

109

全国高等教育自学考试系列辅导教材

高级语言程序设计 复习与考试指导

计算机及应用专业(专科)

周保顺 马 越 孙 琦 编



A0967331

高等教育出版社

内 容 提 要

本书是全国高等教育自学考试计算机及应用专业系列辅导教材之一,是专科高级语言程序设计课程的复习与考试指导教材。它依据自学考试指导委员会制定的考试大纲“识记、领会、识别、应用”的四个层次,紧密结合自考考生特点,以典型例题的形式对每一章节的内容进行透彻的分析与解答,每一章后配有大量自测试题,供考生及时检验学习效果;同时本书还提供了四套模拟试卷,全面涵盖本课程的考核知识点,以利考生熟悉考试过程,从而进入良好的考试状态并顺利通过考试。

本书内容包括:概述,基本数据类型和运算,顺序结构、选择结构和循环结构的程序设计,数组,指针,函数,结构型,共用型和枚举型,文件,编译预处理与带参数的主函数,及附录等。

本辅导教材除适用于计算机及应用专业的考生外,还适用于计算机信息管理、计算机通信工程和计算机应用及教育等专业相同课程的考生,也可以作为工程技术人员、社会读者的学习参考用书。

图书在版编目(CIP)数据

高级语言程序设计复习与考试指导 / 周保顺, 马越, 孙琦编. —北京: 高等教育出版社, 2002.8

高等教育自学考试计算机及应用专业(专科)辅导教材

ISBN 7-04-010742-2

I. 高... II. ①周... ②马... ③孙... III. 高级语言—程序设计—高等教育—自学考试—自学参考资料
IV. TP312

中国版本图书馆CIP数据核字(2002)第044166号

高级语言程序设计复习与考试指导

周保顺 马 越 孙 琦 编

出版发行 高等教育出版社

社 址 北京市东城区沙滩后街55号

邮政编码 100009

传 真 010-64014048

购书热线 010-64054588

免费咨询 800-810-0598

网 址 <http://www.hep.edu.cn>

<http://www.hep.com.cn>

经 销 新华书店北京发行所

印 刷 北京市鑫鑫印刷厂

开 本 787×1092 1/16

印 张 16

字 数 380 000

版 次 2002年8月第1版

印 次 2002年8月第1次印刷

定 价 21.90元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

前 言

本书是全国高等教育自学考试计算机及应用专业（专科）《高级语言程序设计》课程的必备应试指导书。

帮助参加高等教育自学考试的考生顺利地通过考试，并且提高自学的能力和效果，是我们编写本书所遵循的唯一宗旨。

C 语言程序设计是一门实践性很强的课程，既需要掌握概念，又要注重动手编程，上机调试。故在使用本书的过程中，考生要注意知识和技能两方面的学习与实践。

在知识方面：

应熟练掌握 C 语言的数据类型（基本类型、构造类型、指针类型等）和各类运算符，能正确使用表达式实现各种数据的简单加工；应熟练掌握 C 程序的三种基本结构（顺序、选择、循环）的特点，能使用相关语句完成这三种基本结构的程序设计任务；应掌握 C 语言的常用库函数的使用，以及用户函数的定义、调用、参数传递等方法。

在技能方面：

应熟练掌握阅读和分析简短程序的方法和技巧；应熟练掌握设计和调试简短程序的方法及技巧；了解并初步掌握实用程序的开发与调试技术。

关于“考核知识点与考核要求”中有关提法的说明：

本课程要求考生学习和掌握的知识点都是考核的内容。

关于四个能力层次概念的解释：

识记：要求考生能够识别和记忆本课程中规定的有关知识点的主要内容（如定义、表达式、公式、原则、重要结论、方法、步骤及特征、特点等），并能根据考核的不同要求，做出正确的表述、选择和判断。

