



高等学校“十二五”规划教材

C语言程序设计习题集

◎主 编 崔永君 王芬琴

◎副主编 金 静 张黎明

C Y U Y A N C H E N G X U
S H E J I X I T I J I



西安电子科技大学出版社

<http://www.xduph.com>

高等学校“十二五”规划教材

C语言程序设计习题集

主 编 崔永君 王芬琴

副主编 金 静 张黎明

北京航空航天大学
★ 藏书 ★
图书馆

西安电子科技大学出版社



北航

C1714956

TP312C-44
91

072380310

内 容 简 介

本书是《C 语言程序设计》一书的配套习题集,由习题解答、典型题目及解析两部分组成。习题涵盖了 C 语言程序设计考试的主要题型(选择题、程序填空题、阅读程序写运行结果题和编程题等),综合了数据类型、程序结构和典型算法的运用,题目数量较大,每道题都配有详细的分析和解答。教师可根据学生的学习情况有针对性地选用。

本书按基础课程要求编写,题目有易有难,适用范围广。本书可作为高等院校计算机专业的教学用书,也可作为高职高专计算机专业及非计算机专业的教学用书,还可作为工程技术人员的自学参考书。

王 芬 琴 主 编

崔 永 君 主 编

图书在版编目(CIP)数据

C 语言程序设计习题集/崔永君,王芬琴主编. —西安:西安电子科技大学出版社,2013.11

高等学校“十二五”规划教材

ISBN 978-7-5606-3235-3

I. ① C… II. ① 崔… ② 王… III. ① C 语言—程序设计—高等学校—习题集 IV. ① TP312-44

中国版本图书馆 CIP 数据核字(2013)第 289416 号

策 划 杨丕勇

责任编辑 南 景 杨丕勇

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfb001@163.com

经 销 新华书店

印刷单位 陕西天意印务有限责任公司

版 次 2013 年 11 月第 1 版 2013 年 11 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 9.5

字 数 221 千字

印 数 1~3000 册

定 价 18.00 元

ISBN 978-7-5606-3235-3/TP

XDUP 3527001-1

如有印装问题可调换

本社图书封面为激光防伪覆膜,谨防盗版。

前 言

程序设计是高等学校本科教育的必修课程,可以有效地帮助学生了解和掌握计算思维。C 语言以其语言简洁紧凑、使用灵活方便、功能强大、应用面广等诸多优点成为学习计算机程序设计语言的首选。然而,正是由于其功能强、编程限制少、灵活性大,也意味着易出错、调试困难、不好把握。同时,C 语言程序设计是一门实践性很强的课程,在强调理论环节教学的同时,必须配套足够的实践、实验环节的教学,学生只有通过足够的实验、编程才可能学好这门课程。针对上述问题,作者组织了长期从事 C 语言程序设计教学的教师编写了本书。

本书是《C 语言程序设计》(崔永君、彭静主编,西安电子科技大学出版社出版)一书的配套习题集。本书基本内容的难易程度与计算机二级考试相当,选择内容更加注重算法分析及综合编程能力的培养。对程序中难以理解的地方,大多添加了注释,以便读者循序渐进地掌握程序设计的基本思想和基本方法。

本书由崔永君、王芬琴担任主编,金静、张黎明担任副主编,具体编写分工如下:第一部分和第二部分的第 14、17 章由金静编写;第二部分的第 12 章由王芬琴编写;第二部分的第 13、15、16 章由崔永君编写;第二部分的第 18~22 章由张黎明编写。全书由崔永君、王芬琴负责统稿。

本书在编写过程中得到了兰州交通大学博文学院电信工程系计算机教研室全体教师热情的帮助和支持,对李贵栓老师和王芬琴老师的无私奉献,在此表示衷心的感谢!

