

万水计算机培训教材系列



全国计算机等级考试 二级培训教程 (C 语言程序设计)

王正家 张业鹏 等 编著



中国水利水电出版社
www.waterpub.com.cn

万水计算机培训教材系列

全国计算机等级考试二级培训教程

(C语言程序设计)

王正家 张业鹏 等编著

中国水利水电出版社

内 容 提 要

本书根据计算机等级考试新大纲编写。讲述 C 语言的基本概念、基本语法以及如何利用 C 语言进行程序设计。本教程结合二级考试内容,通过大量实例进行讲解。每章有习题,并附有参考答案。

为配合二级考试中的上机考试,本书对上机考试的环境、过程及上机考试过程中出现问题时如何处理进行了说明。附录中对二级考试的大纲、样题、如何报考等考生较关心的问题作了说明。

本书对 C 语言的语法及编程的讲解较通俗易懂,对实际应用中的重点和难点进行了详尽的分析,是初学者学习 C 语言的一本好教材。

图书在版编目(CIP)数据

全国计算机等级考试二级培训教程(C语言程序设计)/王正家、张业鹏等编著. —北京:中国水利水电出版社, 2000.4

(万水计算机培训教材系列)

ISBN 7-5084-0310-X

I.全… II.①王…②张 III. C语言-程序设计-技术培训-教材 IV.TP312

中国版本图书馆CIP数据核字(2000)第04552号

书 名	全国计算机等级考试二级培训教程(C语言程序设计)
作 者	王正家 张业鹏 等编著
审 校	吴振彪
出版、发行	中国水利水电出版社(北京市三里河路6号 100044) 网址: www.waterpub.com.cn E-mail: sale@waterpub.com.cn 电话: (010)63202266(总机)、68331835(发行部)
经 售	全国各地新华书店
排 版	北京万水电子信息有限公司
印 刷	北京市天竺颖华印刷厂
规 格	787×1092毫米 16开本 15.5印张 343千字
版 次	2000年4月第一版 2000年4月北京第一次印刷
印 数	0001—5000册
定 价	20.00元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换

版权所有·侵权必究

本书编写人员

主 编 王正家 张业鹏

副主编 华中平、杨光友、吴斌方

主 审 吴振彪

编 委 朱 越 王志浩 王正家 杨光友

吴斌方 吴振彪 李莉华 丁建军

艾矫燕 华中平 张业鹏 杨 敏

苏旭武 雷顺加 王昌文 钟晓凌

杨光友 周国柱 曲文歧 雷飞跃

前 言

本书是在对二级考试大纲和 C 语言程序设计考试要求进行充分研究,以及对历年来考试试卷进行详尽分析的基础上编写而成。本书讲述全面、重点突出,对考试中的重点和难点进行了详细的讲解。

本书共分三大部分,第一部分为笔试部分,主要讲解 C 语言的基本语法。每章开头为导读,讲述本章相对应的考试大纲和学习中要注意的问题。接下去是正文,讲述相关语法,通过典型实例并结合历年考试试题进行讲解,然后是例题,主要是从历年试题中选择一些体现本章的重点和难点,在考试中较易出错的题目进行详尽地分析,深化对本章的理解。最后是习题,绝大部分习题以考试中题型形式出现,既体现了各章的知识点,又可熟悉考试的题型。

第二部分为上机考试部分,包括上机考试简介、考试的软件与硬件环境、考试内容分析,对考试过程的动态模拟,并对考试中可能出现的各种意外情况(如考试中出现死机等)给出了处理方法。

第三部分为附录。主要讲述等级考试的一些相关说明、二级考试 C 语言部分的大纲、样题、笔试试卷的题型分析以及二级考试答题卷等,这有利于读者对考试有一个全面的理解,以便在实际考试中尽快进入状态。本书的最后还附有各章习题的参考答案和如何进行二级考试报名等内容。

