

高职高专现代信息技术系列教材

C 语言程序设计教程习题解答与实训

宗大华 王晓宇 编

人民邮电出版社

图书在版编目 (CIP) 数据

C 语言程序设计教程习题解答与实训/宗大华, 王晓宇编著.

—北京: 人民邮电出版社, 2005.1

(高职高专现代信息技术系列教材)

ISBN 7-115-12929-0

I. C... II. ①宗...②王... III. C 语言—程序设计—高等学校: 技术学校—教学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字 (2004) 第 138214 号

内 容 提 要

本书是人民邮电出版社出版的高职高专现代信息技术系列教材中《C 语言程序设计教程》的辅导用书。主要内容分成三个大的部分: 第一部分给出《C 语言程序设计教程》所有习题的详细解答; 第二部分讲述 Turbo C 2.0 的集成开发环境, 以及调试 C 语言程序的基本手段; 第三部分是上机实训。

本书是对《C 语言程序设计教程》教材的补充和完善, 旨在通过对程序调试手段的学习和真正动手上机实践, 使学生对 C 语言的认识迈上一个新的台阶。

高职高专现代信息技术系列教材 C 语言程序设计教程习题解答与实训

◆ 编 宗大华 王晓宇

责任编辑 潘春燕

执行编辑 韩学义

◆ 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

北京汉魂图文设计有限公司制作

北京顺义振华印刷厂印刷

新华书店总店北京发行所经销

◆ 开本: 787×1092 1/16

印张: 11

字数: 257 千字 2004 年 9 月第 1 版

印数: 1—0 000 册 2004 年 9 月北京第 1 次印刷

ISBN 7-115-12929-0 /TP·4352

定价: 15.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

高职高专现代信息技术系列教材

编 委 会 名 单

主 编 高 林

执行主编 张强华

委 员 (以姓氏笔画为序)

吕新平 林全新 郭力平 程时兴

编者的话

本书是教材《C 语言程序设计教程》的配套辅导用书，是对教材内容的补充和完善。

本书分成三个大的部分。

第一部分的内容是习题解答。它共分 8 章，每章都由该章的内容要点和习题解答两个部分组成。为了加深理解，在给出习题解答的同时，对于某些题目，还做了适当的分析，以帮助学习者理解为什么会是这样的一个答案。

第二部分的内容是 C 语言程序的上机操作过程。这部分由第 9 章（Turbo C 2.0 的集成开发环境）和第 10 章（C 语言程序的调试）组成。在教材第 1 章中，虽然对 Turbo C 2.0 的集成开发环境做了介绍，但那是很初步的，第 9 章是对其内容的补充和完善。第 10 章的内容是全新的，它讲述了 C 语言程序中的错误类型、初学编程时应该注意的地方，着重介绍了 C 语言向用户提供的调试程序的各种方法。这些方法，对于查找、挖掘程序中隐藏的错误，是非常有用的。

第三部分的内容是实训。本部分针对教材的内容，总共安排了 8 个上机实训。每个实训给出实训目的和多个小题目。希望读者能够根据实训的题目要求，在自己计算机的环境下去完成它们。通过上机实训，通过自己动手，相信会受益匪浅！

参加本书编写的人员还有陈新、陈吉人、宗涛、蒋玮等。由于作者水平有限，难免存在错误和不足，恳请读者批评、指正。

编 者

2004 年 9 月

目 录

第一部分 习题解答

第 1 章 概述	2
1.1 内容要点	2
1.2 习题解答	3
第 2 章 数据类型、运算符与表达式	6
2.1 内容要点	6
2.2 习题解答	9
第 3 章 C 语言程序设计的三种基本结构	13
3.1 内容要点	13
3.2 习题解答	15
第 4 章 数组	25
4.1 内容要点	25
4.2 习题解答	26
第 5 章 指针	34
5.1 内容要点	34
5.2 习题解答	36
第 6 章 函数	43
6.1 内容要点	43
6.2 习题解答	45
第 7 章 用户自定义的数据类型	53
7.1 内容要点	53
7.2 习题解答	58
第 8 章 C 的文件操作函数	67
8.1 内容要点	67
8.2 习题解答	71