尽管在编写此书的过程中编者做了许多努力,但由于编者水平有限,书中难免存在疏漏,恳请广大读者批评指正。

编 者

2013 年 7 月

第 16 章	循环结构程序设计	1
第 17 章	数组	1
第 18 章	函数	1
第 19 章	指针	1
第 20 章	其他数据类型	1
第 21 章	编译预处理与位运算	1
第 22 章	文件	1

目 录

第一部分 习题解答

第1章 C语言概述	2
第2章 基本数据类型、运算符和表达式	4
第3章 顺序结构程序设计	8
第4章 选择结构程序设计	10
第5章 循环结构程序设计	13
第6章 数组	18
第7章 函数	22
第8章 指针	30
第9章 其他数据类型	38
第10章 编译预处理与位运算	48
第11章 文件	53

第二部分 典型题目及解析

第12章 C语言概述	60
第13章 基本数据类型、运算符和表达式	63
第14章 顺序结构程序设计	69
第15章 选择结构程序设计	73
第16章 循环结构程序设计	80
第17章 数组	89
第18章 函数	100
第19章 指针	115
第20章 其他数据类型	127
第21章 编译预处理与位运算	136
第22章 文件	141

语言的特点。它具有以下几个特点:

(1) 简洁、紧凑、使用灵活、方便。C语言共有32个关键字,9种数据类型,书写形式自由,区分大小写。它把高级语言的基本结构和语句与低级语言的实用性结合起来。它可以像汇编语言一样对位、字节和地址进行操作。

(2) 运算符丰富。C语言运算符比任何其他高级语言中难以实现的运算。转换等都作为运算符处理,从而使C语言的运算类型极其丰富,表达式的种类多,使用各种运算符,在其他高级语言中难以实现的运算。

第一部分

(3) 数据类型丰富。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,能用来实现各种复杂的数据结构的运算,概念清晰,概念,使用效率高。

(4) 表达方式灵活实用。C语言提供多种运算符和表达式的使用方法,可以通过多种途径获得,其程序设计更主动、灵活,不受语言限制,大大增强了程序的可移植性。如整型量与字符型数据及逻辑型数据。

(5) 生成目标代码质量高,程序执行效率高。C语言描述问题比汇编语言简单,可读性好,易于调试、修改和移植。

(6) 可移植性好。C语言编写的程序,稍加改动,即可移植到不同的环境中运行。

习

题

解

答

2. 解析: 结构化程序设计方法中的顺序、选择和循环三种基本控制结构,在C语言中都有实现。C语言提供了丰富的运算符和表达式,使得程序编写更加灵活。

3. 解析: 构成一个C语言程序的函数,可以分为两种基本类型:一种是系统函数,另一种是用户自定义函数。系统函数是由C语言编译器提供的,用户可以直接调用。用户自定义函数是由用户根据应用程序的需要而定义的,用以完成特定功能的程序模块。

就是把一个大问题分解成若干个比较容易求解的小问题,然后分别求解。这种解决问题的方法,称为模块化程序设计方法。在C语言中,函数是实现模块化的重要手段。通过函数,可以将程序中的某些部分封装起来,形成一个独立的模块,方便地调用。这不仅可以提高程序的模块化程度,还可以提高程序的效率和可维护性。

$$= 01.372$$

10. 解析: 在C语言中,数组的声明和定义是分开进行的。数组的声明只是告诉编译器数组的存在,而数组的定义则是为数组分配内存空间。在本题中,数组的声明和定义是分开的,因此需要分别进行。

二 简答题

1. 解析: C语言是一种面向过程的编程语言,它的特点是简洁、紧凑、使用灵活、方便。C语言的数据类型丰富,运算符丰富,表达式的种类多,使用各种运算符,在其他高级语言中难以实现的运算。

第 1 章 C 语言概述

长陪一策

一、选择题

1. B

2. C

3. C

4. D

5. A

6. C

7. B

8. C

解析：将该二进制数从右到左每四位分隔成一组，再将每一组转换成对应的十六进制数。1011 对应十六进制数的 B，0110 对应十六进制数的 6，1111 对应十六进制数的 F。