本书紧扣考试大纲,内容完整,概念清楚,通俗易懂,可读性、可操作性强,各章均配有大量与考题相似的例题、习题,方便广大应试者练习提高。

本书可以作为全国计算机等级考试二级——C 语言程序设计的培训教材,也可以作为各类大中专院校的 C 语言程序设计教材,同时也可以作为广大读者自学教材及 C 语言程序设计参考手册。

本书由王正家、张业鹏主编,并对全书初稿进行修改、补充、总撰;华中平、杨光友、吴斌方任副主编。全书由吴振彪主审。参加本书编写工作的还有:周国柱、丁建军、苏旭武、钟晓凌、曲文歧、雷飞跃、李莉华、朱越、雷顺加、王昌文、王志浩等。在本书的编写过程中得到了许多同志的支持,特别是吴振彪教授在百忙之中抽出时间对本书进行了审校,艾矫燕、杨敏等同志认真阅读了本书稿,提出了许多宝贵意见。另外,本书在编写过程中,广泛地参阅了有关论著,同时选用了一些全国计算机等级考试的试题,在这里一并致谢。

由于作者水平有限,书中不足和疏漏之处在所难免,恳请使用本书的广大专家和读者批评指正,以便再版时修改和补充。

编 者

目 录

前言

第 1 章 C 语言概述.....	1
1.1 导读	1
1.1.1 本章的考试要求	1
1.1.2 C 语言的发展简介及其特点	1
1.2 如何建立并运行第一个 C 程序	2
1.2.1 C 程序的建立和运行	2
1.2.2 实例分析	2
1.3 C 语言程序的结构	5
习题 1	7
第 2 章 数据类型、运算符与表达式.....	8
2.1 导读	8
2.1.1 本章的考试要求	8
2.1.2 本章的主要内容及学习重点	8
2.2 标识符	8
2.2.1 标识符	8
2.2.2 关键字	9
2.2.3 预定义标识符	9
2.2.4 用户标识符	9
2.3 整型数据	10
2.3.1 整型常量	10
2.3.2 整型变量	10
2.4 实型数据	11
2.4.1 实型常量	11
2.4.2 实型变量	12
2.5 字符型数据	12
2.5.1 字符型常量	12
2.5.2 字符变量	13
2.5.3 字符串常量	14
2.6 类型的自动转换和强制转换.....	14
2.7 运算符和表达式	15
2.7.1 算术运算符	16
2.7.2 自增、自减运算符（增一、减一运算符）	16

2.7.3 赋值运算符	17
2.7.4 关系运算符	18
2.7.5 逻辑运算符	18
2.7.6 其他运算符	19
2.8 优先级和结合性	20
2.9 应用举例	21
习题 2	22
第 3 章 基本语句	25
3.1 导读	25
3.1.1 本章的考试要求	25
3.1.2 本章主要内容及学习重点	25
3.2 程序的三种基本结构	26
3.3 数据输出	27
3.3.1 putchar 函数	27
3.3.2 printf 函数	28
3.4 数据输入	32
3.4.1 字符输入函数 getchar()	32
3.4.2 格式输入函数 scanf()	33
3.5 程序举例	35
习题 3	37
第 4 章 控制语句	41
4.1 导读	41
4.1.1 本章考试要求	41
4.1.2 本章的主要内容及学习重点	41
4.2 if 语句	42
4.3 switch 语句	46
4.4 循环语句	48
4.4.1 while 语句	48
4.4.2 do - while 语句	49
4.4.3 for 语句	50
4.5 break 语句和 continue 语句	51
4.5.1 break 语句	51
4.5.2 continue 语句	52
4.6 循环的嵌套	53
4.7 语句标号和 goto 语句	53