领会：要求考生能够领悟和理解课程中规定的有关知识点的内涵与外延，熟悉其内容要点和它们之间的区别与联系，能根据考核的不同要求，做出正确的解释、说明和论述。

简单应用：要求考生能够运用本课程中规定的少量知识点，分析和解决一般应用问题，如简单计算、绘图和分析、论证等。

综合应用：要求考生能够运用本课程中规定的多个知识点，分析和解决较复杂的应用问题，如计算、绘图、简单设计、编程和分析、论证等。

参加本书编写的人员有：第一、二、三章及附录由马越同志编写。第四章由孙琦同志编写。第五、六、七、八、九章由周保顺同志编写。

本书既可作为高自考的应试指导书，也可作为大专院校在校学生学习 C 语言程序设计课程的教学辅导书。

在本书的编写过程中，参考了大量有关 C 语言程序设计的书籍和资料，在此不一一列举。编者对这些参考文献的作者表示衷心的感谢。

由于编者水平有限，书中难免存在错误和不当之处，敬请广大考生和读者不吝赐教，以便再版时修改。

作 者
2002 年 6 月

目 录

第一章 概述

1.1 必考知识点及相互关系	(1)
1.2 重点与难点分析	(4)
1.3 典型例题分析及解答	(6)
1.4 自测试题	(8)
自测试题参考答案	(9)

第二章 基本数据类型和运算

2.1 必考知识点及相互关系	(11)
2.2 重点与难点分析	(23)
2.3 典型例题分析及解答	(31)
2.4 自测试题	(33)
自测试题参考答案	(34)

第三章 顺序结构、选择结构和 循环结构的程序设计

3.1 必考知识点及相互关系	(36)
3.2 重点与难点分析	(41)
3.3 典型例题分析及解答	(56)
3.4 自测试题	(62)
自测试题参考答案	(64)

第四章 数组

4.1 必考知识点及相互关系	(65)
4.2 重点与难点分析	(66)
4.2.1 一维数组	(66)
4.2.2 多维数组	(69)
4.2.3 字符数组与字符串	(72)
4.3 典型例题分析及解答	(75)
4.4 自测试题	(83)
自测试题参考答案	(89)

第五章 指针

5.1 必考知识点及相互关系	(93)
----------------------	------

5.2 重点与难点分析	(93)
5.2.1 地址与指针的概念	(93)
5.2.2 指针变量的定义与初始化	(96)
5.2.3 通过指针变量引用一个存储单元 (取值运算符*)	(97)
5.2.4 一维数组和指针	(99)
5.2.5 字符串与指针	(101)
5.2.6 二维数组与指针	(103)
5.2.7 指针数组	(106)
5.2.8 多级指针	(108)
5.2.9 本章重点内容概括	(109)
5.3 典型例题分析及解答	(109)
5.4 自测试题	(113)
自测试题参考答案	(115)

第六章 函数

6.1 必考知识点及相互关系	(117)
6.2 重点与难点分析	(117)
6.2.1 函数的概念	(117)
6.2.2 函数调用中某些问题	(120)
6.2.3 函数的嵌套调用和递归调用	(125)
6.2.4 指针与函数	(127)
6.2.5 系统函数	(131)
6.2.6 本章重点内容概括	(131)
6.3 典型例题分析及解答	(132)
6.4 自测试题	(138)
自测试题参考答案	(142)

第七章 结构型、共用型和枚举型

7.1 必考知识点及相互关系	(143)
7.2 重点与难点分析	(143)
7.2.1 结构型数据特点及定义方法	(143)
7.2.2 结构体变量	(145)
7.2.3 结构体数组	(147)
7.2.4 指向结构体类型数据的指针	(148)

7.2.5 函数间传递结构型数据的方法	(151)
7.2.6 共用体	(152)
7.2.7 枚举型数据	(155)
7.2.8 用户自定义数据类型的方法	(157)
7.2.9 本章重点内容概括	(158)
7.3 典型例题分析及解答	(159)
7.4 自测试题	(162)
自测试题参考答案	(166)

第八章 文件

8.1 必考知识点及相互关系	(168)
8.2 重点与难点分析	(168)
8.2.1 文件的概念	(168)
8.2.2 文件类型指针	(170)
8.2.3 文件的打开与关闭	(171)
8.2.4 文件的读写操作	(172)
8.2.5 文件的定位	(179)
8.2.6 文件错误测试函数	(181)
8.2.7 本章重点内容概括	(182)
8.3 典型例题分析及解答	(182)
8.4 自测试题	(187)
自测试题参考答案	(191)

