

C 语言学习与实验指导

主 编 黄陈蓉

副主编 丁宇辰 温志萍

参 编 于 雷 夏官梦

河海大学出版社

前 言

全国计算机等级考试和江苏省计算机等级考试已进行多年,主要面向非计算机专业的人员,尤其是在校大学生。为了帮助更多的读者顺利通过考试,并掌握相应的操作技能,我们组织了具有多年 C 语言教学经验的老师,编写了本书。

“知识点讲析”结合考试大纲、教材,对历年试卷进行分析归纳,在此基础上对教材中考核的重点和难点进行了详细的讲解,涵盖了大纲中所有的考点。

“典型例题分析”选取具有代表性的经典例题,例题符合考试命题规律的特征,对题目的讲解深入、透彻、循序渐进。

“实验指导”重点训练读者的编程能力、调试程序的能力。

“同步练习”提供了大量的习题,辅助学习前面的理论,以练促学,学练结合。

本书由黄陈蓉主编并负责统稿。第四章、第六章和第十一章由黄陈蓉编写,第二章和第十章由丁宇辰编写,第五章和第九章由于雷编写,第一章、第三章、第七章和第十二章由温志萍编写,第八章和全部附录由夏官梦编写。

在本书的编写过程中,得到了南京工程学院朱承东老师和计算机教研室全体老师的大力支持和帮助,在此一并表示感谢。

由于编者水平有限,书中不足与疏漏之处在所难免,敬请读者批评指正。

编 者

2005 年 8 月

编 委 会

主 任 朱跃龙

副主任 马 民

编 委 (以姓氏笔划为序)

王月敏 王必友 代江滨 汤洪涛

周 松 秦 军 黄陈蓉

目 录

第一章 C 语言概述	1
【学习要求】.....	1
【知识点讲析】.....	1
一、C 语言的特点	1
二、C 语言程序的结构与书写规则	2
三、运行一个 C 语言程序的一般过程	4
【典型例题分析】.....	4
【实验指导】.....	5
一、目的和要求	5
二、实验内容	5
【同步练习】.....	7
第二章 数据类型、运算符与表达式	8
【学习要求】.....	8
【知识点讲析】.....	8
一、C 语言的数据类型	8
二、常量	8
三、变量	10
四、各类数值型数据间的混合运算	11
五、C 语言的运算符和表达式	12
【典型例题分析】	15
【实验指导】	17
一、实验目的	17
二、实验内容	17
【同步练习】	18
第三章 顺序结构程序设计	21
【学习要求】	21
【知识点讲析】	21
一、C 语句概述	21
二、C 程序的三种基本结构	22
三、字符数据的输入输出	22
四、格式化的输入输出	23
【典型例题分析】	28
【实验指导】	32

一、目的和要求	32
二、实验内容	32
【同步练习】	34
第四章 选择结构程序设计	38
【学习要求】	38
【知识点讲析】	38
一、关系运算符和关系表达式	38
二、逻辑运算符和逻辑表达式	38
三、if 语句	39
四、嵌套的 if 语句	41
五、switch 语句	41
六、条件运算符	42
【典型例题分析】	42
【实验指导】	46
一、目的和要求	46
二、实验内容	46
【同步练习】	50
第五章 循环控制	57
【学习要求】	57
【知识点讲析】	57
一、goto 语句	57
二、while 语句	57
三、do—while 语句	58
四、for 语句	58
五、break 和 continue 语句	59
六、嵌套的循环	59
【典型例题分析】	59
【实验指导】	65
一、目的和要求	65
二、实验内容	65
【同步练习】	70
第六章 数组	80
【学习要求】	80
【知识点讲析】	80
一、一维数组	80
二、二维数组	82
三、字符数组	84
四、常用的字符串处理函数	86