4.7.1 语句标号	53
4.7.2 goto 语句	54
4.8 应用举例	54
习题 4	58
第 5 章 数组	64
5.1 导读	64
5.1.1 本章考试要求	64
5.1.2 本章主要内容及学习重点	64
5.2 一维数组的定义和引用	64
5.2.1 一维数组的定义	64
5.2.2 一维数组的初始化	65
5.2.3 一维数组元素的引用	65
5.2.4 一维数组应用举例	65
5.3 二维数组的定义和引用	67
5.3.1 二维数组的定义	67
5.3.2 二维数组的初始化	67
5.3.3 二维数组元素的引用	68
5.3.4 二维数组应用举例	68
5.4 字符数组与字符串	69
5.4.1 字符数组的定义	69
5.4.2 字符数组的初始化	70
5.4.3 字符数组的引用	70
5.4.4 字符串和字符串结束标志	70
5.4.5 字符数组的输入输出	71
5.4.6 字符串处理函数	73
5.4.7 字符数组应用举例	75
习题 5	76
第 6 章 函数	80
6.1 导 读	80
6.1.1 本章考试要求	80
6.1.2 本章主要内容及学习重点	80
6.2 库函数的使用	80
6.2.1 调用标准库函数时要包含其相应的头文件	81
6.2.2 标准库函数的调用形式	81
6.3 函数的定义	81

6.4 函数的值与函数的类型	83
6.5 函数的参数及参数传递.....	84
6.6 函数的调用	84
6.6.1 函数调用的一般形式	84
6.6.2 函数调用时的注意事项	85
6.6.3 函数调用的方式	85
6.7 函数的嵌套调用	85
6.8 函数的递归调用	87
6.9 局部变量和全局变量	88
6.9.1 局部变量	88
6.9.2 全局变量	89
6.10 动态存储变量与静态存储变量.....	91
6.11 外部函数和内部函数	93
6.11.1 内部函数	93
6.11.2 外部函数	94
6.12 应用举例	94
习题 6	96
第 7 章 编译预处理	103
7.1 导读	103
7.2 宏定义	103
7.2.1 不带参数的宏定义	103
7.2.2 带参数的宏定义	104
7.3 “文件包含”处理	106
习题 7	107
第 8 章 指针	109
8.1 导读	109
8.2 指针的概念	109
8.3 指针与指针变量	110
8.3.1 指针变量的定义	110
8.3.2 指针运算符	110
8.3.3 指针变量作函数参数	111
8.4 数组的指针及指向数组的指针变量.....	115
8.4.1 指向数组的指针变量的定义及赋值	115
8.4.2 通过指针引用数组元素	115
8.4.3 用数组名作函数参数	117

8.4.4 用数组和指针作函数参数	119
8.4.5 指向多维数组的指针和指针变量	121
8.5 字符串的指针及其指针变量	127
8.5.1 字符串的实现方法	127
8.5.2 字符串指针作函数参数	128
8.5.3 动态存储分配	130
8.6 函数的指针及指向函数的指针变量	132
8.6.1 函数指针变量的定义与使用	132
8.6.2 用函数指针变量作函数参数	133
8.7 返回指针值的函数	134
8.8 指针数组和指向指针的指针	136
8.8.1 指针数组	136
8.8.2 指向指针的指针	138
8.9 指针数组作 MAIN 函数的参数	139
8.10 应用举例	140
习题 8	143
第 9 章 结构体与共用体	152
9.1 导读	152
9.2 结构体类型变量的定义、初始化与引用	152
9.2.1 结构体类型变量的定义	152
9.2.2 结构体变量的初始化	154
9.2.3 结构体变量的引用	154
9.2.4 结构体数组	154
9.2.5 指向结构体的指针	155
9.2.6 利用结构体作函数参数	157
9.3 用指针与结构体处理链表	158
9.3.1 链表介绍	158
9.3.2 链表建立	159
9.3.3 链表输出	160
9.3.4 链表删除	161
9.3.5 链表插入	162
9.4 共用体	165
9.4.1 共用体的概念	165
9.4.2 共用体变量的引用方式	165
9.5 枚举类型	167
9.6 用 typedef 定义类型	168

