

C 语言程序设计

基础配套练习与检测

黄文胜 主编

重庆大学出版社



C 语言程序设计基础配套练习与检测

主 编 黄文胜

重庆大学出版社

图书在版编目（CIP）数据

C语言程序设计基础配套练习 / 黄文胜主编.—2版.

—重庆：重庆大学出版社，2017.2

ISBN 978-7-5689-0391-2

I . ① C… II . ①黄… III . ① C 语言—程序设计—中等

专业学校—习题集 IV. ①TP312.8-44

中国版本图书馆CIP数据核字（2017）第012541号

C语言程序设计基础配套练习（第二版）

主 编 黄文胜

副主编 廖冬梅

责任编辑：陈一柳 版式设计：李南江

责任校对：谢 芳 责任印制：张 策

*

重庆大学出版社出版发行

出版人：易树平

社址：重庆市沙坪坝区大学城西路21号

邮编：401331

电话：（023）88617190 88617185（中小学）

传真：（023）88617186 88617166

网址：<http://www.cqup.com.cn>

邮箱：fxk@cqup.com.cn（营销中心）

全国新华书店经销

万州日报印刷厂印刷

*

开本：787mm×1092mm 1/16 印张：9.5 字数：234千

2017年2月第2版 2017年2月第6次印刷

印数：1—3 000

ISBN 978-7-5689-0391-2 定价：19.00元

本书如有印刷、装订等质量问题，本社负责调换

版权所有，请勿擅自翻印和用本书

制作各类出版物及配套用书，违者必究

进入21世纪，随着计算机科学技术的普及和发展加快，社会各行业的建设和发展对计算机技术的要求越来越高，计算机已成为各行各业不可缺少的基本工具之一。在今天，计算机技术的使用和发展，对计算机技术人才的培养提出了更高的要求，培养能够适应现代化建设需求的、能掌握计算机技术的高素质技能型人才，已成为职业教育人才培养的重要内容。

按照“以就业为导向”的办学方向，根据国家教育部中等职业教育人才培养的目标要求，结合社会行业对计算机技术操作型人才的需要，我们在调查、总结前些年计算机应用型专业人才培养的基础上，重新对计算机专业的课程设置进行了调整，进一步突出专业教学内容的针对性和实效性，重视对学生计算机基础知识的教学和对计算机技术操作能力的培养，使培养出来的人才能真正满足社会行业的需要。为进一步提高教学的质量，我们专门组织了有丰富教学经验的教师和有实践经验的行业专家，重新编写了这套中等职业学校计算机专业教材。

本套教材编写采用了新的教育思想、教学观念，遵循的编写原则是“拓宽基础、突出实用、注重发展”。为满足学生对计算机技术学习的需求，力求使教材突出以下几个主要特点：一是体现以学生为本，针对目前职业学校学生学习的实际情况，按照学生对专业知识和技能学习的要求，教材在编写中注意了语言表述的通俗性，以任务驱动的方式组织教材内容，以服务学生为宗旨，突出学生对知识和技能学习的主体性；二是强调教材的互动性，根据学生对知识接受的过程特点，重视对学生探究能力的培养，教材编写采用了以活动为主线的方式进行，把学与教有机结合，增加学生的学习兴趣，让学生在教师的帮助下，通过对活动的学习而掌握计算机技术的知识和操作的能力；三是重视教材的“精、用、新”，根据各行各业对计算机技术使用的需要，在教材内容的选择上，做到“精选、实用、新颖”，特别注意反映计算机的新知识、新技术、新水平、新趋势的发展，使所学的计算机知识和技能与行业需要相结合；四是编写的体例和栏目设置新颖，易受到中职学生的喜爱。这套教材实用性和操作性较强，能满足中等职业学校计算机专业人才培养目标的要求，也能满足学生对计算机专业技术学习的不同需要。