9. B

解析：将八进制数每一位分解成三位的二进制数。3 对应于 011，0 对应于 000，7 对应于 111。

10. A

解析：十进制整数转换成 R 进制数，采用除基数 R 取余法，将得到的余数倒序排列。

11. B

解析：将该二进制数从右到左每三位分隔成一组，不足三位的在最左端加零补齐三位，再将每一组转换成对应的八进制数。分组补零后为 000 | 110 | 101，各组对应的八进制数为 0、6、5。

12. B

解析：用位权法将该二进制数展开，再求其和即可。

$$\begin{aligned}(1011011.011)_2 &= 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + \\ &\quad 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3} \\ &= 91.375\end{aligned}$$

13. D

解析：前端加 0x 表示该数是十六进制数，前端加 0 表示该数是八进制数，否则为十进制数，故 A、C 项被排除。八进制数中的有效数字是 0~7，故 B 选项被排除。

二、简答题

1. 解析：C 语言是一种计算机程序设计语言，它既具有高级语言的特点，又具有汇编

语言的特点。它具有以下几个特点:

(1) 简洁、紧凑,使用灵活、方便。C语言共有32个关键字,9种控制语句,程序书写形式自由,区分大小写。它把高级语言的基本结构和语句与低级语言的实用性结合起来,可以像汇编语言一样对位、字节和地址进行操作。

(2) 运算符丰富。C语言共有34种运算符。在C语言中,把括号、赋值号、强制类型转换等都作为运算符处理,从而使C语言的运算类型极其丰富,表达式类型多样。灵活使用各种运算符,可以实现其他高级语言中难以实现的运算。

(3) 数据类型丰富。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,能用来实现各种复杂的数据结构的运算,并引入了指针的概念,使程序效率更高。

(4) 表达方式灵活实用。C语言提供多种运算符和表达式值的方法,对问题的表达可通过多种途径获得,其程序设计更主动、灵活。它的语法限制不太严格,程序设计自由度大,如对整型量与字符型数据及逻辑型数据可以通用等。

(5) 生成目标代码质量高,程序执行效率高。C语言描述问题比汇编语言迅速,工作量小,可读性好,易于调试、修改和移植,而代码质量与汇编语言相当。

(6) 可移植性好。C语言编写的程序,不改动或稍加改动,就可移植到另一个完全不同的环境中运行。

2. 解析:结构化程序设计方法中的三种基本结构为顺序、选择和循环。顺序结构是按从上到下的顺序执行程序语句;选择结构先根据条件做出判断,再决定执行哪一种操作;循环结构是按照一定条件,反复执行某些步骤。

3. 解析:构成一个C语言程序的函数可以划分为两种基本类型:一种是由C语言系统提供的函数,这类函数用户可以直接在程序中调用,从而完成一定的功能,这类函数也被称为C语言的“库函数”或“标准函数”;另一种是由用户在程序中定义的函数,是由用户根据应用程序的需要而定义,用以完成特定功能的程序模块,称为“用户自定义函数”。

4. 解析:人们在求解一个复杂问题时,通常采用的是逐步分解、分而治之的方法,也就是把一个大问题分解成若干个比较容易求解的小问题,然后分别求解。程序员在设计一个复杂的应用程序时,往往也是把整个程序划分为若干个功能较为单一的程序模块,然后分别予以实现,最后再把所有的程序模块像搭积木一样装配起来。这种在程序设计中分而治之的策略,被称为模块化程序设计方法。在C语言中,函数是程序的基本组成单位,因此可以很方便地用函数作为程序模块来实现C语言程序。利用函数,不仅可以实现程序的模块化,使程序设计得简单和直观,提高了程序的易读性和可维护性,而且还可以把程序中普遍用到的一些计算或操作编成通用的函数,以供随时调用,这样可以大大地减轻程序员编写代码的工作量,即提高了代码的可复用性。