第九章 编译预处理与带参数的主函数

9.1 必考知识点及相互关系	(193)
9.2 重点与难点分析	(193)
9.2.1 何为预处理命令	(193)
9.2.2 宏定义	(194)
9.2.3 文件包含	(197)
9.2.4 条件编译	(197)
9.2.5 带参数的主函数	(199)
9.2.6 本章重点内容概括	(200)
9.3 典型例题分析及解答	(201)
9.4 自测试题	(204)
自测试题参考答案	(206)

附 录

模拟试卷一	(208)
模拟试卷二	(216)
模拟试卷三	(224)
模拟试卷四	(231)
模拟试卷一参考答案	(238)
模拟试卷二参考答案	(241)
模拟试卷三参考答案	(243)
模拟试卷四参考答案	(245)

第一章 概 述

本章考核知识点与考核要求：

1. 程序、程序设计和高级语言,达到“识记”层次

- (1) 程序的概念；
- (2) 程序设计的任务和主要步骤；
- (3) 高级语言的概念与作用。

2. C 语言 C 程序,达到“识记”层次

- (1) C 语言的发展；
- (2) C 程序的基本结构。

3. C 语言的字符集、词类和句类,达到“领会”层次

- (1) C 语言的字符集；
- (2) C 语言的词的分类原则；
- (3) C 语言的保留字及其基本作用；
- (4) C 语言的语句分类。

1.1 必考知识点及相互关系

1. 程序、程序设计和高级语言,达到“识记”层次

(1) 程序的概念

• 程序：程序是对解决某个问题的方法步骤的描述；从计算机的角度来说，程序是用某种计算机能理解并执行的计算机语言对所要处理的数据以及处理的方法和步骤所作的完整而准确的描述。这个描述的过程就称为程序设计。对数据的描述就是指明数据采用何种数据类型；对处理方法和步骤的描述就是算法。由此，就有了对程序的更为简洁的定义：

程序 = 数据结构 + 算法。

(2) 程序设计的任务和主要步骤

• 程序设计：程序设计就是将分析解决问题的方法步骤记录下来的过程。

• 程序设计的步骤：分析问题并建立数学模型→算法设计→流程图设计→程序设计→程序调试→文档资料整理。

(3) 高级语言的概念与作用

• 二进制语言：亦称为机器语言，是使用二进制 0、1 代码编写程序的语言。是计算机能够直接识别和执行的计算机语言。是执行速度最快的语言，但是机器语言编写复杂，容易出错。

- 汇编语言：以助记符为特征的语言。

二进制语言和汇编语言依赖于机器(不同型号的计算机所使用的机器语言和汇编语言不同),不宜于推广。我们之为“面向机器的语言”和“低级语言”。

- 高级语言：亦称为“面向问题的语言”,是一种结构化的程序设计语言。用高级语言编写的程序称为“源程序”,它需要翻译成为二进制程序后才能执行。翻译过程有两种方式：一种是翻译一句执行一句,称为“解释”方式;另一种是全部翻译后再执行,称为“编译”方式,编译后的二进制程序称为“目标程序”。

计算机为什么能够自动有序地工作?

计算机是人制造的一种电子设备,它之所以能够自动地有序地进行数字运算和信息处理是因为在主存中已经存放有相应的程序,它依照程序中的指令执行相应的操作,达到数字运算和信息处理的目的。

计算机为什么能够处理各种各样的信息?