第二部分 C 语言程序的上机操作过程

第 9 章 Turbo C 2.0 的集成开发环境	78
9.1 进入菜单	78
9.1.1 Turbo C 的主窗口	78
9.1.2 进入菜单的方法	79
9.1.3 主要菜单项的介绍	79
9.1.4 常用功能键	92
9.2 C 语言程序的上机操作过程	93
9.2.1 对源程序文件的编辑	93
9.2.2 编译、连接和装配	94
第 10 章 C 语言程序的调试	96
10.1 C 语言程序中的错误	96
10.1.1 程序设计中的错误类型	96
10.1.2 初学编程时应该注意的问题	99
10.2 C 语言程序调试方法简介	110
10.2.1 在程序中添加调试语句	110
10.2.2 利用编译时输出的出错信息	116
10.2.3 跟踪/单步式的程序执行方式	127
10.2.4 用 Toggle breakpoint 命令设置断点	136

第三部分 实 训

实训 1：概述	144
实训目的	144
实训内容	144
实训 2：数据类型、运算符与表达式	145
实训目的	145
实训内容	145
实训 3：程序设计的三种基本结构	147
实训目的	147
实训内容	147
实训 4：数组	148
实训目的	148
实训内容	148
实训 5：指针	151
实训目的	151
实训内容	151
实训 6：函数	153
实训目的	153

实训内容	153
实训 7：用户自定义的数据类型	155
实训目的	155
实训内容	155
实训 8：文件操作函数	156
实训目的	156
实训内容	156
附录 Turbo C 编译的主要错误一览	158

第一部分

习 题 解 答

本部分列出了《C 语言程序设计教程》一书中每章的内容要点及所附习题的答案。需要说明的是：第一，对于某些题目，在给出答案的同时，还针对题目做了适当的分析，以帮助学习者理解为什么会是这样的一个答案；第二，关于“编程”题目的答案，每个人的思路不尽相同，因此它不可能是惟一的。

第 1 章 概述

1.1 内 容 要 点

自然语言是人与人之间用来表达思想、交流情感的工具。高级程序设计语言 C，也是一种表达思想、交流情感的工具，只不过是发生在人和计算机之间而已。因此，用 C 语言来编写程序，就如同自然语言一般，也必须符合它的语法规则。

1. 计算机程序设计语言的分类

(1) 机器语言

计算机自带的指令系统是机器语言，计算机硬件能够直接识别和接受它。因此，用机器语言编写的程序，具有质量高、执行速度快、占用存储空间少等优点。但是，它却缺乏直观性，难学、难记、难检查以及难修改。

(2) 汇编语言

汇编语言是一种用助忆符和符号地址组成的指令，基本上与机器指令一一对应。它较为直观、好记，使用起来比较方便。但由于计算机不能直接识别和接受它，必须要经过汇编程序的翻译，才能在计算机上运行。

(3) 高级语言

高级语言接近于人们使用的自然语言和数学语言，因此，人们用它来编写计算机程序，比起使用机器语言和汇编语言，显然要得心应手得多。但用高级语言编写的程序，计算机也不可能直接识别与接受，也必须要有个“翻译”，把用高级语言编写的程序翻译成机器指令的程序，然后才能让计算机去执行。

2. 高级语言程序的两种翻译方式

(1) “编译”方式

事先通过编译程序，将用高级语言编写的源程序整个地翻译成用机器指令表示的机器语言程序（即目标程序），然后执行该目标程序。

(2) “解释”方式

通过解释程序，将用高级语言编写的源程序逐句翻译，翻译一句就执行一句。

3. C 语言程序的特点

用 C 语言编写计算机程序时，有如下的主要特点。

(1) C 语言程序都是由一个个函数组成的，函数是 C 语言程序的基本单位。

(2) 每一个 C 语言程序，都有一个、且只有一个名为 `main` 的主函数，整个程序从它开

始执行。至于 main 函数在程序中所放的位置，与它作为程序开始执行的地位没有什么关系。也就是说，main 函数可以安排在整个程序的最前面、中间，或后面。

(3) C 语言程序中的每一个语句，都以分号“;”作为自己的结束。也就是说，在 C 语言中，分号是一个语句的结束标志。

(4) 在 C 语言程序中，可以用/* */形成注释，以对程序中的所需部分作出说明。“/*”是注释的开始符，“*/”是注释的结束符，必须配对使用。注释部分是不会被翻译成目标指令执行的。