【典型例题分析】	87
【实验指导】	93
一、目的和要求	93
二、实验内容	94
【同步练习】	100
第七章 函数	107
【学习要求】	107
【知识点讲析】	107
一、函数的基本概念	107
二、定义函数的方法	108
三、调用函数的方法	110
四、内部函数和外部函数	113
五、函数参数的传递方式	113
六、全局变量和局部变量	115
七、动态存储与静态存储	116
【典型例题分析】	118
【实验指导】	124
一、目的和要求	124
二、实验内容	124
【同步练习】	128
第八章 编译预处理	138
【学习要求】	138
【知识点讲析】	138
一、概述	138
二、宏定义	138
三、文件包含	141
四、条件编译	142
五、本章总结	143
【典型例题分析】	143
【实验指导】	147
一、目的和要求	147
二、实验内容	147
【同步练习】	148
第九章 指针	153
【学习要求】	153
【知识点讲析】	153
一、指针和地址的概念	153
二、指针与基本类型变量	154

三、指针变量与一维数组	156
四、指针变量与二维(多维)数组	157
五、字符串的指针和指向字符串的指针变量	158
六、指针与函数	160
七、指针数组与指向指针的指针	163
八、指针数组作 main 函数的形参	164
【典型例题分析】	164
【实验指导】	170
一、目的和要求	170
二、实验内容	170
【同步练习】	176
第十章 结构体与共用体	189
【学习要求】	189
【知识点讲析】	189
一、结构体	189
二、结构体数组	193
三、结构体与指针	195
四、链表	199
五、共用体	205
六、枚举类型	206
七、用 typedef 定义类型	208
【典型例题分析】	209
【实验指导】	215
一、目的和要求	215
二、实验内容	215
【同步练习】	218
第十一章 位运算	233
【学习要求】	233
【知识点讲析】	233
一、数值在计算机中的表示	233
二、位运算	234
【典型例题分析】	236
【实验指导】	237
一、目的和要求	237
二、实验内容	237
【同步练习】	239
第十二章 文件	242
【学习要求】	242

【知识点讲析】	242
一、文件的基本概念	242
二、文件类型指针	242
三、文件的打开与关闭	243
四、文件的读写	243
五、文件的定位	244
【典型例题分析】	245
【实验指导】	248
一、目的和要求	248
二、实验内容	248
【同步练习】	251
附录一 C 语言中常用算法	255
一、排序算法	255
二、查找算法	265
三、线性代数方程组的求解——牛顿迭代法	271
四、矩阵方面的算法	272
五、经典瑟夫问题(递归的应用)	275
六、其他常用算法程序实例	277
附录二 关键字与标识符	281
一、关键字	281
二、标识符	281
附录三 Turbo C(V2.0)编译错误信息	282
一、致命错误英汉对照及处理方法	282
二、一般错误信息英汉对照及处理方法	282
【同步练习】参考答案	287
参考文献	294

第一章 C 语言概述

【学习要求】

- 了解 C 语言的特点
- 掌握 C 语言程序的构成
- 了解头文件、数据说明、函数的开始和结束标志
- 掌握源程序的书写格式
- 掌握 Turbo C 编辑、编译、连接、运行的基本操作

【知识点讲析】

一、C 语言的特点

(1) C 语言简洁、紧凑,使用方便、灵活

ANSI C 共有 32 个关键字,如表 1-1 所示:

表 1-1 ANSI C 关键字

auto	break	case	char	const	continue	default	do
double	else	enum	extern	float	for	goto	if
int	long	register	return	short	signed	sizeof	static
struct	switch	typedef	union	unsigned	void	volatile	while

Turbo C 扩充了 11 个关键字:

asm _cs _ds _es _ss cdecl far huge interrupt near pascal

9 种控制语句,程序书写自由。

注意:在 C 语言中,关键字和控制语句都是小写字母。在 C 语言中,必须严格区分大小写。

(2) 运算符丰富

C 语言中共有 34 种运算符,括号、赋值、逗号等都作为运算符处理。

(3) 数据结构类型丰富

C 语言的数据类型有整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等。

(4) 具有结构化的控制语句

(5) 语法限制不太严格,程序设计自由度大

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

(7) 生成目标代码质量高,程序执行效率高

(8) 与汇编语言比,用 C 语言书写的程序可移植性好

二、C 语言程序的结构与书写规则

1. C 语言程序的总体结构

一个完整的 C 语言程序,是由一个 main()函数(又称主函数)和若干个其他函数结合而成的,或仅由一个 main()函数构成。

例 1 仅由 main()函数构成的 C 语言程序。

```
main()
{printf("This is a C program.\n");
}
```

程序运行结果:

This is a C program.