计算机将各种各样的信息,包括文字、数字、声音、图像等等,都用二进制的数字来表示。计算机处理代表二进制数字相应信号,即处理了相应的信息。所以信息的数量化是计算机处理各种各样信息的关键。

2. C 语言和 C 程序,达到“识记”层次

(1) C 语言的发展

① C 语言的产生与发展

C 语言是 1972 年由美国的 Dennis Ritchie 设计发明的,并首次在 DEC PDP-11 上重新编写了 UNIX 操作系统。它由早期的编程语言 BCPL(Basic Combind Programming Language)发展演变而来。在 1970 年,AT&T 贝尔实验室的 KenThompson 根据 BCPL 语言设计出较先进的并取名为 B 的语言,最后导致了 C 语言的问世。随着微型计算机的日益普及,出现了许多 C 语言版本。由于没有统一的标准,使得这些 C 语言之间出现了一些不一致的地方。为了改变这种情况,美国国家标准研究所(ANSI)为 C 语言制定了一套 ANSI 标准,成为现行的 C 语言标准。

② C 语言的特点

C 语言发展如此迅速,而且成为最受欢迎的语言之一,主要因为它具有强大的功能。许多著名的系统软件,如 DBASE III PLUS、DBASE IV 都是由 C 语言编写的。用 C 语言加上一些汇编语言子程序,就更能显示 C 语言的优势了,像 PC-DOS、WORDSTAR 等就是用这种方法编写的。归纳起来 C 语言具有下列特点:

- C 是中级语言,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。C 语言可以象汇编语言一样对位、字节和地址进行操作,而这三者是计算机最基本的工作单元。

- C 是结构式语言,结构式语言的显著特点是代码及数据的分隔化,即程序的各个部分除了必要的信息交流外彼此独立。这种结构化方式可使程序层次清晰,便于使用、维护以及调试。C 语言是以函数形式提供给用户的,这些函数可方便的调用,并具有多种循环、条件语句控制程序流向,从而使程序完全结构化。

- C 语言功能齐全,C 语言具有各种各样的数据类型,并引入了指针概念,可使程序效率更高。另外,C 语言也具有强大的图形功能,支持多种显示器和驱动器。而且计算功能、逻辑判断

功能也比较强大,可以实现决策目的。

- C语言适用范围大,C语言还有一个突出的优点就是适合于多种操作系统,如DOS、UNIX,也适用于多种机型。

(3) Turbo C的产生与发展

Turbo C是美国Borland公司的产品,Borland公司是一家专门从事软件开发、研制的大公司。该公司相继推出了一套Turbo系列软件,如Turbo BASIC,Turbo Pascal,Turbo Prolog,这些软件很受用户欢迎。该公司在1987年首次推出Turbo C1.0产品,其中使用了全然一新的集成开发环境,即使用了一系列下拉式菜单,将文本编辑、程序编译、链接以及程序运行一体化,大大方便了程序的开发。1988年,Borland公司又推出Turbo C1.5版本,增加了图形库和文本窗口函数库等,而Turbo C2.0则是该公司1989年出版的Turbo C2.0在原来集成开发环境的基础上增加了查错功能,并可以在Tiny模式下直接生成.COM(数据、代码、堆栈处在同一64KB内存中)文件。还可对数学协处理器(支持8087/80287/80387等)进行仿真。Borland公司后来又推出了面向对象的程序软件包Turbo C++,它继承发展Turbo C2.0的集成开发环境,并包含了面向对象的基本思想和设计方法。1991年为了适用Microsoft公司的Windows 3.0版本,Borland公司又将Turbo C++作了更新,即Turbo C的新一代产品Borland C++也已经问世了。

(2) C程序的基本结构

① C程序由若干个函数构成,其中必须有且仅有一个主函数。C程序的执行从主函数开始,在主函数结束。主函数的位置可以是任意的。主函数可以调用非主函数,非主函数之间可以相互调用,但不能调用主函数。

② 每个函数的定义分为两部分:函数说明部分和函数体。

函数说明部分的格式为:

返回值的类型 函数名(形式参数1,形式参数2,...)

