

全国高等教育自学考试



# 面向对象程序设计 同步练习册

全国高等教育自学考试指导委员会/组编  
苏长龄 杨鲲/主编

全国高等教育自学考试指定教材  
计算机及应用专业

# 面向对象程序设计 同步练习册

全国高等教育自学考试委员会 组编  
主编 苏长龄 杨鲲

吉林大学出版社

## 图书在版编目(CIP)数据

面向对象程序设计同步练习册/苏长岭主编. —长春:吉林大学出版社,  
2001. 4

ISBN 7-5601-2504-2

I. 面... II. 苏... III. 面向对象语言—程序设计—高等学校  
—教学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字(2001)第 21188 号

### 面向对象程序同步练习册 (全国高等教育自学考试指定教材)

全国高等教育自学考试委员会 组编

主 编 苏长龄 杨 鲲

---

责任编辑、责任校对:吕健波

封面设计:

吉林大学出版社出版  
(长春市解放大路 125 号)

吉林农业大学印刷厂印刷

开本:787×1092 毫米 1/16

2001 年 4 月第 1 版

印张:12.5

2001 年 4 月第 1 次印刷

字数:200 千字

印数:1-5000 册

---

ISBN 7-5601-2504-2/TP·113

定价:16.00 元

凡购买我社的图书,如有缺页、倒页、脱页等质量问题者,请与当地图书销售部门联系调换;版权所有,不得翻印。

## 编者的话

面向对象程序设计课程是一门实践性很强的课程,除了上机实验还要做书面的编程练习,通过阅读、分析和编写程序,掌握面向对象程序设计的程序结构,理解程序设计语言的内在含义和联系,掌握在程序设计过程中分析问题和解决问题的方法。为了帮助考生更好地掌握教学内容,根据全国高等教育自学考试指导委员会的要求,编写了此书。

本书是学习《面向对象程序设计》课程的配套教材,结合考试大纲的要求,书中章节结构和习题与教材一致。针对各章的基本概念和重点内容及考生自学中的难点,每章选择了具有一定代表性的典型习题,通过学习会使考生加深理解和掌握教学内容,提高编程能力。各章容易混淆的概念和难点问题,已在《面向对象程序设计辅导教材》中通过实例给出了详尽说明,所以本书中的习题不再分析讲解,仅给出答案供参考。

参加本书的编写和程序调试工作的有苏长龄、杨鲲、唐宗明等。由于时间仓促,不当之处在所难免,恳请读者批评指正。

编者

2000年12月

# 目 录

<b>第 1 章 面向对象及 C++ 基础知识</b> .....	1
1.1 问答题 .....	3
1.2 选择题 .....	6
1.3 给出分析程序的结果 .....	18
<b>第 2 章 类和对象</b> .....	44
2.1 问答题 .....	45
2.2 选择题 .....	45
2.3 给出分析程序的结果 .....	49
<b>第 3 章 构造函数与析构函数</b> .....	69
3.1 问答题 .....	71
3.2 选择题 .....	71
3.3 给出分析程序的结果 .....	77
<b>第 4 章 继承和派生类</b> .....	94
4.1 问答题 .....	96
4.2 选择题 .....	96
4.3 给出分析程序的结果 .....	101
<b>第 5 章 多态性与虚函数</b> .....	114
5.1 问答题 .....	115
5.2 选择题 .....	115
5.3 给出分析程序的结果 .....	120
<b>第 6 章 进一步使用成员函数</b> .....	133
6.1 问答题 .....	134
6.2 选择题 .....	134
6.3 给出分析程序的结果 .....	140
<b>第 7 章 运算符重载及流类库</b> .....	163
7.1 问答题 .....	164
7.2 选择题 .....	164
7.3 给出分析程序的结果 .....	170
<b>第 8 章 模板</b> .....	185
8.1 选择题 .....	186
8.2 给出分析程序的结果 .....	187
主要参考文献 .....	193

# 第 1 章 面向对象及 C++ 基础知识

## 面向对象及 C++ 基础知识大纲

### (一) 课程内容

1. 面向对象程序设计基础知识。
2. 基本程序结构。
3. 类型修饰符。
4. 函数原型。
5. 内联函数。
6. 引用。
7. 动态内存分配。
8. 编译指令。
9. 改变习惯重新思考。