4. C 语言的基本语法

(1) 字符集

允许出现在 C 语言源程序中的所有字符的总体，称为 C 语言的“字符集”。它由数字、英文字母、图形符号以及转义字符四部分组成。

(2) 保留字

在 C 语言中，具有特定含义的、用于构成语句成分或作为存储类型和数据类型说明的那些单词，被统称为“保留字”，有时也称为“关键字”。比如“if”、“switch”、“static”等，都是 C 语言的保留字。C 语言的保留字只能用小写。

(3) 标识符的构成规则

在 C 语言中，用户通过取不同的名字，来区分程序中出现的常量、变量、函数和数组等。组成名字的字符序列，称为“标识符”。书写一个标识符，必须符合下面所列的语法规则：

- ① 标识符只能以字母或下划线开头；
- ② 在第一个符号的后面，可以跟随字母、数字和下划线；
- ③ 标识符中区分字母的大、小写；
- ④ 标识符的长度一般限制为 8 个字符；
- ⑤ C 语言的保留字不能作为标识符使用。

1.2 习题解答

一、填空

1. 机器语言即是指计算机本身自带的指令系统。
2. 将汇编语言编写的程序翻译成机器语言程序的过程称为汇编，完成这个翻译工作的程序称为汇编程序。
3. 在 C 语言程序中，写“\110”和写“\x68”，分别代表字母H和字母h。
4. 在用 New 命令创建新的 C 源程序时，文件名默认为NONAME.C。
5. C 语言程序都是从名为main的函数开始执行的。

二、选择

1. 下面给出的命令中，(C) 不能保存源程序对应的“.OBJ”文件。

A. Make EXE File

B. Run

C. Save

D. Compile to OBJ

分析：Make EXE File 是一次完成编译和连接的命令；在没有编译、连接的前提下，直接使用 Run 命令，就会先完成编译，再进行连接，最后运行；Compile to OBJ 是文件的编译命令。所以，这三个命令都会保存由源程序产生出的“.OBJ”文件。而 Save 命令用于编辑完源程序后，对源程序的保存，即保存“.C”文件。因此，本题的答案应该是 C。

2. 下面给出的编辑命令中，(B) 是用来定义块首标记的。

A. Ctrl+KK

B. Ctrl+KB

C. Ctrl+KV

D. Ctrl+KH

3. 下面给出的编辑命令中，(A) 是用来定义块尾标记的。

A. Ctrl+KK

B. Ctrl+KB

C. Ctrl+KV

D. Ctrl+KH

4. 以下的 (C) 是不正确的转义字符。

A. \

B. \

C. \81

D. \0

5. 转义字符“\x65”对应的字母是 (C)。

A. A

B. a

C. e

D. E

三、是非判断 (√, ×)

1. Turbo C 中，只有命令 New 才能创建新的源程序文件。(×)

2. 使用主菜单 File 下面的 Save 命令，可以保存“.EXE”文件。(×)

3. Turbo C 是“编译”型的编译程序。(√)

4. 由 Turbo C 产生的所有程序（源程序、目标程序和可执行程序），都是按照主菜单 File 下面的 Change dir 里指定的路径进行存盘的。(×)

5. main 函数在程序中的位置，与它作为程序开始执行的地位没有什么关系。(√)

四、编程

模仿例 1-2 编写程序，功能是从键盘接收两个整数，求出它们的和、差、积，然后分别打印输出。

答：程序编写如下：

```
#include "stdio.h"
main()
{
    int x, y, sum, sub, pro;
    scanf ("%d%d", &x, &y);    /* 从键盘输入两个整数，存放在变量 x 和 y 里 */
    sum = x + y;    /* 求出 x 与 y 的和，存放在变量 sum 里 */
    sub = x - y;    /* 求出 x 与 y 的差，存放在变量 sub 里 */
    pro = x*y;    /* 求出 x 与 y 的积，存放在变量 pro 里 */
    printf ("sum = %d, sub = %d, pro = %d\n", sum, sub, pro);
    /* 打印出 sum、sub、pro 的值 */
}
```

分析：在本程序里，用变量 x 、 y 存放输入的两个整数，用变量 sum 、 sub 和 pro 分别存放变量 x 与 y 的和、差、积。在程序设计时，由于是用变量来存放各类数据以及计算结果，因此，必须给出足够多的变量。

第 2 章 数据类型、运算符与表达式

2.1 内 容 要 点

在计算机中，数据是加工的对象，它有常量、变量之分。对数据的加工，是通过各种各样的运算符来进行的，由此也就组成了名目繁多的表达式。因此，数据、运算符和表达式，是学习和掌握 C 语言的基础，也是本章讲述的主要内容。