形式参数的说明

函数体的格式为:

{

变量定义部分

功能语句

}

其中,每条语句结束时,用“;”作为结束。

用“/*”和“*/”括住的字符为注释语句。注释语句可以出现在程序的任意位置。程序编译时,注释将不参加编译,也不会出现在目标程序中。

3. C语言的字符集、词类和句类,达到“领会”层次

(1) C语言的字符集

C语言的字符集就是ASCII字符集,是用来书写源程序清单时允许出现的所有字符的集合。分为四类:

- 大小写英文字母
- 数字

- 键盘符号
- 转义字符

(2) C 语言的词的分类原则

C 语言的词由字符构成,主要分为 6 大类:

- 常量
- 变量
- 运算符
- 函数调用
- 表达式
- 保留字

(3) C 语言的保留字及其基本作用

C 语言的保留字是有特定用途、特定含义的英文单词。共有 32 个,用户在给常量、变量、函数等对象起名称时,不得使用这些保留字。

(4) C 语言的语句分类

C 语言的语句由词构成,C 语言的基本语句包括:

- 数据定义语句
- 赋值语句
- 函数调用语句
- 表达式语句
- 流程控制语句
- 复合语句
- 空语句
- 其他语句

对于每种语句来说,有规定的书写格式,称为“语法结构”;有规定完成的操作,称为“语法功能”。语句是用来构造程序的,用户可根据语句的功能按照语法结构书写程序。

1.2 重点与难点分析

1. 标识符

标识符是用户定义的一种字符序列。C 语言规定,标识符是“由字母或下划线”开头的字母、数字、下划线组成的字符串,最多 32 个字符(通常 PC 机中前 8 个字符有效)。以下列举几个正确的和不正确的标识符名称:

正确的	不正确的	错误原因
count	6count	由于 6count 是以数字开头,所以错误。
tall	ta! ll	由于 ta! ll 中包含非法字符“!”,所以错误。
test	test l	由于 test l 中包含有空格,所以错误。

在 C 语言中,大小写字母是有区别的,所以,count、Count 和 COUNT 是三个不同的标识符,代

表不同的含义。

标识符命名时要注意两点：

(1) 要合法,只能“由字母或下划线”开头,后续字符可以是字母、数字、下划线;在 PC 机中使用长度不能超过 8 个字符,不能使用 C 语言规定的保留字。命名时必须满足此条件。

(2) 要做到望名知意,以便从标识符看出所标识的对象。这一点仅仅是为了增加程序的可读性,C 语言的语法并不要求必须做到望名知意。例如:学生编号可以命名为:stu-no 或 xsbh,同时如果命名为:teacher、worker 或其他合法而不能表义名字也是可以的,但是程序的可读性就比较差。

2. 转义字符

转义字符是由“反斜杠字符(\)”开始后跟单个字符或若干个字符组成的。通常用来表示键盘上的控制代码或特殊符号,例如:语句

```
main()
{
    printf( " ab\tcd\nef\t " );
    printf( " ab   \bcd   \b\bef\n " );
}
```

请注意程序中的“转义字符”,第 1 个 printf 函数先在第 1 行左端开始输出“ab”,然后遇到“\t”,它代表一个“Tab”符号,其作用是“跳到下一个制表位置”,通常系统中一个“制表位”占 8 列;下一个制表位置从第 9 列开始,所以在第 1 行第 9~10 列上输出“cd”;下面是“\n”,它代表一个“回车换行符号”,其作用是“使当前位置移到下一行的开头”,然后输出“ef”;后面的“\t”使当前位置跳到第 9 列;第 2 个 printf 函数先在当前位置输出“ab”然后遇到“\b”,它代表一个“左退一格符号 backspace”,其作用是“退一格”;故在“ab”和“cd”之间仅有一个空格,而不是两个空格;同时在“cd”和“ef”之间没有空格,cd 后的两个“\b\b”向后退了两格。

程序输出的结果为:

```
ab      cd
ef      ab cdef
```

3. 重点记忆

(1) 转义字符及其作用(10 个),见表 1-1。

表 1-1 转义字符及其作用

字符形式	作 用
\n	换行,将当前位置移到下一行开头
\t	跳至下一个制表符位置
\b	退格,将当前位置移到前一位
\r	回车,将当前位置移到本行开头
\f	换页,将当前位置移到下页开头
\\	反斜杠字符“\”
\'	单撇号字符“'”
\"	双撇号字符“”
\ddd	1~3 位 8 进制数所对应的字符
\xhh	1~2 位 16 进制数所对应的字符