9.7 应用举例	168
习题 9	170
第 10 章 位运算	175
10.1 导读	175
10.1.1 考试要求	175
10.1.2 本章主要内容及学习重点	175
10.2 位运算符的含义及使用	175
10.2.1 与位有关的知识	175
10.2.2 按位运算符	177
10.3 应用举例	178
习题 10	179
第 11 章 文件	180
11.1 导读	180
11.1.1 本章的考试要求	180
11.1.2 C 语言文件概述及本章学习重点	180
11.2 文件类型指针	181
11.3 文件的打开与关闭、读写与定位	181
11.3.1 文件的打开 (fopen 函数)	181
11.3.2 文件的关闭 (fclose 函数)	182
11.3.3 文件的读写	182
11.4 文件的定位	188
11.5 应用举例	190
习题 11	190
第 12 章 上机考试指导	193
12.1 上机考试简介	193
12.2 上机考试环境	193
12.2.1 上机考试的软、硬件环境	193
12.2.2 考试用机的系统配置文件与自动批处理文件	194
12.2.3 汉字操作系统 UC DOS 中批处理 UP.BAT 的内容	194
12.3 上机考试时间及考试内容分析	194
12.3.1 DOS 常用命令操作题	195
12.3.2 程序修改调试运行	195
12.3.3 程序编制调试运行	198
12.4 上机考试动态模拟	201

12.5 上机考试中的相关问题.....	204
12.6 上机考试模拟习题及解答.....	206
12.6.1 模拟习题	206
12.6.2 参考答案	208
附 录	210
附录 1 全国计算机等级考试说明.....	210
附录 2 全国计算机等级考试二级考试大纲.....	212
附录 3 C 语言程序设计考试要求	213
附录 4 大纲中提供的等级考试样卷及答案.....	215
附录 5 试卷分析	226
附录 6 各章习题参考答案	227
附录 7 全国计算机等级考试答题卡.....	233
附录 8 全国计算机等级考试二级证书.....	234

第 1 章 C 语言概述

1.1 导 读

1.1.1 本章的考试要求

- (1) 程序的构成, main 函数和其他函数。
- (2) 头文件、数据说明、函数的开始和结束标志。
- (3) 源程序的书写格式。
- (4) C 语言的风格。

1.1.2 C 语言的发展简介及其特点

C 语言的产生与 UNIX 系统的发展关系密切。UNIX 系统是一个通用、复杂的计算机管理系统。在 UNIX 上实现了汇编语言后, UNIX 系统又以汇编语言来编写。汇编语言的主要优点是能充分体现计算机硬件指令级的特性,形成的代码质量较高,但它的可读性、可移植性以及描述问题的性能远不如当时所用的高级语言。于是人们试图找到一种既具有汇编语言特性又能克服其缺点的新语言来进行系统软件的设计。1963 年英国剑桥大学的 M.Richards 推出了 BCPL (Combined Programming Language) 语言,1970 年美国 Bell 实验室的 D.M.Rithie 和 K.Thompson 又作了进一步简化,设计出很简单而且很接近硬件的 B 语言,并用 B 语言编写了 UNIX 操作系统和大量的实用程序,但 B 语言功能有限,后来加进了构造类型等必要功能并作了规整,形成了 C 语言。现在,在原基础上又有所改进的 C 语言已成了世界上应用最广泛的几种计算机语言之一。

C 语言之所以能存在和发展,并具有生命力,是因为它有如下特点:

- (1) C 语言结构紧凑、简洁、灵活、使用方便。
- (2) 运算符丰富。灵活多样的运算符可以实现其他高级语言中难以实现的运算。
- (3) 数据结构丰富,具有现代化语言的各种数据结构。
- (4) 具有结构化的控制语句。用函数作为模块以实现程序的模块化。是一种结构化语言。
- (5) 语法限制不太严格,程序设计自由度大。例如,整型量与字符型数据以及逻辑型数据可以通用。因而 C 编程时,不要过分依赖 C 编译程序去查错。
- (6) C 语言允许直接访问物理地址,能进行位操作,能实现汇编语言的部分功能,可以直接对硬件进行操作。
- (7) 生成目标代码质量高,程序执行效率高。