1. 常量

在 C 语言中，一个常量的类型，是由它的书写格式确定的。写出一个常量，就知道它是什么类型，进而也就知道了它所占用的存储空间是多少。

在 C 语言中，可以使用的常量有：整型常量、实型常量、字符常量和字符串常量。

(1) 整型常量可以有十进制、八进制、十六进制 3 种书写形式。十进制整型常量就是通常意义下的整数；八进制整型常量是由在通常意义下的八进制整数前加上前缀数字“0”构成；十六进制整型常量是由在通常意义下的十六进制整数前加上前缀“0x”(数字 0 和小写字母 x)构成。注意：八进制整型常量前的数字“0”，以及十六进制整型常量前的“0x”，它们只起一个标识的作用，用以告诉 C，后面是八进制整型常量或是十六进制整型常量，没有什么实际的意义。

无论是整型、短整型还是长整型常量，都是将其数值转换成相应的二进制数，存放在各自的存储区里的。它们有自己的取值范围。如果是长整型常量，那么在程序中书写时，还需要在它的末尾加上后缀小写的字母“l”，或大写的字母“L”，以便与整型或短整型常量区分开来。

(2) 实型常量只有十进制一种书写形式。或写为通常意义下的实数（由整数、小数点和小数 3 部分构成）；或写为指数形式（由尾数、小写字母 e 或大写字母 E，以及指数 3 部分构成）。指数形式有一个“规范化”的表示问题。

(3) 字符常量是用单引号前、后括住的单个字符。注意，单引号只起一个标识作用，并不算作字符常量的组成部分。由于转义字符本质上是一个字符，所以用单引号前、后括住一个转义字符，也就形成了一个字符常量。字符常量在内存中，存放的是该字符对应的 ASCII 码值。

(4) 字符串常量是用双引号前、后括住的零个或若干个字符，简称为“字符串”。一个字符串常量中所包含的字符个数，称为该字符串的“长度”。在内存中存放字符串时，不仅要存放每个字符相应的 ASCII 码值，而且必须在最后一个字节存放一个 ASCII 码值为 0 的字符

“\0”，作为该字符串的结束标记。

2. 变量

在程序执行过程中，允许其值发生变化的量，被称为“变量”。在 C 语言中，用变量来保存程序执行过程中的输入数据、中间结果以及最终结果等。

变量有 5 个重要的属性。变量名字：由它标识是这个变量还是那个变量；变量的数据类型：由它决定该变量所占存储区的大小；变量的存储类型：由它决定在哪个存储区域里为变量分配存储区；变量的初始值：由它决定该变量最初有值还是没有值，值是多少；变量的地址：由它指明该变量在内存中的具体位置。

因此，在程序中说明一个变量时，完整的做法是：要给出该变量的名字，要给出该变量的数据类型，要给出该变量的存储类型，还要给出该变量所取的初始值。至于地址，是由系统根据数据类型和存储类型分配而得。

(1) 变量名

程序中用到的每一个变量，都应该有一个自己的名字，以示区别。在 C 语言中，为变量起名应符合标识符的命名规则。变量名只能以字母或下划线开头。

(2) 变量的数据类型说明

整型变量的数据类型说明符：int、signed、unsigned、long、short。

注意：signed 与 unsigned 的区别在于对（二进制）数的最高位的解释不同，前者把最高位当作符号位看待，后者的最高位仍用于存储数据。

实型变量的数据类型说明符：float、double。

字符型变量的数据类型说明符：char。

注意：C 语言没有字符串变量！

(3) 变量的存储类型说明

系统用于存储用户数据的内存空间，有两个地方：静态存储区和动态存储区。

自动型变量的存储类型说明符：auto。

寄存器型变量的存储类型说明符：register。

静态型变量的存储类型说明符：static。

外部型变量的存储类型说明符：extern。

(4) 变量的初始化

在对变量进行说明的同时给变量赋予值，称为“变量的初始化”。

对于自动（auto）型和寄存器(register)型变量，如果在说明时给出初始化，那么每次运行到说明该变量所在的函数时，都必须进行初始化；如果在说明时没有进行初始化，那么在执行到第一次给它赋值的语句前，它的取值是不确定的。