(2) C 语言的保留字(32 个)主要用于数据类型、存储类型的定义和构成语句。

① 数据类型标识符(15 个):

char	字符型	int	整型	long	长整型
short	短整型 s	Signed	有符号	unsigned	无符号
float	单精度实型	double	双精度实型	struct	结构型
union	共用型	enum	枚举	typedef	类型定义
const	常量说明符	volatile	易变量说明符		
void	空类型(1,用于明确的表示一个函数不返回任何值; 2,用于定义一个空类型的指针)				

② 存储类型标识符(4 个):

auto	自动型	extern	外部型
register	寄存器型	static	静态型

③ 用于构成语句(12 个):

break	中止	continue	继续	goto	跳转
return	返回	if	如果	else	否则
switch	开关	case	情况	default	缺省
do	执行	while	当	for	对于

④ 运算符(1 个):

sizeof	长度运算符
--------	-------

1.3 典型例题分析及解答

例 1-1 C 源程序的组成单位是()。

答案: 函数

例 1-2 一个 C 源程序中至少应包括一个()。

答案: 主函数(main 函数)

例 1-3 关于 C 语言,下列说法中正确的是()。

- A. 主函数名是由程序设计人员按照“标识符”的规则选取的。
- B. 分号和回车符号都可以作为一个语句的结束符号。
- C. 程序的执行,总是可以从源程序的第一行开始。
- D. 再程序清单的任意位置都可以插入任意个空格。

分析:

对于 A,主函数名规定是 main,不可以由程序设计人员自由选取;

对于 B,分号是表示语句结束的唯一符号;

对于 C,程序的执行总是从主函数的第一条语句开始,并从主函数结束的。

对于 D,正确

答案: D

例 1-4 C 语言提供的合法的关键字是()。

- A. swicth B. CHAR
C. Case D. default

分析:

对于 A, 正确的应为: switch;

对于 B, 正确的应为: char;(不可以大写字符)

对于 C, 正确的应为: case;(不可以大写第一个字符)

对于 D, 正确

本题要求考生记住保留字表及其正确的使用方法。

答案: D

例 1-5 在 C 语言中,合法的字符常量是()。

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

分析:

字符常量是用两个单引号(')前后括住的单个字符;转义字符是由“反斜杠字符(\)”开始,后跟单个字符或若干个字符组成的,通常用来表示键盘上的控制代码或特殊符号。由转义字符可以组成字符常量,如: '\111 ','\x41 '等都是由转义字符构成的字符常量。由于“\”在转义字符中作为特殊符号使用,而“'”作为字符常量的标记,所以若用到这两个字符时,必须写成转义字符形式,即用 '\\ '和 '\ ' 来分别表示字符“\”和“'”,而不能直接写成 '\ '和 "'。本题要求考生记住转义字符表及其正确的使用方法。

对于 A, 形式错误;

对于 B, 正确;

对于 C, 引号中有两个字符;对于 D, 用双引号括住的表示字符串;

答案: B

例 1-6 在 C 语言中,以下不正确的转义字符是()。

- A. '\\ ' B. '\ ' C. '\0 ' D. '\024 '

分析: 同例 1-4

答案: D

例 1-7 在 C 语言中,请选出可作用户标识符的一组标识符。()

- A. do define IF B. a1 _ a2 _ 456 DO
C. Fo - abc Case D. 2a Do sizeof

分析: C 语言规定,标识符是“由字母或下划线”开头的字母、数字、下划线组成的字符串。长度不能超过 8 个字符,不能使用 C 语言规定的保留字。命名时必须满足此条件。

对于 A, do 是 C 语言的保留字;

对于 B, 正确;

对于 C, - abc,是以减号开头的;

对于 D, sizeof 是 C 语言的保留字;

