

万水全国计算机等级考试教材系列

全国计算机等级考试要点指导、真题解析与强化训练——二级 C 语言

匡 松 主编

缪春池 吕峻闽 副主编

中国水利水电出版社

内 容 提 要

本书根据 2001 年教育部考试中心对全国计算机等级考试二级(C 语言)调整后的内容编写而成,满足 2002 年 4 月及以后的考试要求,为广大考生顺利通过计算机等级考试提供最为有效的过快捷径。

本书分为“要点指导”、“真题解析”、“强化训练”、“上机题”四大部分。真题解析部分对近年来的二级(C 语言)全部考试真题进行了详细地分析和解答,能让读者(考生)做到举一反三,从而更好地理解和掌握等级考试的内容、范围及难度;强化训练部分提供了大量针对性很强的模拟试题(附有答案),这些试题都经过精心设计和锤炼,采用标准题型,突出了考点、重点及难点;上机题部分提供了多套上机题,并给出了这些上机操作题的指导及操作提示。

图书在版编目(CIP)数据

全国计算机等级考试要点指导、真题解析与强化训练——二级 C 语言/匡松等编著. —北京:中国水利水电出版社, 2002

(万水全国计算机等级考试教材系列)

ISBN 7-5084-1180-3

I. 全… II. 匡… III. ①电子计算机—水平考试—自学参考资料②C 语言—水平考试—自学参考资料 IV. TP3

中国版本图书馆 CIP 数据核字(2002)第 062869 号

书 名	全国计算机等级考试要点指导、真题解析与强化训练——二级 C 语言
主 编	匡 松
副 主 编	缪春池 吕峻闽
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: mchannel@public3.bta.net.cn (万水) sale@waterpub.com.cn 电话: (010) 68359286 (万水)、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 17 印张 382 千字
版 次	2002 年 8 月第一版 2002 年 8 月北京第一次印刷
印 数	0001—5000 册
定 价	22.00 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

丛书序

为了适应计算机应用技术的发展和我国计算机应用水平的实际情况，力求考试能体现考生对当前先进技术的掌握程度，教育部考试中心于 2001 年 11 月对全国计算机等级考试的科目和内容进行了调整，并将在 2002 年下半年启用新版计算机等级考试合格证书。

为了适应最新等级考试的科目和内容的调整，帮助广大考生顺利通过计算机等级考试，提供最为有效的过关捷径，我们将陆续推出以下几种图书：

-  一级 Windows
-  二级 QBASIC
-  二级 C 语言
-  二级 FoxBASE+
-  二级 Visual FoxPro
-  二级 Visual Basic
-  三级 PC 技术
-  三级信息管理技术
-  三级网络技术
-  三级数据库技术

每本书中分为“要点指导”、“真题解析”、“强化训练”、“上机题”四大部分。

要点指导：按照大纲所要求的内容，对重要的知识点进行了总结、归纳和指导。

真题分析：对近几年来全部考试真题进行了详细地分析和解答。在每道真题的后面指出了该题的考试时间（例如，“2001 年上”表示 2000 年上半年的考试题）。这些详细而透彻的解答能让读者（考生）举一反三，茅塞顿开，从而更好地理解 and 掌握等级考试的内容、范围及难度。

强化训练：提供了大量针对性很强的模拟试题（附有答案）。模拟试题经过精心设计和锤炼，采用标准题型，突出考点、重点、难点，应试导向准确。

上机题：提供了多套上机题，并给出了这些上机操作题的指导及操作提示。

本丛书的特点及目的是不仅让读者“看”和“理解”真题，而且同时进行实战性地“练”，做到看、理解、做题实战的全面训练，使读者在短期内获得最好的效果，从而系统地复习、巩固和强化所学的计算机知识，加深对基本概念的理解，熟悉等级考试的形式和题型，掌握要点，克服难点，熟练掌握答题方法及技巧，适应考试氛围，为顺利通过等级考试打下坚实基础，树立成功信心。