对于在函数内说明的静态（static）型变量，如果在说明时给出了初始化，那么这个初始化的工作只在第一次运行时进行，下次再运行到说明该变量所在的函数时，不会重新初始化，而是继承上次运行后保留的当前值参与这次运算；如果在说明时没有进行初始化，那么 C 编译程序会自动为其进行初始化，将整型（int）变量赋予初值 0，将实型（float、double）变量赋予初值 0.0，将字符（char）型变量赋予初值空字符（即 ASCII 码值为 0 的字符“\0”）。下次再运行到该变量所在的函数时，不会重新初始化，而是继承上次运行后保留的当前值参与这次运算。

(5) 变量的地址

程序中说明了一个变量之后,就会在内存里分配一个存储区与之对应。这个存储区的第 1 个字节的地址,就是该变量的“地址”。一个变量的地址,可以通过在变量前加“&”符号来得到。变量的地址是一个无符号的整数。

3. 运算符与表达式

用来表示各种运算的符号称为“运算符”。有些运算符只需要一个运算对象,这种运算符称为“单目运算符”;有的需要两个,称为“双目运算符”;最多的则需要三个,称为“三目运算符”。

用运算符把运算对象连接在一起组成的式子,称为“表达式”。每个表达式按照运算规则进行运算,最终都会得到一个结果,它就是该表达式的“值”。当表达式中有多个运算符时,运算执行的先后顺序,称为“运算符的优先级”。对于相同级别的运算符,将由“运算符的结合性”来决定它们的运算顺序。在 C 语言中,有两种可能的结合性:“自左向右”的,意即由左向右遇到谁就先做谁;“自右向左”的,意即由右向左遇到谁就先做谁。

(1) 算术运算符与算术表达式

人们对算术运算符和由它们组成的算术表达式并不生疏,但要注意它们的特殊写法和新运算符的确切含义。

- 乘法运算符:“*”。
- 除法运算符:“/”。
- 模运算符:“%”。
- 增 1、减 1 运算符:“++”和“--”。

(2) 赋值运算符与赋值表达式

- 基本赋值运算符:“=”。

C 语言里的赋值运算符虽然形同是数学中的等号,但在 C 语言里它已经完全丧失了“等于”的原义,这是必须注意的。

- 算术自反赋值运算符:“+=”、“-=”、“*=”、“/=”、“%=”。

(3) 关系运算符与关系表达式

- 关系运算符:“<”、“<=”、“>”、“>=”、“==”、“!=”。

• 用关系运算符将两个运算对象连接起来所形成的表达式,称为“关系表达式”。关系表达式的最终结果是逻辑值:如果关系成立,则表达式取逻辑值“真”,用数值 1 表示;如果关系不成立,则表达式取逻辑值“假”,用数值 0 表示。

(4) 逻辑运算符与逻辑表达式

- 逻辑运算符:“!”、“&&”、“||”。

- 要把多个简单的条件组合成复杂的条件,就必须利用逻辑运算符。

• 参加逻辑运算时,用非 0 值表示逻辑真,而逻辑运算的结果是用数值 1 来表示逻辑真;至于逻辑假,无论是参与逻辑运算时还是求得的运算结果,都是以数值 0 来表示的。

• 对于&&运算符,只要其左边的运算对象为假,则整个表达式肯定取值“假”(数值 0),C 编译程序不会再对右边的运算对象进行求值。

• 对于||运算符,只要其左边的运算对象为真,则整个表达式肯定取值“真”(数值 1),C 编译程序不会再对右边的运算对象进行求值。

(5) 条件运算符与条件表达式

- 条件运算符：“?:”
- 条件运算符是 C 语言里惟一的一个三目运算符。由条件运算符构成的整个表达式，称为“条件表达式”。

(6) 逗号运算符和逗号表达式

- 逗号运算符：“,”
- 利用逗号运算符把若干个表达式“连接”在一起，这样的表达式称为“逗号表达式”。
- 逗号表达式的执行过程是：从左到右顺序计算诸表达式的值，并且把最右边表达式的值作为该逗号表达式的最终取值。

(7) 位运算符和位表达式

- 位逻辑运算符：“~”、“&”、“|”、“^”。
- 由位逻辑运算符和运算对象构成的表达式，称为“位逻辑表达式”。
- 移位运算符：“<<”、“>>”。
- 由位移运算符和运算对象构成的表达式，称为“移位表达式”。
- 位自反赋值运算符：“&= ”、“|= ”、“^= ”、“<<= ”、“>>= ”。