为了便于组织教学，我们将根据计算机专业技术发展的要求和教学的实际需要，研究开发出与教材配套的有关教学资源材料供大家参考和使用，进一步提高教学的实效性。希望重新推出的这套教材能受到广大师生喜欢，为职业学校计算机专业的发展做出贡献。

中等职业学校计算机专业教材编写组
2015年7月



C YUYAN CHENGXU
SHEJI JICHU

XUYAN

序言

本书是以重庆市中等职业学校计算机类专业的基础课程《C语言程序设计基础》的课程标准为准绳，以重庆大学出版社出版的《C语言程序设计基础教程》一书为基础，在重庆市中等职业技术教育信息技术类专业教学指导委员会专家指导下编写的配套练习手册。作者团队成员或参与《C语言程序设计基础教程》教材的编写，或是多年执教中职C语言程序设计基础课程的教师，他们具有丰富的C语言运用和教学经验，为本书内容质量提供了有力的保障。

本书提供了每个模块的针对练习，每个单元的诊断检测和全书内容的综合练习与目标检测，并设计了多种题型，如填空题、选择题、判断题、程序填空题、阅读程序题、编程题、按要求写表达式等，以期通过多维度对知识点进行强化练习来达到巩固学习成效的目的。每个单元前罗列了本单元的学习目标，有利于学习者合理安排时间和努力方向。本书以掌握C语言的基础知识和提高编程应用能力为目标，无偏、难、怪题，不纠缠于知识点的理论阐释，而着重于知识点的实际应用。使用者要注意，由于C语言有多种版本的编译器，同样的代码经不同编译器编译后程序的执行结果或存在差异，本书中所有程序及代码的结果都是在C-Free下运行的结果。

本书由重庆市商务学校黄文胜担任主编，黄文胜编写本书中第一、三单元的相关练习与检测题和全书的综合练习题与目标检测题，廖冬梅编写第二、六单元的练习与检测题，张琴编写第四、五单元的练习与检测题。囿于编者水平和教学经验，以及排版中出现失误，书中恐有错漏出现，在此先致歉意，并真诚地请把信息反馈给我们，以改进我们的工作，联系电子邮箱为：hungws@21cn.com。

编 者

2016年8月



C YUYAN CHENGXU
SHEJI JICHU

QIANYAN

前言



C YUYAN CHENGXU SHEJI JICHU
PEITAO LIANXI

MULU

目 录

第一单元 C语言基础	001
模块练习 模块一 / C语言程序的结构	002
模块练习 模块二 / C语言程序的基本数据对象	003
模块练习 模块三 / 数据运算和表达式	004
模块练习 模块四 / 在程序中输入输出数据	007
模块练习 模块五 / 算法的表示	012
单元综合练习 / C语言基础综合练习1	013
单元综合练习 / C语言基础综合练习2	017
单元检测题 / C语言基础单元检测题1	021
单元检测题 / C语言基础单元检测题2	024
第二单元 程序流程控制	028
模块练习 模块一 / 顺序程序设计	029
模块练习 模块二 / 分支程序设计	032
模块练习 模块三 / 循环程序设计	037
单元综合练习 / 程序流程控制综合练习1	041
单元综合练习 / 程序流程控制综合练习2	044
单元检测题 / 程序流程控制单元检测1	048
单元检测题 / 程序流程控制单元检测2	052
第三单元 构造数据对象	056
模块练习 模块一 / 数组	057