本丛书主要编写人员有：匡松、梁庆龙、李朔枫、缪春池、吕峻闽、郭黎明、董事尔、

帅青红、付强、邓放、刘虹、罗文山、郑团结。

由于编者水平有限，书中难免有错误和不足之处，敬请读者批评指正。

编者

2002年6月于西南财经大学

目 录

丛书序

第 1 章 C 语言的结构与数据类型及其运算	1
1.1 要点指导	1
1.1.1 程序的构成、main 函数和其他函数	1
1.1.2 头文件、数据说明、函数的开始和结束标志	2
1.1.3 源程序的书写格式	3
1.1.4 C 语言的风格	3
1.1.5 C 的数据类型及其定义方法	3
1.1.6 C 运算符的种类及运算优先级	6
1.1.7 C 表达式类型和求值规则	6
1.2 真题解析	8
1.2.1 选择题	8
1.2.2 填空题	15
1.3 强化训练	16
1.3.1 选择题	16
1.3.2 填空题	31
1.3.3 答案	36
1.4 上机题	37
1.4.1 改错	37
1.4.2 编程	41
第 2 章 基本语句与结构程序设计	46
2.1 要点指导	46
2.1.1 基本语句	46
2.1.2 选择结构程序设计	48
2.1.3 循环结构程序设计	50
2.2 真题解析	52
2.2.1 选择题	52
2.2.2 填空题	73
2.3 强化训练	79
2.3.1 选择题	79
2.3.2 填空题	90

2.3.3	答案	96
2.4	上机题	97
2.4.1	改错	97
2.4.2	编程	101
第 3 章	数组与函数	109
3.1	要点指导	109
3.1.1	数组的定义与使用	109
3.1.2	函数的使用	111
3.2	真题解析	115
3.2.1	选择题	115
3.2.2	填空题	128
3.3	强化训练	135
3.3.1	选择题	135
3.3.2	填空题	148
3.3.3	答案	155
3.4	上机题	156
3.4.1	改错	156
3.4.2	编程	161
第 4 章	指针、结构、联合与位运算	171
4.1	要点指导	171
4.1.1	指针	171
4.1.2	结构体与共用体	178
4.1.3	位运算	182
4.2	真题解析	183
4.2.1	选择题	183
4.2.2	填空题	203
4.3	强化训练	209
4.3.1	选择题	209
4.3.2	填空题	222
4.3.3	答案	226
4.4	上机题	227
4.4.1	改错	227
4.4.2	编程	231
第 5 章	编译预处理与文件操作	239
5.1	要点指导	239
5.1.1	编译预处理	239

5.1.2	文件操作	241
5.2	真题解析	244
5.2.1	选择题	244
5.2.2	填空题	246
5.3	强化训练	249
5.3.1	选择题	249
5.3.2	填空题	253
5.3.3	答案	258
5.4	上机题	258
5.4.1	改错	258
5.4.2	编程	260

第 1 章 C 语言的结构与数据类型及其运算

本章大纲要求

(1) C 语言的结构

- 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。

(2) 数据类型及其运算

- 1) C 语言的数据类型(基本类型、构造类型、指针类型、空类型)及其定义方法。
- 2) C 语言运算符的种类、运算优先级和结合性。
- 3) 不同类型数据间的转换与运算。
- 4) C 语言表达式类型(赋值表达式、算术表达式、关系表达式、逻辑表达式、条件表达式、逗号表达式)和求值规则。

1.1 要点指导

1.1.1 程序的构成、`main` 函数和其他函数

1. 程序的构成

C 语言的源程序是由函数构成的, 至少包含一个 `main` 函数, 也可以包含 `main` 函数和其他函数。函数是 C 程序的基本单位, 可以由用户根据自己的需要进行定义, 还可以是系统提供的库函数。

由用户编写的 `main` 函数和若干其他函数从结构和功能上都是相对独立的, 只是由于相互之间的调用关系而联系在一起。C 程序的执行从 `main` 函数开始, 再通过 `main` 函数对其他函数的调用以及其他函数相互之间的调用, 然后层层返回调用点, 最后回到调用的起点 `main` 函数而结束。