第2章 基本数据类型、运算符和表达式

一、选择题

1. B

2. C

3. C

4. D

解析：十进制整数不能加千位分隔符，故 A 选项不正确；用指数形式表示的十进制实数，指数必须为整数，故 B 选项不正确；C 选项中，斜杠字符要用转义字符“\”表示，故不正确；D 选项表示该字符串常量包含一个 ASCII 码值为八进制数 007 的字符，是正确的。

5. D

解析：字长是计算机系统的主要性能指标，是计算机作为整体能一次性传递和处理的二进制数的位数，不同的计算机系统字长也不同。

6. D

7. C

解析：C 语言中规定标识符由数字、字母和下划线组成，且由字母和下划线打头，故 C 不符合要求。

8. B

解析：用十进制指数形式表示实数时，字母 E 和 e 之前必须要有数字，E 和 e 之后的指数必须是整数，故 A、C、D 可排除。

9. D

10. D

11. C

12. AB

解析：C 语言允许不同数据类型的常量、变量和表达式执行混合运算，系统会自动类型转换，或者用户可使用强制类型转换。A 选项中，(int)表示将 double 型的 x 与 y 相减的结果强制转换成 int 型，整型变量 a 再与其做求余操作，故正确。B 选项中关系运算符 != (不等于)的优先级高于赋值运算符，先判断表达式 $x \neq y$ 的值，为真则将其值 1 赋给整型变量 a；反之，则将 0 赋给整型变量 a，故正确。

13. D

14. C

16. C

解析：由于输入值之间的分隔符是逗号，则 scanf 语句中格式控制符 %d 之间也必须用逗号隔开，故选择 C。

17. D

解析：x-- 是自减后缀表达式，先使用原值再自减，因此先输出原值 11，再自减为 10；--y 是自减前缀表达式，y 先自减为 10 再输出。

18. D

解析：(double)b 表示将整型变量 b 的值强制类型转换成双精度浮点型，然后将其与浮点型变量 c 求和得到 25.000000，再调用系统函数 sqrt 对和开平方根得到 5.000000。由于变量 a 是整型，根据系统自动类型转换的规律，a 将被赋值为整数 5。根据 printf 函数中的输出格式控制符，可得结果为 11，12.000000，13。

19. A

解析：语句 i=(int)x 将 x 的值 3.6 截取整数部分得到整数 3 后赋值给变量 i，变量 x 中的值没有变化，还是初值 3.6。

20. A

解析：字符型变量与整型常量进行算术运算时用字符型变量的 ASCII 码值运算，c-25 的结果是 97(字符 'z' 的 ASCII 码值为 122)，该结果以字符格式(%c)输出，则输出为字符 'a'(字符 'a' 的 ASCII 码值为 97)。

二、填空题

1. scanf; printf

2. --x 或 x = x-1

解析：printf 中输出格式说明符为 %d，表示以带符号的十进制整型输出。输出结果为十进制数 13，转换为八进制数 015，因此该结果必由 x 中的八进制数 016 自减而来。

3. b++

解析：条件表达式由条件运算符构成，并常用条件表达式构成一个赋值语句，其一般形式为 x = <表达式 1> ? <表达式 2> : <表达式 3>，其意义是：先求解表达式 1，若为非 0(真)，则求解表达式 2，将表达式 2 的值赋给 x；若为 0(假)，则求解表达式 3，将表达式 3 的值赋给 x。

题目中关系表达式 b < c 为真，则空中的表达式值应为 3，结合 b 最终结果为 4，可推出该空填 b++，即先将 b 的旧值 3 赋给 a，再自增为 4。

4. 6.000000

解析：自增运算符优先级高于乘法*，表达式 ++a*++a 等价于(++a)*(++a)。a 先自增为 2，再自增为 3，相乘结果以小数形式输出，隐含六位小数。

5. ch+32; %d

解析：小写字母的 ASCII 码值比对应大写字母的 ASCII 码值大 32。例如，大写字母 A 的 ASCII 码值为 65，小写字母 a 的 ASCII 码值为 97。题目中条件表达式的表达式 1 用于判断字符变量 ch 是否是大写字母，若是则将其 ASCII 码值加 32 转换成对应小写字母的