模块练习 模块二 / 结构类型	061
模块练习 模块三 / 枚举	064
模块练习 模块四 / 指针	066
单元检测题 / 单元检测题1	069
单元检测题 / 单元检测题2	072
第四单元 实现模块化程序设计	076
模块练习 模块一 / 函数	077
模块练习 模块二 / 变量的作用域	081
单元综合练习 / 实现模块化程序设计综合练习1	086
单元综合练习 / 实现模块化程序设计综合练习2	090
单元检测题 / 实现模块化程序设计单元检测题1	094
单元检测题 / 实现模块化程序设计单元检测题2	098
第五单元 文件操作	102
模块练习 模块一、二 / 文件概述及使用文件	103
单元综合练习 / 文件操作综合练习1	106
单元综合练习 / 文件操作综合练习2	110
单元检测题 / 文件操作单元检测题1	114
单元检测题 / 文件操作单元检测题2	118
第六单元 程序设计实践	122
模块练习 模块一 / 判定数的特性	123
模块练习 模块二 / 数据的统计	125
模块练习 模块三 / 排序	128
模块练习 模块四 / 查找	130
单元综合练习 / 程序设计实践综合练习1	132
单元综合练习 / 程序设计实践综合练习2	136
单元检测题 / 程序设计实践单元检测题1	139
单元检测题 / 程序设计实践单元检测题2	143

C语言基础

知识内容概述

本单元主要描述了用C语言实现程序设计的基础知识，包括C语言源程序的基本组成结构，上机执行C程序的方法和步骤；在程序表达数据，数据处理方法和相关的规则；如何在程序中输入输出数据以及算法的表示。这些内容是实现程序编写的基础。

教学目标

知识要点	了 解	理 解	掌 握	运 用
源程序结构	√			
C语言程序的组成元素			√	
上机执行C程序方法			√	√
数据的概念及分类		√		
书写各类常量			√	
变量的概念与使用		√	√	√
表达式及表达式的值	√			
计算表达式的值			√	√
给变量赋值		√	√	√
计算自增自减表达式			√	√
比较数的大小		√	√	√
执行逻辑运算		√	√	√
计算逗号表达式	√			
输入数据		√	√	√
输出数据		√	√	√
C语言语句的种类与作用	√			
算法及表示方法	√			
算法的流程图表示			√	
基本流程结构		√		
结构化程序设计方法	√			

[模块练习 模块一]

NO.1

C语言程序的结构

一、填空题

1. 源程序是指_____程序代码，它必须经过_____或_____转变成用_____表示的_____才能在计算机上运行。
2. C语言规定了一套在程序设计时可以使用的符号，它们是_____、_____、_____。
3. 标识符是指用来标识_____、_____、_____、_____等程序对象名称的有效_____。
4. C语言的标识符只能由_____、_____和_____组成，且第一字符必须是_____或_____。
5. C语言中对一些标识符规定了专门的用途，如int, if, while, case等，它们（能、不能）_____用作用户标识符，这些标识符被称为_____。
6. C语言源程序是由_____组成的，人们称C语言为_____的程序设计语言。其中必须包含一个且只有一个名为_____的函数，被称为_____。
7. 一个函数由_____和_____两部分组成。
8. 在计算机上运行C程序经过_____、_____、_____、_____4个步骤。
9. C语言的注释除用以提高程序的阅读性外，还可以用于_____。
10. 在C-Free中，按_____键编译源程序，按_____键直接运行程序。

二、判断题

- | | |
|------------------------------|-----|
| 1. C语言源程序中的一个程序行就是一个语句。 | () |
| 2. /*...*/标注的注释将降低程序的执行效率。 | () |
| 3. C语言程序中的标识符是程序对象的名字。 | () |
| 4. 运行C语言程序时，总是从程序的第一行语句开始执行。 | () |
| 5. C语言中要区分字母的大小写。 | () |
| 6. 标识符不可以使用大写字母。 | () |
| 7. 编译后的C语言程序就可以执行了。 | () |
| 8. 在C-Free中，按F11开始调试程序。 | () |
| 9. 命名标识符时，首字符不能是数字。 | () |
| 10. C语言中不能使用汉字。 | () |

三、选择题

- 1.下面可作为标识符首字符的是（ ）。
- | | | | |
|-----|-----|-----|-----|
| A.总 | B.X | C.8 | D.# |
|-----|-----|-----|-----|