2. 函数的组成

一个函数由说明部分和函数体组成。说明部分包括函数名、函数的返回类型、函数性质、函数的形式参数名称和类型, 函数名后的圆括号内是形式参数表, 如果有多个参数则用逗号隔开, 也可以没有参数, 则括号为空; 函数体用一对大括号括起, 包括变量定义和由语句组成的执行部分, 由用户根据函数的功能编写。

3. `main` 函数

`main` 函数是每个 C 语言源程序的必然组成部分, 一般来说是无参函数。无论 `main` 函

数在文件的哪个位置, C 程序的执行总是从 main 函数开始, 最后回到 main 函数结束。main 函数可以去调用其他函数, 但不能被其他函数所调用。

4. 其他函数

除 main 函数之外的其他函数, 有的由用户编写, 有的是系统库函数。可以被 main 函数调用, 但不能去调用 main 函数, 其他函数之间也可以相互调用。标准 C 提供了一百多个库函数, Turbo C 和 MS C 则提供了三百多个库函数。

1.1.2 头文件、数据说明、函数的开始和结束标志

1. 头文件

头文件是指经常用在源文件的开头、被预编译命令 `#include` 包含进来的文件。一般以“.h”作扩展名。

包含头文件的命令格式是:

```
#include "头文件名"
```

或

```
#include <头文件名>
```

头文件的内容一般包含一些公用的或常用的宏定义、构造型数据类型的定义以及全局变量的定义等。当源文件需要这些定义时, 不必重新定义, 只需把所对应的头文件包含进来, 相当于把头文件的全部内容插到当前源文件的开头, 合二为一后再进行统一编译。

使用头文件的好处主要在于能够避免一些重复的定义, 大的系统要使用的公共变量可以集中在一个头文件中, 需要时只需简单地用 `#include` 命令包含即可; 另外如果要修改符号常量的值, 只需修改头文件里的宏定义, 而不必去修改使用该常量的每一处文件或函数。但要注意的是, 被包含的头文件一旦被修改, 则包含此文件的所有文件都要重新进行编译。

注意: 在调用系统库函数时, 也要在源文件的开头把库函数对应的头文件包含进来, 这样库函数才能被正常调用。一般属于同一类型的库函数对应一个头文件。例如, 数学类函数的头文件是 `math.h`, 字符类函数的头文件是 `ctype.h`, 字符串类函数的头文件是 `string.h`, 输入输出类函数的头文件是 `stdio.h`。

2. 数据的说明

C 语言的数据分为常量和变量, 常数可以直接使用, 符号常量需要进行宏定义, 而变量则必须在使用前先进行名称和类型的定义。在函数头中, 要对形式参数变量的名称和类型进行定义; 在函数体中, 要对在该函数中即将使用的局部变量进行类型和名称的定义; 在函数之外, 还要对全局变量进行定义。每个变量在定义后, 在编译时就能根据其类型分配相应的存储单元。

符号常量的定义格式为:

```
#define 符号常量名 值
```

变量的定义格式为:

```
类型名 变量名;
```

或

```
类型名 变量名=初值;
```

相同类型的变量由逗号隔开统一定义。当然变量的性质还需要根据变量定义的位置等情况进一步确定。

3. 函数的开始和结束标志

函数的开始和结束标志就是指包括函数体的最外层的一对大括号。函数体的执行从左大括号后的第一条语句开始，到右大括号前的最后一条语句结束。在函数体中，由于复合语句的存在，可能出现若干个或若干层大括号，要分清楚对应关系，不要漏掉。最外层的大括号才是函数的开始和结束标志。