ASCII 码值，因此条件表达式的表达式 2 填入 `ch+32`。题目要求输出字符变量的 ASCII 码值，`printf` 函数中的格式说明符应该填入十进制整型格式 `%d`。

6. 4; 11
解析：逗号运算符的优先级低于赋值运算符，因此表达式 `x = a + 1, b + 6` 等价于 `(x = a + 1), b + 6`，`x` 的值被赋为 4，整个逗号表达式的值为最后一个表达式 `b + 6` 的值 11。表达式 `y = (a + 1, b + 6)` 的值为逗号表达式中最后一个表达式 `b + 6` 的值 11。因此，输出结果为 `x = 4, y = 11`。

7. 4
解析：根据运算符的优先级和结合方向，题目中的条件表达式等价于 `k = (a > b ? a : (c > d ? c : d))`，即若 `a > b`，则取 `a` 的值；否则，再判断 `c` 是否大于 `d`，是则取 `c` 的值，否则取 `d` 的值。因此结果为 `d` 的值 4。

8. 2008
解析：`printf` 函数中的 `%d` 说明符是十进制输出，八进制数 010 以十进制输出的结果为 8。

9. 4.900000; 4
解析：`y = (int)x` 表示将 `x` 的值 4.9 截掉小数部分强制转换成整数 4 赋值给整型变量 `y`。`%f` 表示将 `x` 以小数形式输出，默认包含六位小数。

10. 15
解析：`scanf` 函数中格式说明符前面的整数指定了输入数据的宽度，因此 `x` 中读入整数 12，`y` 中读入整数 3。

三、程序设计

1. 解析：将三个数按从小到大的顺序排序后输出即可。源代码如下：

```
#include<stdio.h>

main()
{
    float x,y,z;
    scanf("%f,%f,%f",&x,&y,&z);
    /*找出三个数中的最大值，放入变量 z 中*/
    if(x>y)
    {
        t=x;
        x=y;
        y=t;
    }
    if(y>z)
    {
        t=y;
        y=z;
        z=t;
    }
}
```

2.25 z=t; 37

解析: } 变量 b 中存放的 25 是八进制数(0 打头), c 中存放的 25 是十进制数。根据格式说明 /*找出剩余两个数中的最大值, 放入变量 y 中*/, 分别为 21 和 37。

3. n1 if(x>y) n2=%d

4. 2. {

5. 5.0. t=x;

解析: x=y; scanf 函数的格式说明字符串中, 格式字符和格式修饰符之间用空格分隔。在输入数据 y=t; 输入与这些字符相同的字符。例如本题中的两个空格, 输入时要用

三、程序设计

printf(" %.3f %.3f %.3f",x,y,z);

}

2. 解析: 判断闰年的条件是能被 4 整除但不能被 100 整除, 或能被 400 整除的年份是闰年。源代码如下:

main()

{

int y;

scanf("%d",&y);

/*读入要判断的年份*/

if((y%4==0&& y%100!=0)|| (year%400==0))

printf("%d is a leap year\n",y);

/*是闰年*/

else

printf("%d is not a leap year\n",y);

/*不是闰年*/

}

3. 解析: 根据题意, 删除原程序中第 10 行代码。

%d\n",a*b*c);