例 2 由 main()函数和 1 个其他函数 max()构成的 C 语言程序。

```
int max(int x, int y)
{return(x>y ? x : y);}

main()
{int num1,num2;
 printf("Input two numbers:\n");
 scanf("%d", &num1);
 scanf("%d", &num2);
 printf("max=%d\n", max(num1, num2));
}
```

程序运行结果:

```
Input two numbers:
6 ✓
9 ✓
max=9
```

从以上两个例子,可以看出:

(1) 函数是 C 语言程序的基本单位

main()函数的作用,相当于其他高级语言中的主程序;其他函数的作用,相当于子程序。

(2) C 语言程序总是从 main()函数开始执行

一个 C 语言程序,总是从 main()函数开始执行,而不论其在程序中的位置。当主函数执行完毕时,亦即程序执行完毕。

2. 函数的一般结构

任何函数(包括主函数 main())都是由函数说明和函数体两部分组成。其一般结构如下:

[函数类型]	函数名(函数参数表)	函数说明部分
	{说明语句部分;	}函数体部分
	执行语句部分;	
	}	

说明:

(1) 函数说明部分

由函数类型(可缺省)、函数名和函数参数表三部分组成,其中函数参数表的格式为:

数据类型 形参 1, 数据类型 形参 2,……

如例 2 中,max()函数,其函数说明各部分如图 1-1 所示。

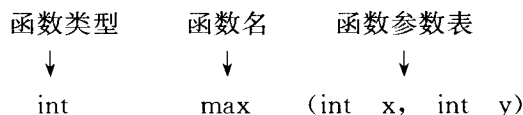


图 1-1 函数说明部分结构图

(2) 函数体部分

在函数说明部分的下面、大括号(必须配对使用)内的部分。

函数体一般由说明语句和可执行语句两部分构成:

- 说明语句

说明语句部分由变量定义、自定义类型定义、自定义函数说明、外部变量说明等组成。

- 可执行语句

一般由若干条可供执行的语句构成。

注意:C 语言本身没有输入输出语句,输入输出操作是由库函数来完成的。

如例 2 中,main()函数,其函数体各部分如图 1-2 所示。

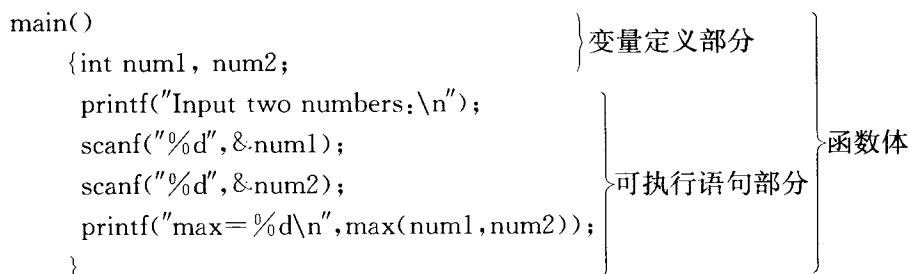


图 1-2 函数体结构示意图

注意:函数体中的变量定义语句,必须在使用该变量之前。

如下面程序中,变量定义语句“int max;”的位置是非法的:

```
main()
{
    int x=3,y=6; /* 变量定义语句:定义 2 个整型变量 x,y,将 3 赋给 x,6 赋给 y */
    max=x>y? x:y;
    int max; /* 变量定义语句:出现在“max=x>y? x:y;”之后,非法! */
    printf("max=%d\n",max);
}
```

3. 源程序书写格式

- 1) 所有语句都必须以分号“;”结束,函数的最后一个语句也不例外。
- 2) 程序行的书写格式自由,既允许 1 行内写几条语句,也允许 1 条语句分写在几行上。
- 3) 允许使用注释。C 语言的注释格式为: /* */。

注意：“/ *”和“* /”必须成对使用，且“/”和“*”、以及“*”和“/”之间不能有空格，否则都出错；注释的位置，可以单占 1 行，也可以跟在语句的后面；如果 1 行写不下，可另起 1 行继续写。