1.1.3 源程序的书写格式

C语言的书写格式比较自由，一行可以写几条语句，一个语句也可以写在多行上。但每条语句和数据定义后必须有一个分号。

C语言的注释符号由/*和*/组成，中间的字符即为注释信息。注释信息可以增强程序的可读性。

一般的语句和函数、数据定义用小写字母表示，符号常量名用大写字母表示。

1.1.4 C语言的风格

C语言的特点有以下6点：

- 语言简洁、使用方便：共只有 32 个关键字，9 种控制语句，语句简炼，书写自由。
- 运算符丰富：共有 34 种运算符，包括算术运算符、关系运算符、逻辑运算符、位运算符、赋值运算符、指针运算符、条件运算符、逗号运算符、下标运算符、类型转换运算符等多种类型。
- 数据结构多样：有整型、实型、字符型、枚举类型等基本类型，有数组、结构体、共用体等构造类型以及指针类型。能够实现复杂的数据结构。
- 控制语句使用方便：有两路分支、多路分支和循环结构几种控制语句，便于结构化模块的实现和控制。
- 可以直接访问地址、进行位操作，从而能对硬件进行操作，因此 C 语言既具有高级语言编写简单方便的优点，又有低级语言与硬件结合紧密的优点。
- 移植性好：由 C 语言编写的程序基本不用太多的修改就可以用于不同型号的计算机。

1.1.5 C的数据类型及其定义方法

C语言的数据类型分为基本类型、构造类型、指针类型、空类型。

C语言的数据分为常量和变量两大类。

常量是指在程序运行过程中其值不能改变的量，主要有整型常量、实型常量、字符型常量和字符串常量几种类型。常量又可分为常数和符号常量两类，常数一般从表达字符的组成和形式即可判别其类型；符号常量是用#define 命令定义一个标识符代表某一个常数，

则凡在程序中出现的该标识符都会在运行前被对应的常数所代替而参与运算，习惯上符号常量名由大写字母组成。

变量是指在运行过程中其值可以改变的量。应在使用前为其取一个名字（一般小写），再为其定义数据类型，以后就会在程序编译过程中为其分配适合其类型的存储单元，用于存放其值。因此变量在使用前必须先作定义。定义变量的数据类型的基本格式为：

类型说明符 变量名；

如果在定义的同时就要为变量赋初值，格式为：

类型说明符 变量名 = 初值；

下面主要介绍一下各种数据类型的特点和定义方式。

1. 基本类型

数据的基本类型包含整型、字符型（char）、实型、枚举类型（enum）4 种。

➤ 整型

整型变量又分为基本型（int）、短整型（short int）、长整型（long int）和无符号型（unsigned）。具体类型不同，变量所分配的存储字节数就不同，因此对应的数据表示范围各异。而不同机型上相同类型的变量分配的存储字节数也可能不同。要注意不同类型变量的占用字节数及表示数的范围的关系。就常用的 PC 机而言，基本型、无符号的短整型和无符号的基本型、短整型变量均用两个字节（16 位）存储，但短整型变量的范围是-32768~32767，而无符号短整型变量的范围是 0~65535。长整型变量用 4 个字节存储，数据的范围就大大地拓宽了，能够表达更大的数值。在选择变量的类型时，要根据估计的变量使用范围合理选择，做到量体裁衣。例如，如果一个变量的数值变化范围超过-32768~32767，就必须把它定义为长整型变量。对整型常量，如果要表示为长整型常量，则必须在其数值后面加字母 l 或 L；常量没有无符号型。

注意：整型常量还有 3 种进制的表示形式，十进制是常用的方式，不需特别说明，八进制数要在数字后加字母 O，十六进制数要在数字后加字母 H。

➤ 实数类型

实型变量又分为单精度型（float）和双精度型（double）。单精度型变量在内存中占 4 个字节，提供 7 位有效数字，双精度型变量在内存中占 8 个字节，提供 15~16 位有效数字，所能表示数的范围和精度都大大拓宽和提高。在选择实型变量的类型时，也要根据估计的变量使用范围和要求的精度进行合理选择，做到量体裁衣。实型的常量有两种表达方式：十进制数形式由数字和小数点组成，比较常用，注意绝对值小于 1 的小数的小数点前的 0 可以省略，如 0.57 也能表示为 .57；指数形式为“基数 E 幂”或“基数 e 幂”，注意基数可以是整数或小数，但幂必须是整数。如 1.5e3 和 15E2 都代表 1500。