4. 略。

解析: 由于 y 为 int 型, 且 y 为奇数, 故 y 为奇数时, y%2 为 1, 为偶数时, y%2 为 0。

解析: 由于 y 为 int 型, 且 y 为奇数, 故 y 为奇数时, y%2 为 1, 为偶数时, y%2 为 0。

解析: 由于 y 为 int 型, 且 y 为奇数, 故 y 为奇数时, y%2 为 1, 为偶数时, y%2 为 0。

解析: 由于 y 为 int 型, 且 y 为奇数, 故 y 为奇数时, y%2 为 1, 为偶数时, y%2 为 0。

解析: 由于 y 为 int 型, 且 y 为奇数, 故 y 为奇数时, y%2 为 1, 为偶数时, y%2 为 0。

解析: 由于 y 为 int 型, 且 y 为奇数, 故 y 为奇数时, y%2 为 1, 为偶数时, y%2 为 0。

解析: 由于 y 为 int 型, 且 y 为奇数, 故 y 为奇数时, y%2 为 1, 为偶数时, y%2 为 0。

解析: 由于 y 为 int 型, 且 y 为奇数, 故 y 为奇数时, y%2 为 1, 为偶数时, y%2 为 0。

解析: 由于 y 为 int 型, 且 y 为奇数, 故 y 为奇数时, y%2 为 1, 为偶数时, y%2 为 0。

ASCII 码值, 因此条件表达式的表达式 2 填入 $ch+32$ 。题目要求: 若 x 为偶数, 则 y 为 x 的值; 若 x 为奇数, 则 y 为 $x+1$ 的值; printf 函数中的格式说明符应该填入十进制整型格式 $\%d$ 。

6.4: 11

10 中 x 量变入第 1 个最大值由第 2 个西余除出其

解析: 逗号运算符的优先级低于赋值运算符, 因此表达式 $x=(x+1), b=b, x$ 的值为 $x+1$ 。

第 3 章 顺序结构程序设计

表达式 $y=(a+1), b=b, x$ 的值为 $x+1$ 。题目要求: 若 x 为偶数, 则 y 为 x 的值; 若 x 为奇数, 则 y 为 $x+1$ 的值。结果为 $x=4, y=1$ 。

7.4

10.4

解析: 根据运算符的优先级和结合方向, 题目中的条件表达式 $x \neq 0 ? x : x+1$ 的值为 x 。

一、选择题

1. A 此结果为 d 的值 4。

解析: scanf 函数的格式说明, 对于字符串中除格式字符和格式修饰符之外的其他字符, 在输入数据时应输入与这些字符相同的字符。

2. C

解析: 格式说明符 $\%o$ 表示以八进制无符号整数形式输出。变量 m 中存放的是八进制数 256(以 0 开头), 输出结果不变; n 中存放的是十进制数 246, 输出结果是其对应的八进制数 400。

3. C

解析: printf 函数中指定的列宽小于数据的实际宽度, 则按数据的实际长度输出。y 是八进制数 12, 格式说明符 $\%2d$ 表示输出其对应的十进制数 10。

4. B

解析: 语句 $a=f/c*=(x=6.5)$ 中 x 被赋值为 6.5; 复合赋值运算符 $\%2d$ 表示输出其对应的十进制数 10。

5. A

6. D

解析: (x,y) 是一个逗号表达式, 其值为最后一个表达式的值, 即 y 的值。因此输出结果是 y 的值 2003。

7. C

解析: printf 函数中指定的列宽 2 小于数据的实际宽度 4, 则按数据的实际长度输出。

8. D

解析: $(++x,y++)$ 是逗号表达式, 其值是最后一个表达式 $y++$ 的值。 $y++$ 是后缀表达式, 所以先输出 y 的旧值 3 后再将 y 自增。

9. D

10. B

二、填空题

1. 11

解析: 整型变量用八进制格式($\%o$)输出, 十进制数 9 对应的八进制数是 11。

2.25 21 37

解析: 变量 `b` 中存放的 25 是八进制数(0 打头), `c` 中存放的 25 是十六进制数(0x 打头), 格式说明符 `%d` 表示要用十进制形式输出 `b` 和 `c` 的值, 分别为 21 和 37。

```
3. n1=%d\nn2=%d
```

```
4. 2, 1
```

```
5. 5.0, 4, c=3
```