### (二) 自学要求

本章主要介绍面向对象设计的基础知识和 C++ 语言对 C 语言的基本改进部分。这些改进部分包括基本程序结构、类型修饰符、函数原型、内联函数、引用和编译指令等,并且讨论了如何改变习惯重新思考,利用已有知识学习 C++ 的问题。

本章的要求是理解面向对象程序设计的思想及 C++ 语言中的新思想,目的是为学习 C++ 的类打下基础。

### (三) 考核知识点及考核要求

1. 面向对象程序设计基础知识,要求达到“识记”层次。
  - 1.1 面向对象程序设计的思想。
  - 1.2 C++ 语言与 C 语言的关系。
2. 基本程序结构,要求达到“综合应用”层次。
  - 2.1 程序的基本结构和作用,正确书写。
  - 2.2 通过对比 C 语言,认识 C++ 的注释、换行、标准输入及输出语句。
  - 2.3 头文件及常量的定义。
  - 2.4 不使用宏来定义常量的意义。
  - 2.5 主函数的基本作用。
3. 类型修饰符,要求达到“领会”层次。
  - 3.1 const 修饰符的使用方法。
  - 3.2 const 修饰符的作用。
4. 函数原型,要求达到“综合应用”层次。
  - 4.1 函数原型的作用及意义。

- 4.2 函数原型的使用方法。
- 5. 动态内存分配,要求达到“简单应用”层次。
  - 5.1 new 的作用。
  - 5.2 delete 的作用。
- 6. 内联函数,要求达到“综合应用”层次。
  - 6.1 内联函数的作用。
  - 6.2 内联函数的使用方法。
- 7. 引用,要求达到“综合应用”层次。
  - 7.1 引用的作用。
  - 7.2 引用的使用方法。
- 8. 编译指令,要求达到“领会”层次。
  - 8.1 编译指令的几种形式及其作用。
  - 8.2 在程序中应用编译指令。
- 9. 改变习惯重新思考,要求达到“识记”层次。
  - 9.1 应从哪几个方面初步转变思想。
  - 9.2 C++的基本知识。
  - 9.3 使用已有知识学习 C++。

## 1.1 问答题

### 1. 面向对象的编程语言有哪些特点？

**答案：**面向对象的编程语言能更直接地描述客观世界中存在的事物以及它们之间的关系。C++语言是目前应用最广的面向对象的编程语言。面向对象的编程语言将客观事物看作具有属性和行为的对象，通过抽象找出同一类对象的共同属性(静态特征)和行为(动态特征)，形成类。通过类的继承与多态可以很方便地实现代码重用，大大缩短了软件开发周期，并使软件风格统一。因此，面向对象的编程语言使程序能够比较直接地反映问题域的本来面目，使软件开发人员能够利用人类认识事物所采用的一般思维方法来进行软件开发。

### 2. 什么是结构化程序设计方法？

**答案：**结构化程序设计的思路是：自顶向下、逐步求精。其程序结构是按功能划分为若干个基本模块；各模块之间的关系尽可能简单，在功能上相对独立；每一模块内部均是由顺序、选择和循环三种基本结构组成。结构化程序设计由于采用了模块分解与功能抽象和自顶向下的方法，从而有效地将一个较复杂的设计任务分解成许多易于控制和处理的子任务，便于开发和维护。虽然结构化程序设计方法具有很多的优点，但它仍是一种面向过程的程序设计方法，它把数据和处理数据的过程分离为相互独立的实体。当数据结构改变时，所有相关的处理过程都要进行相应的修改，程序的可重用性差。

### 3. 什么是对象？什么是面向对象方法？

**答案：**从一般意义上讲，对象是在现实世界中一个实际存在的事物。对象是一个独立单位，它具有自己的静态特征和动态特征。面向对象方法中的对象，是系统中用来描述客观事物的一个实体，它是用来构成系统的一个基本单位，由一组属性和一组行为构成。

面向对象的方法将数据及对数据的操作方法封装在一起作为一个整体(对象)。对同类型对象抽象出其共性，形成类。类中的大多数数据，只能用本类的方法进行处理。类通过一个简单的外部接口与外界发生关系，对象与对象之间通过消息进行通讯。这样，程序模块间的关系更为简单，实现了程序模块的独立性、数据的安全性。通过实现继承与多态性，还可以大大提高程序的可重用性，使得软件的开发和维护都更为方便。