(8) 用 C 语言写的程序可移植性比汇编语言好,基本上不用修改就可以用于各种型号的计算机和各种操作系统。

所以, C 既有高级语言的许多功能,又有低级语言的许多功能,既可以用来编写应用软件,也可以用来编写系统软件。

1.2 如何建立并运行第一个 C 程序

对计算机来说,它并不能直接识别由高级语言编写的程序,它只能接受和处理由 0 和 1 这种面向机器的代码构成的二进制数据(即机器语言)。计算机能按机器语言的指令顺序自动进行操作。我们把由高级语言编写的程序称为“源程序”,把二进制代码表示的程序称为“目标程序”。把源程序转换成机器能够接受的目标程序的工作由编译程序完成,编译程序能把用户按规定语法写出的语句翻译成二进制的机器指令。

C 程序从建立到生成可执行文件,一般都经过编辑源程序、编译、连接而成,对连接后生成的可执行文件可在相应的操作系统下运行。

由 C 语言构成的指令序列称为 C 源程序, C 源程序一般以 .c 为后缀。在 DOS 下 C 源程序经 C 编译程序编译之后生成一个后缀为 .obj 的二进制文件,又称目标文件。最后由连接程序(Link)把 .obj 文件和 C 语言提供的库函数连接起来生成后缀为 .exe 的可执行文件。exe 文件可在系统提示符下直接运行,键入该文件的名字即可。

1.2.1 C 程序的建立和运行

1. 编辑 C 源程序

在 UNIX 下可用屏幕编辑程序 vi 或文本编辑程序 ed。在 DOS 下可用任何文本编辑程序,如 EDIT。一般地, C 语言编译系统都提供一个集成环境,其中有源程序编辑窗口。

2. 编译

源程序经过编译后得到目标程序。如果在编译过程中发现源程序有语法错误,系统会给出“出错信息”,用户应对源程序进行编辑修改后再进行编译,直到编译通过为止。在不同的操作系统和编译系统下方法不一样,读者可阅读系统所附的说明。

3. 连接

将目标程序和库文件或其他目标程序连接,连成可执行的目标程序。

4. 运行

1.2.2 实例分析

下面,以 DOS 系统下最为常用的 Borland C 集成环境为例具体讲解(Turbo C 集成环境及用法与此类似,不必再讲)。

先介绍一个 C 程序。

【例 1-1】该程序运行后，在屏幕上输出“Hello,world”。

程序如下：

```
#include "stdio.h"

main ()
{
    printf("Hello, world\n");
    return 0;
}
```

操作步骤如下：

(1) 调入 Borland C (简称 BC) 程序，只需在相应目录下键入：

bc✓

则出现如图 1-1 所示界面。最上面的一栏为系统菜单，点击 File 菜单，会出现其相应的子菜单，见图 1-2，该菜单项主要进行文件输入输出，“New”新建文件，“Open”打开原先编辑过的文件等。在图中选 New，则出现源文件编辑窗口。用户可在其中编辑源文件，如图 1-3 所示。编辑完毕，可选择 File 下的 Save 存盘，比如存为文件“11.c”。



图 1-1

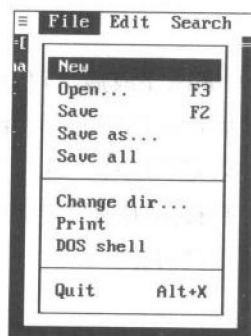


图 1-2

(2) 可选择 Compile 菜单下的 Compile 进行编译，选择 Link 进行连接。如图 1-3 所示。



图 1-3

编译时, 系统会在 Message 窗口给出一些警告信息和出错信息, 见图 1-4, 在该窗口中移动光标, 源程序中相应语句会在编辑窗口高亮度显示, 按回车键光标进入编辑窗口, 用户可如此反复修改, 直到编译通过。