- 2.在C-Free中把源程序从编译、连接和运行顺次连续执行的快捷键是（ ）。
- A.F9 B.F5 C.F11 D.F10
- 3.下面合法的用户标识符是（ ）。
- A.Long B.长度 C.100m D.for
- 4.下面关于C语言叙述正确的是（ ）。
- A.C语言源程序中有且只有一个main函数
- B.C源程序中每一行可以写多个语句，它们之间用冒号分隔
- C.一个C语句以句号结束
- D.C语言程序执行完源程序中的最后一个语句后结束
- 5.C语言源程序编译后生成的目标文件扩展名是（ ）。
- A.C B.obj C.exe D.lib

[模块练习 模块二]

NO.2

C语言程序的基本数据对象

一、填空题

1. 定义值为2012的符号常量VER的命令是_____。
2. 变量包含两个方面的内容是_____和_____。一个变量对应计算机的_____, 在C语言中使用变量必须遵守_____原则, 定义变量的格式为_____。
3. C语言提供的常用基本数据类型有_____, _____, _____。
4. 整型数据类型包括_____, _____, _____3种。
5. 实型常量的十进制小数形式要求小数点两边_____, 而指数形式中E的前边必须有_____, E的后边的数必须为_____。
6. 字符常量是用_____括起来的_____个字符。在C语言中一些不能直接输入的字符采用_____来表示, 它是以_____字符开头的字符序列, 换行符表示为_____。
7. 字符串常量是用_____括起来的_____, 字符串的末尾有一个表示字符串终结的字符_____, 称之为_____。
8. 变量名是变量对应内存单元的_____, 变量名在程序中代表_____。
9. 基本整型数在内存中占_____字节的存储空间, 可表示的整数范围是_____。
10. “=”读作_____, 其作用是_____。

二、判断题

- 1.在C语言中所有字符都可以表示成转义字符的形式。 ()
- 2.类型void表示不是C语言支持的任何数据类型。 ()

- 3.C语言中的字符采用ASCII编码,可表示128个字符。()
- 4.数据30000比30占用更多的存储空间。()
- 5.在C语言中32768是长整型数。()
- 6.在程序中使用控制字符和非显字符只能以转义字符形式书写。()
- 7.'\112'是错误的字符表示。()
- 8.变量所分配的内存空间取决于其存储的数据大小。()
- 9.未经过初始化的变量中没有存储数据。()
- 10.定义变量主要就是声明变量的数据类型,以便内存分配。()

三、选择题

- 1.下面正确的整型常量是()。
A.009 B.x789 C.2e3 D.900
- 2.下面不正确的实型常量是()。
A.12.50 B..625 C.345. D.2.5e3.8
- 3.下面合法的标识符是()。
A._123 B.int C.6pin D.xrc-1
- 4.将字符g赋值给字符变量ch,正确的表达式是()。
A.ch="g" B.ch=71 C.ch='\0147' D.ch='\147'
- 5.下面对变量的定义正确的是()。
A.int x, y B.fk: float; C.char ch; D.double int;
- 6.若计算s=10!,则变量s的数据类型应定义为()。
A.int B.long C.float D.double
- 7.给变量x赋初值正确的是()。
A.x=10 B.10=x C.x==10 D.10==x
- 8.下面定义符号常量PI正确的是()。
A.#define PI=3.14 B.#define PI=3.14; C.#define PI 3.14 D.#define PI 3.14;

四、定义并初始化变量

- 1.定义3个长整型变量lp、lq、lh。
- 2.定义2个双精度实型变量dx、dy,并为dx设置初值为0.000000351。
- 3.定义一个字符变量cc,设其初值为换行符。

[模块练习 模块三]