➤ 字符类型

字符类型常量以单引号作分界符，除了一般可见的字符外，一些不可见的控制符号或特殊符号是以“\”开头的转义字符来表达。例如：“\n”表示换行符，“\r”表示回车符，“\\”表示反斜线本身，“\'”表示单引号（与作为分界符的单引号相区分）。字符类型变量用 char 来定义。特别要注意字符数据在内存中的存储方式：字符数据在内存中是用字符对应的 ASCII 码存

储,其存储形式实质是采用整数形式,由于ASCII码集合中有256个字符,因此用0~255这个范围内的256个整数分别与每个字符对应。因此字符型数据和整型数据在一定范围(0~255)内,形式上可以相互引用。表现在这两种类型的变量能够以对方形式常量来赋值、以对方的形式来输出,还可以同时出现在算术表达式中参与运算。例如:

```
int      n = 'a' ;                /*相当于 n = 97*/
char     c = 65 ;                 /*相当于 c = 'A'*/
printf("n = %d , n = %c",n , n ) ; /*输出为: n = 97 , n = a*/
printf("c = %d , c = %c",c , c ) ; /*c = 65 , n = A*/
printf("同一字母的小写字符与大写字母的 ASCII 码之差为: %d", n - c ) :
                                           /*差为: 32*/
printf("字母 A 后面的第 5 个字母为: %c", 'A' + 5 ) ;
                                           /*输出的字母为 F*/
```

虽然字符型数据和整型数据在一定范围内可以通用,但其外在表现形式并不影响各自的数据类型,上例中n本质上还是整型变量,c本质上还是字符型变量。

另外,根据字符数据的存储特性,任一字符常量都能用转义字符表示,形式为: '\ddd'表示ASCII码为1位~3位8进制数的对应字符, '\xhh'表示ASCII码为1位~2位16进制数的对应字符。例如 '\101'和 '\x41'都代表字符'A'。

与字符常量相区别,字符串常量用双引号作分界符,且引号内部可有多个字符,不要把字符常量和字符串常量混淆起来。字符常量与字符型变量相对应,而字符串常量则与字符数组相对应。特别注意不要把一个字符串常量赋值给字符变量。另外,C语言会在字符串的末尾自动加上一个字符串结束标志'\0',作为判断字符串是否结束的依据。也就是说常量串在内存中占据的字节数总是比它实际字符的数量多一个字节用于存放'\0',但'\0'并非字符串的实质组成部分,而只是作为一个控制符存在。因此字符串"k"实际包含两个字符'k'和'\0',也不能将其赋值给字符型变量。

➤ 枚举类型

枚举类型就是把变量的值全部列举出来,变量只能在这个范围内取值。枚举类型的定义格式是:

```
enum      枚举类型名      {枚举元素表}枚举变量表;
```

如果变量定义在枚举类型定义之后,格式是:

```
enum      枚举类型名      枚举变量表;
```

枚举类型名的命名规则同变量名的命名规则,不要和已有的类型名重名。

枚举元素表中的元素之间用逗号相隔。枚举元素一般按符号常量处理,称为枚举常量,每个枚举常量用一个标识符表示,并且有惟一的值与之相对应。一般情况下,由C语言在编译时按枚举元素的顺序依次赋值为0、1、2、…。但也可以在定义枚举类型时,在枚举元素表中对枚举常量的值进行定义,形式是:枚举常量名=整数。被重新定义的枚举常量与新值对应,排在被重新定义的常量之后的枚举常量,则以该新值为基准,依次递增1,直到最后一个元素或又遇到被重新赋值的元素。

2. 构造类型

构造类型是指把多个基本类型相同或不同的变量有机组合起来形成的数据类型。包括

数组类型、结构体 (struct) 和共用体 (union) 类型。这 3 种类型将在后面的章节中介绍。

1.1.6 C 运算符的种类及运算优先级

1. 运算符的种类

C 语言的运算符分为以下几类:

- 算术运算符: + - * / % ++ --
- 关系运算符: > >= < <= == !=
- 逻辑运算符: ! && ||
- 位运算符: << >> ~ ^ &
- 与类型有关的运算符: 指针运算符 “*”、取地址运算符 “&”、下标运算符 “[]”、强制类型转换运算符 “(类型)”, 分量运算符 “.” 和 “->”。
- 其他运算符: 赋值运算符 “=”, 逗号运算符 “,”, 条件运算符 “?:”, 求字节数运算符 “sizeof”。