三、运行一个 C 语言程序的一般过程

Turbo C(简称 TC)是一个集源程序编辑、编译、连接、运行与调试于一体、用菜单驱动的集成软件环境。

运行一个 C 语言程序的一般过程是：

- 1) 启动 TC, 进入 TC 集成环境。
- 2) 编辑(或修改)源程序。
- 3) 编译。如果编译成功, 则可进行下一步操作; 否则, 返回 2) 修改源程序, 再重新编译, 直至编译成功。
- 4) 连接。如果连接成功, 则可进行下一步操作; 否则, 根据系统的错误提示, 进行相应修改, 再重新编译、连接, 直至连接成功。
- 5) 运行。通过观察程序运行结果, 验证程序的正确性。如果出现逻辑错误, 则必须返回 2) 修改源程序, 再重新编译、连接和运行, 直至程序正确。
- 6) 退出 TC 集成环境, 结束本次程序运行。

【典型例题分析】

例 1 C 语言程序的执行总是从_____。

- | | |
|-----------------|----------------|
| A. main 函数的入口开始 | B. 编译预处理命令开始 |
| C. 源程序的第一个函数开始 | D. 源程序的第一行语句开始 |

答案:A。

解析:一个 C 语言程序,总是从 main() 函数开始执行,当主函数执行完毕时,亦即程序执行完毕。

例 2 下列说法中正确的是_____。

- A. C 程序书写时,不区分大小写字母
- B. C 程序书写时,一行只能写一个语句
- C. C 程序书写时,一个语句可分成几行书写
- D. C 程序书写时,每行必须有行号

答案:C。

解析:C 语言严格区分大小写字母,如“A1”和“a1”被认为是两个不同的标识符,C 程序的书写非常灵活,既可以一行多句,又可以一句多行,而每行不加行号。

例 3 以下叙述正确的是_____。

- A. 在对一个 C 程序进行编译的过程中,可发现注释中的拼写错误
- B. 在 C 程序中,main 函数必须位于程序的最前面
- C. C 语言本身没有输入输出语句
- D. C 程序的每行中只能写一条语句

答案:C。

解析:注释,仅仅是进行语句说明,并不执行,程序进行编译时,不会对其进行任何检查;

C 语言程序总是从 main() 函数开始执行, 而不论其在程序中的位置; C 程序的书写非常灵活, 既可以一行多句, 又可以一句多行; C 语言本身没有输入输出语句, 它是由函数来完成输入输出操作的。

例 4 在 C 语言的源程序中, 主函数的位置_____。

- A. 必须在第一个函数的后面
- B. 必须在源程序的最后
- C. 任意
- D. 必须在源程序的开头

答案:C。

解析:一个 C 语言程序, 总是从 main() 函数开始执行, 而不论其在程序中的位置。因而主函数的位置可以任意。

例 5 C 语言规定, 必须用_____ 作为主函数的名。

- A. stdio
- B. include
- C. main
- D. function

答案:C。

解析:在 C 语言中, 主函数名必须为 main。

例 6 在 C 语言中, 每个语句和数据的定义是用_____ 结束的。

- A. 括号
- B. 句号
- C. 分号
- D. 逗号

答案:C。

解析:在 C 语言中规定, 每个语句必须用分号(;) 结束。

例 7 一个函数一般由两部分组成, 它们分别是_____ 和_____ 。

答案:函数说明; 函数体。

解析:每个函数的定义分为函数说明部分(又称为函数首部)和函数体 2 个部分。

例 8 注释的开始和结束标志分别为_____ 和_____ 。

答案:/* ; */。

解析:C 语言中可以利用“/*”和“*/”对 C 语言程序中的任何部分做注释。“/*”和“*/”必须成对使用, 且“/”和“*”以及“*”和“/”之间不能有空格。

【实验指导】

一、目的和要求

熟悉 Turbo C 运行环境、创建 C 程序文件的过程。

二、实验内容

1. 在 C 编辑环境下, 输入下列程序。然后将其以 p1.c 为文件名保存到 D 盘的 cc 目录中(若 D 盘中没有 cc 子目录, 则需要自己建立)

1) 启动 C 编辑环境(假设 TC 的安装目录为 C:\tc)。

开始→程序→MS-DOS 方式, 在 MS-DOS 方式下键入:

cd\tc ✓

2) 在如下的提示符下, 键入 tc。

C:\tc>tc ✓

则启动 C 语言编辑环境。

3) 按 Esc 键后, 光标落在编辑区中, 键入下面程序:

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

```
main()
{int k,kl;
k=32767;
kl=k+1;
printf("%d,%d\n",k,kl);}
```

4) 程序输入完成后,利用 Alt+F 将 File 菜单的子菜单显示在屏幕上,选择 Save,此时显示一个对话框,在对话框中输入:d:\cc\pl.c 则程序被保存。

5) 利用 Alt+X 退出 C 语言编辑环境。

6) 再次启动 C 语言编辑环境。

7) 打开上面保存的程序 pl.c。

同时按下 Alt+F 将 File 菜单的子菜单显示在屏幕上,选择 Load,此时显示一个对话框,在对话框中输入:d:\cc\pl.c,则此时 pl.c 程序被显示在程序的编辑窗口中。

8) 将上面的程序改名为 ppl.c 保存到 D 盘的 cc 目录中(D:\cc)。

同时按下 Alt+F 将 File 菜单的子菜单显示在屏幕上,选择 Write to,此时显示一个重命名对话框,在对话框中输入:d:\cc\ppl.c,则此时 ppl.c 程序被保存到 D 盘 cc 子目录中,此时编辑窗口右上方的文件名为:ppl.c。

9) 编译 ppl.c 程序。

同时按下 Alt+C,则屏幕上显示编译子菜单,当光带落到第一个子菜单项 Compile to OBJ 上时按回车键,则程序编译开始,编译完成后显示编译结果,如图 1-3 所示。

```
Main file: PP1.C
Compiling: EDITOR - PP1.C

      Total      File
Lines compiled: 223      223
Warnings: 1          1
Errors: 0             0

Available memory: 268K
Warnings : Press any key
```

图 1-3 ppl.c 程序编译结果

10) 运行 ppl.c 程序。

浏览完编译结果后,若程序没有错误,可按键盘中的任意一个键,则回到编辑窗口,此时同时按下 Alt+R,则显示编译子菜单,当光带落到第一个子菜单项 Run 上时按回车键,则链接、运行开始。运行完成后,利用 Run 子菜单中的 User screen 命令观察运行结果。

2. 按上面“输入、调试、编译、链接、执行”程序的步骤,完成下面程序的输入、编译和执行

```
int max(int x, int y)
{return( x>y ? x : y );}

main()
{int num1,num2;
printf("Input two numbers:\n");
scanf("%d", &num1);
scanf("%d", &num2);
printf("max=%d\n", max(num1, num2));
}
```

分析上面程序的运行结果,写出运行结果,并给出运行结果的说明。

【同步练习】

一、选择题

1. 一个 C 程序的执行是从_____。
 - A. 本程序的 main 函数开始,到 main 函数结束
 - B. 本程序文件的第一个函数开始,到本程序文件的最后一个函数结束
 - C. 本程序文件的第一个函数开始,到本程序 main 函数结束
 - D. 本程序的 main 函数开始,到本程序文件的最后一个函数结束
2. 以下叙述不正确的是_____。
 - A. 一个 C 源程序必须包含一个 main 函数
 - B. 一个 C 源程序可由一个或多个函数组成
 - C. C 程序的基本组成单位是函数
 - D. 在 C 程序中,注释说明只能位于一条语句的后面
3. 一个 C 语言程序是由_____。
 - A. 一个主程序和若干个子程序组成
 - B. 函数组成
 - C. 若干过程组成
 - D. 若干子程序组成

二、填空题

1. 函数体用_____符号开始,用_____符号结束。
2. 一个 C 语言程序至少应包含一个_____。

第二章 数据类型、运算符与表达式

【学习要求】

- 掌握 C 语言的数据类型及其在内存中的表示方法
- 掌握运算符及其优先级和结合性
- 掌握常量和变量
- 掌握 C 语言的表达式
- 掌握变量赋初值运算符及其表达式

【知识点讲析】

一、C 语言的数据类型

在程序设计中经常使用各种变量,这些变量必须预先加以说明,即先说明,后引用。对变量的说明应该包括三个方面:数据类型、存储类型和作用域。其中最重要的方面是数据类型。

所谓数据类型就是按被说明量的性质、表示形式、占据存储空间的多少、构造特点来划分的一种数据结构。C 语言提供的数据结构是以数据类型的形式出现的。C 语言的数据类型如图 2-1 所示:

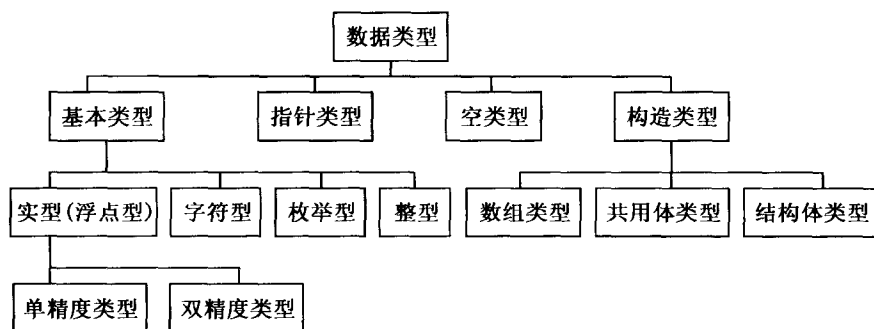


图 2-1 C 语言的数据类型

二、常量

常量是指在程序运行过程中其值不能被改变的量。常量可分为整型常量、实型常量、字符常量、字符串常量和符号常量。

在 C 语言源程序中,常量是可以不经说明而直接引用的。

1. 整型常量

整型常量就是整常数。整型常量有三种形式:十进制整型常量、八进制整型常量和十六进制整型常量。

十进制整型常量,即十进制整数。下面的数据均是正确的十进制整数。

214, 0, -123, 78L 等,基本数字范围为 0~9(其中 L 为长整型)。

但下面的数据是不正确的十进制数:45D,E78,45D2D等。(含有非法字符)

八进制整型常量,即八进制整数。以0开头的数据即是八进制整数,下面的数据均是正确的八进制整数。

012,038L,-058,0369等,基本数字范围为0~7。

但下面的数据是不正确的八进制数:512,0E78等。(没有以0开头,包含了非八进制数码)

十六进制整型常量,即十六进制整数。以0x开头的数据即是十六进制整数,下面的数据均是正确的十六进制整数。

0x12,0x38L,-0x58,0x36a,0xAFBa等,基本数字范围为0~9,A~F或a~f。

但下面的数据是不正确的十六进制数:512,0h78等。(没有以0x开头,包含了非十六进制数码)

从上面的分析可以看出,整型常量的区分是以前缀为根据的,因此书写时应注意不要把前缀写错,以免造成不正确的结果。

2. 实型常量

实数在C语言中也称浮点数,有两种表示形式来表示一个实数。这两种表示形式分别为十进制数形式和指数形式。

十进制数形式:由数字和小数点组成,小数点在这种形式中是必需的。如下的数据均是合法的实数:0.0,1.0,4.55,144,154.987,-156.568。

指数形式:由十进制数字加阶码标志“e”和“E”以及阶码(只能为整数)组成。注意:阶码标志“e”和“E”之前必须有数字,之后的指数必须为整数。如下的数据均是合法的实数:2.3E3,3.5E-5,0.4e-6,-2.84e-3。如下的数据均不是合法的实数:E2(E前无数字),3.5e3.5(e后为小数),2.84-e3(负号位置不对),0.4e(e后无数字),e(e前无数字,e后无整数)。

3. 字符常量

字符常量是用单引号引起来的一个字符。如'x','y','+', '-', '8','%'等都是合法的字符常量。C语言中字符常量:①只能用单引号括起来。②只能是单个字符,字符可以是字符集的任意字符。

但对于一些非显示字符或特殊形式的字符,C语言还允许用一种特殊形式的字符常量,就是以“\”开头的字符序列,称为转义字符。如表2-1所示。

表 2-1 转义字符

字 符 形 式	功 能
\n	换行
\t	横向跳格(即跳到下一个输出区)
\v	竖向跳格
\b	退格
\r	回车
\f	走纸换页
\\	反斜杠字符“\”
\'	单引号(撇号)字符
\ddd	1到3位8进制数所代表的字符
\xhh	1到2位16进制数所代表的字符