NO.3

数据运算和表达式

一、填空题

1. 运算符是用于表示_____的符号或符号的组合。
2. 运算符两侧操作数的数据类型必须_____。

- ## 二、判断题

- 1.运算符%的操作数只能是整数。()
- 2.字符与整数之间不能进行运算。()
- 3.进行强制数据类型转换时，不会改变被转换数据的类型。()
- 4.字符和整数相加减的结果是另一个字符。()
- 5.C语言中任何类型的数据都可以参加逻辑运算。()
- 6.不能给变量赋值不同类型的数据。()
- 7.逗号表达式的值是所有子表达式的值之和。()
- 8.赋值运算符*=的优先级高于+=。()
- 9.表达式x++的值与x--的值相同。()
- 10.表达式m=n=9是把9分别赋值给变量m和n。()

三、选择题

- 1.以下符合C语言语法的赋值表达式是()。
- A.a=9+b+c=d+9 B.a=(9+b, c=d+9)
- C.a=9+b, b++, c+9 D.a=9+b+++=c+9
- 2.若有代数式 $\frac{3ad}{cd}$, 则不正确的C语言表达式是()。
- A.a/c/d*b*3 B.3*a*b/c/d C.3*a*b/c*d D.a*b/d/c*3
- 3.已知各变量的类型说明如下, 则以下符合C语言语法的表达式是()。
- int m=8, n, a, b;
- unsigned long w=10;
- double x=3.14, y=0.12;
- A.a+=a-= (b=2) * (a=8) B.n=n*3=18

C.x%3

D.y=float(m)

4.已知字母A的ASCII码为十进制数65,且S为字符型,则执行语句S='A'+6-'3';后,S中的值为()。

A.'D'

B.68

C.不确定的值

D.'C'

5.在C语言中,要求运算数必须是整型的运算符是()。

A./

B.++

C.*=

D.%

6.若有说明语句:char s='\72';则变量s()。

A.包含一个字符

B.包含两个字符

C.包含三个字符

D.说明不合法,s的值不确定

7.若有定义:int m=7; float x=2.5, y=4.7;;则表达式x+m%3*(int)(x+y)%2/4的值是()。

A.2.500000

B.2.750000

C.3.500000

D.0.000000

8.设变量x为float类型,m为int类型,则以下能实现将x中的数值保留小数点后两位,第三位进行四舍五入运算的表达式是()。

A.x=(x*100+0.5)/100.0

B.m=x*100+0.5, x=m/100.0

C.x=x*100+0.5/100.0

D.x=(x/100+0.5)*100.0

9.表达式13/3*sqrt(16.0)/8的数据类型是()。

A.int

B.float

C.double

D.不确定

10.设以下变量均为int类型,则值不等于7的表达式是()。

A.(m=n=6, m+n, m+1)

B.(m=n=6, m+n, n+1)

C.(m=6, m+1, n=6, m+n)

D.(m=6, m+1, n=m, n+1)

11.假设所有变量均为整型,则表达式(x=2, y=5, y++, x+y)的值是()。

A.7

B.8

C.6

D.2

12.已知s是字符型变量,下面不正确的赋值语句是()。

A.s='\012';

B.s='u+v';

C.s='1'+2';

D.s=1+2;

13.已知s是字符型变量,下面正确的赋值语句是()。

A.s='abc';

B.s='\08';

C.s='\xde';

D.s='\';

14.若有以下定义,则正确的赋值语句是()。

int x, y;

float z;

A.x=1, y=2,

B.x=y=100

C.x++;

D.x=int(z);

15.设x, y均为float型变量,则不正确的赋值语句是()。

A.++x;

B.x*=y-2;

C.y=(x%3)/10;

D.x=y=0;

16.下列语句中符合C语言的赋值语句是()。

A.a=7+b+c=a+7; B.a=7+b++=a+7; C.a=7+b, b++, a+7 D.a=7+b, c=a+7;

四、按要求写表达式

$$1. y = \frac{a(x-b)}{b-c} + 6x$$

$$2. y = 3x^2 - 7x + 5$$

- 3. $y = \frac{a+b}{cd}$
- 4. $y = \frac{1}{2} \left(ax + \frac{a+x}{4a} \right)$
- 5. x 的绝对值大于36
- 6. alp是字母
- 7. year是闰年
- 8. cc是空白字符
- 9. m 是个位数是7的能被3整除的整数
- 10. n 不是奇数