### 4. 什么叫做封装？

**答案：**封装是面向对象方法的一个重要原则，就是把对象的属性和服务结合成一个独立的系统单位，并尽可能隐蔽对象的内部细节。

### 5. 面向对象的软件工程包括哪些主要内容？

**答案：**面向对象的软件工程是面向对象方法在软件工程领域的全面应用，它包括面向对象的分析(OOA)、面向对象的设计(OOD)、面向对象的编程(OOP)、面向对象的测试(OOT)和面向对象的软件维护(OOSM)等主要内容。

### 6. C++语言有哪些主要特点和优点？

**答案：**C++语言的主要特点表现在两个方面，一是全面兼容C，二是支持面向对象的方法，C++是一个更好的C，它保持了C的简洁高效、接近汇编语言、具有良好的可读性和可移植性等特点，对C的类型系统进行了改革和扩充，C++的编译系统能检查出更多的类型错误，C++语言最重要的特点是支持面向对象的方法。



7. 如下所示:

```
const double PI=3.14159265;  
#define double PI=3.14159265
```

使用关键字 const 定义符号常量和用宏定义 #define 对标识符定义的区别是什么?

**答案:** const 定义常量是一个说明语句,用分号结束;而 #define 命令是一个预处理命令,不能用分号结束。const 定义的常量是有类型的,编译时可以检查出类型错误;而 #define 命令定义的宏仅产生文本替换,不管内容是否正确。const 可以定义一个局部常量,可局部在一个函数体内;而 #define 定义的宏,作用域是从定义时开始,直到使用 #undef 取消其定义时为止,如果不取消其定义,直到整个文件结束。

8. 注释有什么作用? C++ 中有哪几种注释的方法? 它们之间有什么区别?

**答案:** 注释在程序中的作用是对程序进行注解和说明,以便于阅读。编译系统在对源程序进行编译时不理睬注释部分,因此注释对于程序的功能实现不起任何作用,也不会增加最终产生的可执行程序的大小。适当地使用注释,能够提高程序的可读性。在 C++ 中,有两种给出注释的方法:一种是沿用 C 语言的方法使用“/\*”和“\*/”括起来注释文字,一般用于多行注释。另一种方法是使用“//”,从“//”开始,直到它所在行的行尾,所有字符都被作为注释处理。

9. if(x=5)和 if(x==5)这两条语句的差别是什么?

**答案:** 语句 if(x=5)把 5 赋给 x,赋值表达式的值为 true,作为 if 语句的条件;语句 if(x==5)首先判断 x 的值是否为 5,若相等,条件表达式的值为 true,否则为 false。

10. 什么叫常量? 什么叫变量?

**答案:** 所谓常量是指在程序运行的整个过程中其值始终不可改变的量,除了用数字表示常量外,也可以用 const 为常量命名,这就是符号常量;在程序的执行过程中其值可以变化的量称为变量,变量是需要用名字来标识的。

11. 变量有哪几种存储类型?

**答案:** 变量有以下几种存储类型:auto 存储类型:采用堆栈方式分配内存空间,属于一时性存储,其存储空间可以被若干变量多次覆盖使用;register 存储类型:存放在通用寄存器中;extern 存储类型:在所有函数和程序段中都可引用;static 存储类型:在内存中是以固定地址存放的,在整个程序运行期间都有效。

12. C++ 中的函数是什么? 什么叫主调函数,什么叫被调函数,二者之间有什么关系? 如何调用一个函数?

**答案:** 一个较为复杂的系统往往需要划分为若干子系统,高级语言中的子程序就是用来实现这种模块划分的。C++ 语言中的子程序体现为函数。调用其他函数的函数被称为主调函数,被其他函数调用的函数称为被调函数。一个函数很可能既调用别的函数又被另外的函数调用,这样它可能既充当主调函数又充当被调函数。

调用函数之前先要声明函数原型。按如下形式声明:

**类型标识符 被调函数名(含类型说明的形参表);**

声明了函数原型之后,便可以按如下形式调用子函数:

**函数名(实参列表);**

13. 比较值调用和引用调用的相同点与不同点?