2. 运算优先级

C 语言常用的几类运算符的优先级关系是: 逻辑非 “!” > 算术运算符 > 关系运算符 > 逻辑与 “&&” > 逻辑或 “||” > 赋值运算符 > 逗号运算符。注意在混合运算时分清运算的先后。如果运算顺序与优先顺序不符, 考虑用加圆括号来改变运算次序。

3. 不同类型数据间的转换与运算

整型、字符型、实型数据可以混合运算, 如果表达式内的运算量的类型不同, 不同类型的数据要先转换成同一类型, 再进行运算。转换方向由低到高为: char/short、int、unsigned、long、float、double。当然也可以使用强制类型转换符来转变类型, 但变量本身的类型并未发生改变。例如: (double)n, (int)(a+b)。

1.1.7 C 表达式类型和求值规则

与运算类型相对应, 包含某种运算符的表达式则称为某种类型的表达式, 因而 C 语言主要有算术表达式、关系表达式、逻辑表达式、赋值表达式、条件表达式、逗号表达式等几种主要表达式。下面简述每种表达式的运算规则和注意事项。

1. 算术表达式

算术运算符的优先级别是先乘除后加减, 结合方向为从左至右结合。其中除法运算符 “/”, 作用是取两个数相除的整数部分, 舍去小数。对相除的结果为负数的情况, 多数是向 0 方向取整, 例如 $-7/2 = -3$ 。如果参与运算的两个数中有实数, 结果也是实数。

求余运算符 “%” 的作用是取两数相除的余数部分。注意 “%” 两边的运算量必须为整型数据。

自加运算符 “++” 和自减运算符 “--” 的作用分别是让变量的值加 1 或减 1。这两个运算符是单目运算符, 只针对一个变量运算, 不能对表达式进行运算。其功能相当于加、减表达式的简化, 单自加和自减运算符与运算量的位置关系不同, 有时对结果的影响就有所不同。自加和自减运算符与变量的位置关系有两种: 一是运算符在前、运算量在后, 则

先对变量进行自加或自减的运算，再让变量带着改变后的值参与所在位置的其他运算；二是运算量在前、运算符在后，则先让变量带着原值参与所在位置的其他运算，再对变量进行自加或自减的运算。

2. 关系表达式

关系运算符有 6 种，<、<=、>、>= 的优先级高于 == 和 !=。主要注意在判别两个表达式的值是否相等时，一定要用关系运算符 ==（两个等号组成），不要误写成赋值运算符 =，否则，意思发生根本变化。

关系表达式的结果是两个逻辑值：真和假。关系表达式成立则值为真，不成立则值为假。C 语言中以 1 或一切非零的常数代表“真”，以 0 代表“假”。

注意：一般注意不要出现两个关系运算符出现在一个关系表达式中的情况，而要一分为二，用一个逻辑运算符连接分开的两个关系表达式。例如，数学表达式 $1 < a < 10$ 要表达成 $a > 1 \ \&\& \ a < 10$ ，这样才正确，否则直接按数学表达式的形式表达，将会和原意不符。

3. 逻辑表达式

逻辑运算符有 3 种，运算优先级从高到低的顺序是：逻辑非“!”、逻辑与“&&”、逻辑或“||”。其运算规则如表 1-1 所示。

表 1-1 逻辑运算符的运算规则

a	b	!a	a && b	a b
1	1	0	1	1
1	0	0	0	1
0	1	1	0	1
0	0	1	0	0

逻辑表达式的结果是两个逻辑值：真和假。1、非 0 的整数实数、非'\0'的字符型数据都可以代表“真”，0 和'\0'代表“假”。

4. 赋值表达式

赋值运算符“=”连接的是左边的变量和右边的表达式，把表达式的值赋值给左边的变量。当两边的数据类型不一致时，再赋值时会自动进行类型的转换。

C 语言把赋值号和其他的运算符结合起来，组成复合赋值运算符，使表达式的形式更加简洁，提高编译的效率。形式如下：

<变量名> <其他运算符>= <表达式>;