五、计算表达式的值

- 1. int a=23, b=6; 表达式a-a/b*b+a%b
- 2. int m=6; 表达式m+=m*=m-2
- 3. int x=9, y; 表达式y=x--+1
- 4. int a, b; a=b=1, a++, b-=10, a+b
- 5. char ch='A'; 表达式ch+=32
- 6. char c1, c2; 表达式c1='9', c2='0', c1-c2
- 7. float x; 表达式x=7/2
- 8. int m=19, n=5; float y=17.99; 表达式m/n+ (int) (y/2+0.5)

[模块练习 模块四]

NO.4

在程序中输入输出数据

一、填空题

- 1. 在C语言中基本的输入设备是_____，输出设备是_____。
- 2. scanf () 的参数由_____和_____两个部分构成。第一部分是_____组成的字符串，用于把输入的字符序列转换成需要类型的数据，第二部分是_____组成的输入列表。
- 3. 连续输入多个数值型数据时，数据之间用_____分隔。
- 4. 格式转换说明符的作用是_____，如%d的作用是_____。
- 5. 变量pn的地址是_____，_____、_____不能进行地址运算。
- 6. 格式转换说明符要与输入地址列表中的变量_____。
- 7. scanf () 格式控制字符串中的普通字符需要_____，printf () 格式控

制字符串中的普通字符会_____。

8. 为了在程序使用输入/输出函数,需要在源程序开始处写上_____预处理命令。

9. printf () 输出实数时默认保留_____位小数,不足位时用_____补全,超过则_____。

10. 要正确输出长整型数,对应的格式转换说明符应是_____。

二、判断题

1. 不论是输入什么类型的数据,其最初形式都是一个字符序列。 ()
2. 连续输入多个字符数据时,默认用空格分隔。 ()
3. scanf () 函数中格式控制字符串中的普通字符可起提示作用。 ()
4. getchar () 函数是字符专用输入函数,格式为getchar (字符变量)。 ()
5. 格式转换说明符在遇到不能识别的字符时,停止数据转换。 ()
6. %d 可以对应字符变量,输出的是字符的ASCII码。 ()
7. 字符数据在输出时,要去掉定界符。 ()
8. printf () 输出数据后自动换行。 ()
9. 建议不要在scanf () 的格式控制串中使用普通字符。 ()
10. putchar ("A") 的功能是输出字符A。 ()

三、选择题

1. putchar函数可以向终端输出一个 ()。

A. 整型变量表达式值 B. 字符串 C. 实型变量值 D. 字符或字符型变量值
2. 以下程序段的输出结果是 ()。


```
int a=12345; printf ("%2d\ n", a);
```

A. 12 B. 34 C. 12345 D. 提示出错、无结果
3. 若x和y均定义为int 型,z定义为double型,以下不合法的scanf () 函数调用语句为 ()。

A. scanf ("%d%lx, %le", &x, &y, &z); B. scanf ("%2d*%d%lf ", &x, &y, &z);

C. scanf ("%x%*do", &x, &y); D. scanf ("%x%o%6.2f", &x, &y, &z);
4. 有如下程序段:


```
int x1, x2;
char y1, y2;
scanf ("%d%c%d%c", &x1, &y1, &x2, &y2);
```

若要求x1, x2, y1, y2的值分别为10, 20, A, B, 正确的数据输入是 ()。

A. 10A □ 20B B. 10 □ A20B C. 10 □ A □ 20 □ B D. 10A20 □ B
5. 若变量已正确说明为float类型,要通过语句scanf ("%f %f %f ", &a, &b, &c); 给a赋予10.0, b赋予22.0, c赋予33.0, 不正确的输入形式为 ()。