解析: 对于 `scanf` 函数的格式说明字符串中除格式字符和格式修饰符之外的其他字符, 在输入数据时应输入与这些字符相同的字符。例如本题中的两个逗号和“`c=`”。

三、程序设计

1. 解析: 要将原字母以其后面的第四个字母代替, 将其 ASCII 码值加 4 即可。例如, 字母 'A' 的 ASCII 码值是 65, 69 是字母 'E' 的 ASCII 码值。源代码如下:

```
#include<stdio.h>
main()
{
    char c1='C',c2='h',c3='i',c4='n',c5='a';
    c1+=4;
    c2+=4;
    c3+=4;
    c4+=4;
    c5+=4;
    printf("%c%c%c%c%c",c1,c2,c3,c4,c5);
}
```

2. 略。

3. 解析: 根据题意, 删除原程序第 4、5 条语句, 将最后一条语句改为 `printf("a*b*c=%d\n",a*b*c);`。

4. 略。

```
#include<stdio.h>
main()
{
    float profit,bonus;
    printf("please input your profit:");
    scanf("%f",&profit);
    if(profit<10000)
        bonus=profit*0.1;
    else if(profit<20000)
        bonus=10000*0.1+(profit-10000)*0.075;
    else if(profit<40000)
        bonus=10000*0.1+(10000*0.075+20000*0.05);
    else if(profit<60000)
        bonus=10000*0.1+(10000*0.075+20000*0.05+10000*0.03);
    else
        bonus=10000*0.1+(10000*0.075+20000*0.05+10000*0.03+10000*0.015);
    printf("bonus=%f",bonus);
}
```

第4章 选择结构程序设计

一、选择题

1. D

解析: if 语句的判断表达式是赋值语句, 变量 a 被赋值为 b 和 c 的和 0, 因此该表达式的值为 0, 执行 else 分支。一定要注意区分赋值运算符=和条件运算符等于号==。

2. A

解析: x 的值为 1, 进入多分支结构 case 1, a 自增为 1; 由于该分支后没有 break 语句, 接着执行语句标号 case 2 后的语句。因此 a 自增为 2, b 自增为 1。

3. C

解析: 自左向右计算逻辑或表达式 A||B 的值时, 一旦得出运算量 A 的值为 True(非零值), 则表达式一定为 True, 所以不再计算 B; 自左向右计算逻辑与表达式 A&&B 的值时, 一旦得出运算量 A 的值为 False(零), 则整个表达式一定为 False, 所以不再计算 B。表达式 (j++||k++) 中先使用 j 的旧值 1, j 再自增为 2, 由于该逻辑或表达式的值可以确定为真, k++ 不再执行, k 值仍然为 2。到此还不能确定整个逻辑与表达式的值, 接着判断表达式 i++ 的值, 先使用 i 的旧值 1, i 再自增为 2。到此, 整个表达式的值为 true(1)。输出值为 2, 2, 2。

4. C

解析: 该题目是嵌套的 if else 语句构造的多分支结构。

5. D

6. A

解析: p=f(j,k) 表示调用函数 f, 将函数 f 的返回值赋给变量 p, 其中 j 和 k 作为形参将其值传递给实参 a 和 b。

7. C

解析: case 后面必须是常量表达式, 不能是变量, 而题目中的 a、b、c 是变量。

8. D

9. B

解析: B 选项的功能与题目相反。

二、填空题

1. 4

解析: C 语言中, 从最内层开始 else 总是与它上面最近的未曾配对的 if 配对, 同时两个 if else 语句不能交叉。将题目中配对的 if else 语句加 {} 和缩进格式(配对的 if else 左对齐)后如下:


```

if(a==1)
{
    if(b!=2)
    {
        if(c==3) d=1;
        else d=2;
    }
    else {
        if(c!=3) d=3;
        else d=4;
    }
}
else d=5;