其功能相当于：

<变量名> = <变量名> <其他运算符> <表达式>;

一共有 10 种复合赋值表达式：+=、-=、*=、/=、%=、<<=、>>=、&=、^=、|=。

例如， $a *= b + c$; 转换成一般的赋值表达式为 $a = a * (b + c)$;

赋值表达式的值就等于左边变量的值，一个赋值表达式中可以包含多个赋值表达式，不加括号的情况下，赋值运算符按“从右至左”的结合顺序运算。一个赋值表达式中还可以包含多个复合赋值表达式，运算时要注意其中某个变量的值被改变后，要带着新值参与

余下的运算。

5. 条件表达式

条件表达式相当于两路分支选择语句的一种简化形式, 包含条件运算符“?:”。即当 if 语句的两路分支均为一个赋值语句时, 可以用简单的条件表达式(条件表达式?表达式 1: 表达式 2) 来代替 if 语句进行赋值。例如:

```
if ( a > b )  
    c = a;  
else  
    c = b;
```

可改写为:

```
c = ( a > b ? a : b ) ;
```

6. 逗号表达式

逗号表达式用逗号把若干个表达式连接起来, 一般形式为:

表达式 1, 表达式 2, ..., 表达式 n

逗号表达式按表达式从左至右的排列顺序进行运算, 然后以最后那个表达式 n 的值作为整个逗号表达式的值。逗号表达式一般用于 for 型循环语句的循环头中。

1.2 真题解析

1.2.1 选择题

真题 1-1 C 语言提供的合法关键字是_____。(1998 上)

- A) swith B) cher C) Case D) default

【答案】 D)

【解析】 选项 A) 应为多分支语句中的 switch, 选项 B) 应为定义字符类型的 char, 选项 C) 应为多分支语句中的分支关键字 case (小写)。因此, 选项 A)、B)、C) 都有书写上的错误, 只有选项 D)——多分支语句中默认分支关键字正确。

真题 1-2 在 C 语言中, 合法的字符常量是_____。(1998 上)

- A) '084' B) '\x43' C) 'ab' D) "\0"

【答案】 B)

【解析】 选项 A) 错在表示转义字符的 3 位八进制数中不能出现 8, 选项 C) 错在字符串要用双引号括起, 选项 D) 错在单个字符应用单引号括起。选项 B) 表示用两位 16 进制数代表转义字符。

真题 1-3 若已定义 x 和 y 为 double 类型, 则表达式: $x=1, y=x+3/2$ 的值是_____。(1998 上)

- A) 1 B) 2 C) 2.0 D) 2.5

【答案】 C)

【解析】 $3/2$ 的结果要舍去小数部分而得 1, 但由于 x、y 均为 double 类型, 整型数

据向实型数据转换, 因此, 最后结果为 2.0。

真题 1-4 设 a 为整型变量, 不能正确表达数学关系: $10 < a < 15$ 的 C 语言表达式是_____。(1998 上)

A) $10 < a < 15$

B) $a == 11 \parallel a == 12 \parallel a == 13 \parallel a == 14$

C) $a > 10 \&\& a < 15$

D) $!(a \leq 10) \&\& !(a \geq 15)$

【答案】 A)

【解析】 选项 A) 中 $10 < a$ 的结果是逻辑值 0 或 1, 再与 15 比较大小没有意义, 因此错误, 一定要注意当两个关系运算符出现在一个关系表达式中, 是否与本意相符, 多数情况下要一分为二, 再用逻辑运算符连接。选项 C) 和 D) 显然是同一意思。

真题 1-5 若 t 为 double 类型, 表达式 $t=1, t+5, t++$ 的值是_____。(1998 上)

A) 1

B) 6.0

C) 2.0

D) 1.0

【答案】 D)