A.10<回车> B.10.0, 22.0, 33.0<回车>

22<回车>

33

C.10.0<回车>

D.10 22<回车>

22.0 33.0<回车>

33<回车>

6.有如下程序,若要求x1, x2, y1, y2的值分别为10, 20, A, B, 正确的数据输入是()。

```
int x1, x2;
```

```
char y1, y2;
```

```
scanf ("%d%d", &x1, &x2);
```

```
scanf ("%c%c", &y1, &y2);
```

A.1020AB

B.10 20 ABC

C.10 20<回车>

D.10 20AB

AB

7.已有定义int a=-2; 和输出语句: printf ("%8lx", a); , 以下正确的叙述是()。

A.整型变量的输出格式符只有%d一种

B.%x是格式符的一种, 它可以适用于任何一种类型的数据

C.%x是格式符

D.%8lx不是错误的格式符, 其中数字8规定了输出字段的宽度

8.有如下程序段, 对应正确的数据输入是()。

```
float x, y;
```

```
scanf ("%f%f", &x, &y);
```

```
printf ("a=%f, b=%f", x, y);
```

A.2.04<回车>

B.2.04, 5.67<回车>

5.67<回车>

C.A=2.04, B=5.67<回车>

D.2.055.67<回车>

9.有如下程序段, 从键盘输入数据的正确形式应是()。

```
float x, y, z;
```

```
scanf ("x=%d, y=%d, z=%d", &a, &y, &z);
```

A.123

B.x=1, y=2, z=3

C.1, 2, 3

D.x=1 y=2 z=3

10.以下说法正确的是()。

A.输入项可以为一个实型常量, 如scanf ("%f", 3.5);

B.只有格式控制, 没有输入项, 也能进行正确输入, 如scanf ("a=%d, b=%5d");

C.当输入一个实型数据时, 格式控制部分应规定小数点后的位数, 如scanf ("%4.2f", &f);

D.当输入数据时, 必须指明变量的地址, 如scanf ("%f", &f);

11.根据定义和数据的输入方式, 输入语句的正确形式为()。

已有定义：float x, y;

数据的输入方式：1.23<回车>

4.5<回车>

A.scan ("%f", &x, &y);

B.scnaf ("%f%f", &x, &y);

C.scnaf ("%3.2f %2.1f", &x, &y);

D.scnaf ("%3.2f%2.1f", &x, &y);

12.根据下面的程序及数据的输入和输出形式，程序中输入语句的正确形式应该为（ ）。

```
#include "stdio.h"
```

```
main ( )
```

```
{ char s1, s2, s3;
```

```
    输入语句;
```

```
    printf ("%c%c%c", s1, s2, s3);
```

```
}
```

输入形式：A B C<回车>

输出形式：A B

A.scnaf ("%c%c%c", &s1, &s2, &s3);

B.scnaf ("%c % % c%c", &s1, &s2, &s3);

C.scnaf ("%c, %c, %c", &s1, &s2, &s3);

D.scnaf ("%c%c", &s1, &s2, &s3);

13.以下程序的执行结果是（ ）。

```
#include "stdio.h"
```

```
main ( )
```

```
{ int x=2, y=3;
```

```
    printf ("x=%d, y=%d\n", x, y);
```

```
}
```

A.x=%2, y=%3

B.x=%d, y=%d

C.x=2, y=3

D.x=%d, y=%d

14.以下程序的输出结果是（ ）。

```
#include "stdio.h"
```

```
main ( )
```

```
{ printf ("\nstring1=%15s*", "programming");
```

```
    printf ("\nstring2=%-5s*", "boy");
```

```
    printf ("string3=%2s*", "girl");
```

```
}
```

A.string1=programming
string2=boy*
string3=gi*

B.string1= programming*
string2=boy *string3=gi*

C.string1=programming
string2= boy*string3=girl*

D.string1= programming*
string2=boy *string3=girl*