```

解析: 当 a=1 时, 进入内层循环, 当 b=2 时, 进入内层循环, 当 c=3 时, 进入内层循环, 此时 d=1, 当 c!=3 时, d=3, 当 c=3 且 b!=2 时, d=2, 当 b!=2 且 a!=1 时, d=5, 所以 d 的值为 4。

2. 1, 0

解析: 自左向右计算逻辑与表达式 A&&B 的值时, 一旦得出运算量 A 的值为 False(零), 则整个表达式一定为 False, 所以不再计算 B。表达式(++a<0)先将 a 自增为 0 再与 0 比较, 结果为假, 则整个逻辑与表达式为假, B 表达式不再运算。执行 else 分支的输出语句, 结果为 1, 0。

3. A B other

解析: 函数 getchar() 读入从键盘输入的字符 'A' 后赋给变量 a, 字符 A 的 ASCII 码值为 65, 进入第一个分支输出字符 'A'。由于该分支没有 break 语句, 顺次执行语句标号 case 66 及 default 后的语句。

三、程序设计

1. 解析: 分析题目可知, 必须使用多分支结构, 可使用 if else 语句, 也可使用 switch 语句实现多分支结构。

源代码如下:

```

#include "stdio.h"

main()
{
    float profit, bonus;
    printf("please input your profit=");
    scanf("%f", &profit); // 读入利润
    if(profit < 100000) // 利润小于 100000
        bonus = profit * 0.1;
    else if(profit < 200000) // 利润在 100000 ~ 200000
        bonus = 100000 * 0.1 + (profit - 100000) * 0.075;
    else if(profit < 400000) // 利润在 200000 ~ 400000
        bonus = 100000 * 0.1 + 100000 * 0.075 + (profit - 200000) * 0.05;
    else if(profit < 600000) // 利润在 400000 ~ 600000
        bonus = 100000 * 0.1 + 100000 * 0.075 + 200000 * 0.05 + (profit - 400000) * 0.03;
}

```

```

else if(profit<1000000)                //利润在 600000~1000000
    bonus=100000*0.1+100000*0.075+200000*0.05+200000*0.03+(profit-
        600000)*0.015;
else
    bonus=100000*0.1+100000*0.075+200000*0.05+200000*0.03+ 400000*
        0.015+ (profit-1000000)*0.01;    //利润大于 1000000
printf("Your bonus=%f",bonus);
}

```

2. 略。

3. 解析：要判断输入的年月日是这一年的第几天，将这个月之前的总天数加上输入的日即可。要注意的是，如果输入的年份是闰年，则二月是 29 天。源代码如下：

```

#include "stdio.h"
main()
{ int year,month,day;
  int s=0,s1=31,s2=59,s3=90,s4=120,s5=151,s6=181,s7=212,s8=243,
    s9=273,s10=304,s11=334;    //si 表示 1 月到 i 月总共的天数
  printf("请输入年月日:");
  scanf("%d,%d,%d",&year,&month,&day);
  switch(month)
  {
    case 1:s=day; break;
    case 2:s=s1+day; break;
    case 3:s=s2+ day; break;
    case 4:s=s3+ day; break;
    case 5:s=s4+ day; break;
    case 6:s=s5+ day; break;
    case 7:s=s6+ day; break;
    case 8:s=s7+ day; break;
    case 9:s=s8+ day; break;
    case 10:s=s9+ day; break;
    case 11:s=s10+ day; break;
    case 12:s=s11+ day; break;
    default: printf("data error\n"); break;
  }
  if(((year%4==0)&&(year%100!=0))||((year%400==0))&&(month>2)) //判断是否是闰年
    s=s+1;
  //闰年的二月是 29 天，若输入的月份是 3 月~12 月，则在其原天数 s 上再加 1
  printf("该日为当年第%d 日",s);
}

```