图 1-4

(3) 编译完成后, 可选择 Run 菜单下的子菜单 Run 进行运行, 如图 1-5 所示。运行完毕, 光标自动返回源程序编辑窗口。也可不编译, 直接选择 Run 菜单下的子菜单 Run, 由系统自动完成编译、连接并运行。此时可按快捷键 Alt+F5 (即不需要从菜单中一级一级地选择, 直接敲击键盘便可执行相应命令) 看运行结果, 如果不对, 接着对源程序进行调试, 直至正确为止。

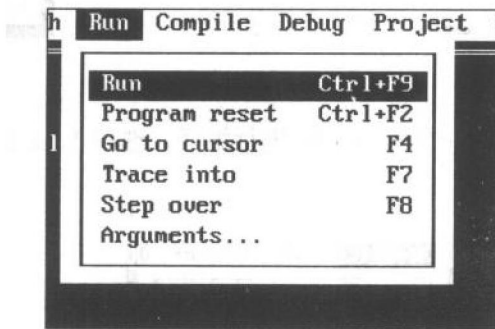


图 1-5

各菜单的快捷键在相应菜单的右侧, 例如运行命令 Run 的快捷键为 Ctrl+F9 (参见图 1-5)。

可按 Alt+X, 离开编程环境, 回到操作命令状态。

如果对系统有什么不懂的地方, 按 F1 键即可得到相关的提示。这里就不详细讲述了。

1.3 C语言程序的结构

这里，我们先来看一个简单的C程序。

【例 1-2】简单C程序实例。

```
#include "stdio.h"          /* 头文件 */
int min (int x, int y);      /* 函数说明 */
main ()                     /* 主函数 */
{
    int a, b, c;             /* 定义变量 a,b,c 为整型变量 */
    scanf ("%d,%d",&a, &b); /* 输入函数，输入变量 a,b 的值 */
    c = min (a, b);          /* 调用 min 函数，并将 min 函数返回的值赋给 c */
    printf ("min= %d\n",c); /* 打印函数，输出：min=c 的值 */
    return 0;               /* 返回 0 */
}

int min (x, y)              /* 定义 min 函数，函数值为整型，x, y 为形式参数 */
int x, y;                  /* 对形式参数作类型定义 */
{
    int z;                  /* 对函数内用到的变量 z 予以定义 */
    if (x < y) z = x;
    else z = y;
    return (z);             /* 将 z 的值返回，通过 min 函数带回调用处 */
}
```

本程序包括两个函数：主函数 `main` 和被调用的函数 `min`。`main` 函数由系统调用，一个程序总是从该函数开始执行。`min` 函数的作用是将 `x` 和 `y` 中较小者的值赋给变量 `z`。`return` 语句将 `z` 的值返回给主调函数 `main`。返回值是通过函数名 `min` 带回到 `main` 函数的调用处。`main` 函数中的 `scanf` 是“输入函数”的名字（`scanf` 和 `printf` 都是 C 语言提供的标准输入输出函数）。程序中 `scanf` 的作用是输入 `a` 和 `b` 的值。`&a` 和 `&b` 中的“&”的含义是“取地址”，此 `scanf` 函数的作用是：将两个数分别输入到变量 `a` 和 `b` 的地址所标志的单元中，即输入变量 `a` 和 `b`。其中，“%d”是输入输出的“格式字符串”，由“%”后跟格式字符构成，用来指定输入输出格式，“%d”表示“十进制整数类型”（其他如：“%s”表示字符串类型、“%c”表示字符类型等。关于这部分的内容详见第3章）。

“`c = min (a, b);`”语句调用 `min` 函数，在调用时将实际参数 `a` 和 `b` 的值分别送给 `min` 函数中的形式参数 `x` 和 `y`。经过执行 `min` 函数得到一个返回值，把这个值赋给变量 `c`。

“`printf ("min=%d\n",c)`”语句中双引号内的“`min=%d`”在输出时“`min=`”不是格式控制字符，照原样输出，“%d”是格式控制字符，表示输出表列中的变量 `c` 以十进制整数的形式输出。这样，程序运行情况如下：