答案: B

1. C 程序中主函数的位置是在程序的()。
A. 开头
B. 任意位置
C. 结尾
D. 两个函数之间
2. C 程序的书写格式要求()。
A. 一行上只能写一条语句,一条语句只能写在一行上
B. 一行上可以写多条语句,但一条语句只能写在一行上
C. 一行上可以写多条语句,一条语句也可以写在多行上
D. 一行上只能写一条语句,但一条语句可以写在多行上
3. 以下叙述不正确的是()。
A. 在对一个 C 程序进行编译的过程中,可以发现注释中的拼写错误
B. 在 C 程序中主函数可以放在程序任意位置
C. C 语言本身没有输入输出语句
D. C 程序的基本组成单位是函数
4. 以下叙述正确的是()。
A. 在 C 程序中注释语句只能位于一段程序的开始。
B. 在 C 程序中注释语句只能位于某个语句的后面。
C. 一个 C 语言程序是由一个主程序和若干子程序组成的。
D. 程序编译时,注释语句不参加编译,也不会出现在目标程序中。
5. 下列正确的标识符是()。
A. friend
B. 9-number
C. pen box
D. tc.exe
6. C 语言中的标识符只能由字母、数字、下划线三种字符组成,且第 1 个字符为()。
A. 必须是下划线
B. 必须是字母
C. 必须是字母或下划线
D. 可以是字母、数字、下划线中任意一字符
7. 下面四个选项中,均是 C 语言关键字的选项是()。
A. void while type
B. auto union volatile
C. float break go
D. return if scanf
8. 下面四个选项中,均不是 C 语言关键字的选项是()。
A. else const getc
B. static struct if
C. define printf go
D. for enum do
9. 下面四个选项中,不是合法转义字符的选项是()。
A. \\
B. \c
C. \101
D. \xf

10. 下面四个选项中,是合法转义字符的选项是()。

A. \f

B. \abc

C. \a

D. \8

二、填空题

1. 程序设计就是_____问题的方法步骤,并将其记录下来的过程。

2. 在 C 程序清单的_____处可以插入空格符号或回车符号。

3. 在 C 程序中主函数的函数名规定为_____。

4. 可以用“_____”和“_____”括住的任意字符,称为“注释”。注释可以出现在程序的_____位置。

5. C 语言中的标识符只能由_____、_____和_____三种字符组成。

自测试题参考答案

一、单项选择题

1. B

2. C

3. A

4. D

5. A

6. C

7. B

8. C

9. B

10. C

二、填空题

1. 分析解决

2. 任何一处

3. Main

4. / * * / 任意

5. 字母 数字 下划线

第二章 基本数据类型和运算

在“高级语言程序设计”课程中,本章是全书的基础、全书的重点。基础知识的学习是最重要的,有了基础知识才能够做题和编写程序。因此需要考生认真学习。

本章考核知识点与考核要求:

1. 数据类型的概念及其分类,达到“识记”层次

- (1) 数据类型的概念;
- (2) C语言中数据类型的分类。

2. 基本类型(整型、实型、字符型、字符串)常量的书写方法,达到“领会”层次

- (1) 整型(短整、长整、无符号短整、无符号长整)常量;
- (2) 实型(单精度、双精度)常量;
- (3) 字符常量;
- (4) 字符串常量。

3. 基本类型(整型、实型、字符型)变量的定义、初始化方法,达到“领会”层次

- (1) 定义整型(短整、长整、无符号短整、无符号长整)变量,并对其初始化;
- (2) 定义实型(单精度、双精度)变量,并对其初始化;
- (3) 定义字符型变量,并对其初始化。

4. 基本类型数据在内存中的存放方式,达到“识记”层次

- (1) 整型(短整、长整、无符号短整、无符号长整)数据在内存中的存放方式;
- (2) 字符型数据在内存中的存放方式;
- (3) 字符串常量在内存中的存放方式。

5. 变量的存储类型及其定义方法,达到“领会”层次

- (1) 变量的存储类型概念;
- (2) 不同存储类型变量的存储与使用特点;
- (3) 变量存储类型的定义方法。

6. 变量的生存期和作用域,达到“领会”层次

- (1) 内部变量和外部变量的概念和定义方法;
- (2) 通过变量的定义方式确定变量的生存期(全局变量和局部变量);
- (3) 通过变量的定义方式确定变量的作用域。

7. 基本运算符的运算规则和优先级别,达到“简单应用”层次

- (1) 算术运算符的运算对象、运算规则和优先级;
- (2) 关系和逻辑运算符的运算对象、运算规则和优先级;