4. 数据类型的转换

(1) 当计算表达式，且表达式里的各个变量的数据类型不一致时，把参加运算的数据都自动转换成数据长度最长的数据类型，然后进行计算。

(2) 当把表达式的计算结果赋予变量，结果的数据类型与左边变量的数据类型不一致时，把结果的数据类型自动转换成变量的数据类型，然后赋予该变量。

(3) 如果有必要，可以使用强制式的数据转换方式。

2.2 习题解答

一、填空

1. 在 C 语言中，写一个十六进制的整数，必须在它的前面加上前缀 0x。
2. 在 C 语言中，是以 \0 作为一个字符串的结束标记的。
3. 增 1 运算符的功能是将运算对象加 1 后，把结果 回存 到运算对象。
4. 利用减 1 运算符“--”，下面的两个语句：

b = 5 + a; a -= 1;

表达的功能，可以由一个语句来实现。这个语句是：b = 5 + a --;。

分析：题目中所给的第一个语句“b = 5 + a;”的含义是把变量 a 的内容加上 5 以后，赋给变量 b。第二个语句“a -= 1;”的含义是把变量 a 减 1 后回存。因此，可以利用后缀减 1 运算符的“先使用后进行运算”的特性，来把第二个语句功能结合到第一个语句上，从而成为一个语句。

5. 用 C 语言描述：“x 是小于整数 m 的偶数”的表达式是 (x%2==0)&&(x<m)。

二、选择

1. 以下有关增 1、减 1 运算符中, 只有(D)是正确的。

- A. `---a` B. `++100` C. `a--b++` D. `a++`

分析: 根据 C 语言的规定, 当遇到连续出现多个加号或减号时, 就从左到右尽可能多地若干个加号或减号组成一个运算符。因此在 A 中, `---a` 的含义是`--(-a)`。这就是说, `--`将作用在`(-a)`上。但 `a` 是变量, `(-a)`却不是变量, 所以, A 是错的。在 B 中, 是把`++`作用在常数 100 上, 这显然是不对的。在 C 中, `a--`和 `b++`之间没有其他运算符存在, 不能构成一个表达式, 所以也是错的。

2. 逻辑表达式 `5&2||5|2` 的值是(B)。

- A. 0 B. 1 C. 2 D. 3

分析: 题目中涉及到 3 个运算符: “&”、“|”和“||”。前两个是位逻辑运算符, 第三个是逻辑运算符。由教材的表 2-3 知道, 位逻辑运算符的优先级高于逻辑运算符的优先级。因此逻辑表达式 `5&2||5|2` 的计算顺序实际是: `(5&2)|| (5|2)`。若以一个字节 (8 个二进制位) 来表示数值 5 和 2, 那么 `5&2` 是:

```

      00000101
& : 00000010
-----
      00000000
即值为 0 (逻辑假)。而 5|2 是:
      00000101
| : 00000010
-----
      00000111

```

即值为 7 (逻辑真)。因此, 逻辑表达式 `5&2||5|2` 的值是 1 (逻辑真)。

3. 设有变量说明: `int x = 5, y = 3;`那么表达式

`x > y ? (x = 1) : (y = -1)`

运算后, `x` 和 `y` 的值分别是(B)。

- A. 1 和 -1 B. 1 和 3 C. 5 和 -1 D. 5 和 3

分析: 注意, `x = 1` 和 `y = -1` 都是赋值语句。由于变量 `x` 的初值是 5, `y` 的初值是 3, 表明 `x > y` 成立。因此, 整个条件表达式的值应该是取 `x = 1` 的值, 即把 1 赋予变量 `x`。所以做完该表达式后, `x` 变为 1, `y` 保持不变, 仍为 3。所以本题的答案是 B。

4. 设有变量说明: `float x = 4.0, y = 4.0;`使 `x` 为 10.0 的表达式是(D)。

- A. `x -= y * 2.5` B. `x /= y + 9` C. `x *= y - 6` D. `x += y + 2`

5. 设有变量说明: `int a = 7, b = 8;`那么语句:

`printf ("%d, %d\n", (a + b, a), (b, a + b));`

的输出应该是(A)。

- A. 7, 15 B. 8, 15 C. 15, 7 D. 15, 8

分析: `printf` 语句中的输出变量列表里, 给出的是两个逗号表达式。第一个逗号表达式