【解析】 逗号表达式的值为最后一个子表达式的值, 第一个表达式给 t 赋值为 1.0 后, 第二个表达式并没有改变其值, 而第三个表达式由于自加在后, 因此结果仍为 1.0。

真题 1-6 若有定义和语句:

```
char s[10]; s="abcd"; printf("%s\n",s);
```

则结果是(以下□代表空格)_____。(1998 上)

A) 输出 abcd

B) 输出 a

C) 输出 abcd□□□□

D) 编译不通过

【答案】 D)

【解析】 注意不能在数组定义点之外把字符串通过赋值运算符对数组直接赋值, 而要用库函数 `strcpy(s, "abcd")`。

真题 1-7 若有以下定义和语句:

```
int u=010,v=0x10,w=10;
```

```
printf("%d,%d,%d\n",u,v,w);
```

则输出结果是_____。(1998 上)

A) 8,16,10

B) 10,10,10

C) 8,8,10

D) 8,10,10

【答案】 A)

【解析】 u 被八进制数 10 赋值, 等于十进制数 8; v 被十六进制数 10 赋值, 等于十进制数 16; w 直接被十进制数 10 赋值; 因此分别输出十进制数 8, 16, 10。

真题 1-8 若有以下定义和语句:

```
char c1='b',c2='e';
```

```
printf("%d,%c\n",c2-c1,c2-'a'+'A');
```

则输出结果是_____。(1998 上)

A) 2,M

B) 3,E

C) 2,E

D) 输出项与对应的格式控制不一致, 输出结果不确定

【答案】 B)

【解析】 因为 e 是 b 后的第 3 个小写字母, $c2-c1$ 的结果是 3; $c2-'a'+'A'$ 相当于 $4+'A'$,

即大写字母 A 后的第 4 个字符 E。

真题 1-9 下面程序的输出是_____。(1998 上)

```
main()
{ unsigned a=32768;
  printf("a=%d\n",a);
}
```

- A) a=32768 B) a=32767 C) a=-32767 D) a=-1

【答案】 C)

【解析】 无符号数 32768 的最高位为 1，以整型数输出则是负数。

真题 1-10 若有定义：int x,y;char a,b,c;，并有以下输入数据（此处<CR>代表换行符，□代表空格）：

1□2<CR>

A□B□C<CR>

则能给 x 赋整数 1，给 y 赋整数 2，给 a 赋字符 A，给 b 赋字符 B，给 c 赋字符 C 的正确程序段是_____。(1998 上)

- A) scanf("x=%d y=%d",&x,&y);a=getchar();b=getchar();c=getchar();
B) scanf("%d %d",&x,&y);a=getchar();b=getchar();c=getchar();
C) scanf("%d%d%c%c%c",&x,&y,&a,&b,&c);
D) scanf("%d%d%c%c%c%c%c%c",&x,&y,&a,&a,&b,&b,&c,&c);

【答案】 D)

【解析】 进行连续整数的输入时，两整数间有空格或其他分隔号相隔，即可分别赋值给两个不同变量；而给连续几个字符变量赋值，不能用分隔符，否则会把分隔符当成输入字符赋值给对应变量的。在选项 D) 中，第一个%c 对应 1□2 后的回车符，先赋给 a，第二个%c 对应字符 A，再赋给 a，以此类推，最终将 B 赋给 b、C 赋给 c。而其他 3 个答案都会出现对应错误。

真题 1-11 在以下一组运算符中，优先级最高的运算符是_____。(1998 上)

- A) <= B) = C) % D) &&

【答案】 C)

【解析】 在运算符的优先关系中，算术运算符优先于关系运算符，关系运算符又优先于逻辑运算符。选项 A) 是关系运算符，选项 B) 是赋值运算符，选项 C) 是算术运算符，选项 D) 是逻辑运算符。

真题 1-12 若有以下程序片段：

```
char str[]="ab\n\012\\\"";
printf("%dd",strlen(str));
```

上面程序片的输出结果是_____。(1998 上)

- A) 3 B) 4 C) 6 D) 12

【答案】 C)

【解析】 ab 后的 4 个字符都是由反斜线引出的转义字符，\n 是换行符，\012 也是换