 signed 时默认为有符号的数，通常称 int 为基本整型。

3. 整数在内存中的存放形式

无符号的整数在内存中存放时，将整数对应的二进制数存放到相应字节个数的内存单元中。