**答案：**值调用是指当发生函数调用时，给形参分配内存空间，并将实参的值传递给形参，这一过程是参数值的单向传递过程，一旦形参获得了值，便与实参脱离关系，此后形参发生改变，不会影响到实参。引用调用将引用作为形参，在执行主调函数中的调用语句时，系统自动用实参来初始化形参。这样形参就成为实参的一个别名，对形参的任何操作也就直接作用于实参。

14. 什么叫内联函数？它有哪些特点？

**答案：**定义时使用关键字 `inline` 的函数叫做内联函数。编译器在编译时在调用处用函数体进行替换，节省了参数传递、控制转移等开销。使用内联函数时应注意：(1)内联函数体内不能有循环语句和 `switch` 语句；(2)内联函数的定义必须出现在内联函数第一次被调用之前；(3)对内联函数不能进行异常接口声明。

15. 函数原型中的参数名与函数定义中的参数名以及函数调用的参数名必须一致吗？

**答案：**不必一致，所有的参数是根据位置和类型而不是名字来区分的。

16. 重载函数时通过什么来区分？

**答案：**重载函数的函数名是相同的，但它们的参数的个数和数据类型不同，编译器根据实参和形参的类型及个数的最佳匹配，自动确定调用哪一个函数。

17. 什么叫做作用域？有哪几种类型的作用域？什么叫做局部变量？什么叫做全局变量？

**答案：**作用域讨论的是标识符的有效范围，作用域是一个标识符在程序正文中有效的区域。C++的作用域分为函数原型作用域、块作用域(局部作用域)、类作用域和文件作用域。局部变量一般来讲就是具有块作用域的变量；全局变量就是具有文件作用域的变量。

18. 什么叫做可见性？可见性的一般规则是什么？

**答案：**可见性是标识符是否可以引用的问题。可见性的一般规则是：标识符要声明在前，引用在后；在同一作用域中，不能声明同名的标识符。对于在不同的作用域声明的标识符，遵循的原则是：若有两个或多个具有包含关系的作用域，外层声明的标识符如果在内层没有声明同名标识符时，外层标识符在内层仍可见；如果内层声明了同名标识符，则外层标识符不可见。

19. 数组 `A[3][4][5]`一共有多少元素？

**答案：** $3 \times 4 \times 5 = 60$  个元素。

20. 在数组 `A[10]`中第一个元素和最后一个元素是哪一个？

**答案：**第一个元素是 `A[0]`，最后一个元素是 `A[9]`。

21. 用一条语句定义一个有  $2 \times 5$  个元素的二维整形数组，并依次赋予  $1 \sim 10$  的初值。

**答案：**`int a[2][5] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};`

或：`int a[2][5] = {{1, 2, 3, 4, 5}, {6, 7, 8, 9, 10}};`

22. 运算符 `*` 和 `&` 的作用是什么？

**答案：**`*` 称为指针运算符，是一个一元操作符，用于访问指针所指向的对象的值；`&` 称为取地址运算符，也是一个一元操作符，用来得到一个对象的地址。

23. 什么叫做指针？指针中储存的地址和这个地址中的值有何区别？

**答案：**指针是一种数据类型；具有指针类型的变量称为指针变量。指针变量存放的是另一个对象的地址，这个地址中的值就是另一个对象的内容。

24. 在字符串“Hellp !”中结束符是什么？

答案：是 NULL 字符。

25. 引用和指针有何区别？何时只能使用指针而不能使用引用？

答案：引用是一个别名，不能为 NULL 值；指针是一个存放地址的变量，当需要对变量重新赋以另外的地址或赋值为 NULL 时只能使用指针。

26. `const int *p1` 和 `int *const p2` 的区别是什么？

答案：`const int *p1` 声明了一个指向整型常量的指针 `p1`，因此不能通过指针 `p1` 来改变它所指向的整型值；`int *const p2` 声明了一个指针型常量，用于存放整型变量的地址，这个指针一旦初始化后，就不能被重新赋值了。

## 1.2 选择题

1.  (1) 不是面向对象系统的基本特征。

(1) A. 继承

B. 类

C. 对象

D. 重载

答案：D

2. 下列各种高级语言中， (1) 是最早提出了对象的概念。

(1) A. Algol 60

B. C++

C. Ada

D. Simula 67

答案：D

3. 在 C++ 与 C 语言的关系描述中， (1) 是不对的。

(1) A. C 语言和 C++ 是兼容的

B. C++ 是对 C 进行了改进

C. C++ 和 C 都是面向对象系统的

D. C 语言是 C++ 的一个子集

答案：C

4. 下列各种高级语言中， (1) 是面向对象的程序设计语言。

(1) A. C++

B. PASCAL

C. BASIC

D. FORTRAN

答案：A

5. C++ 对 C 语言作的重要改进是  (1)，使 C 语言从面向过程变成为面向对象。

(1) A. 增加了一些新的运算符

B. 引进了类和对象的概念

C. 允许函数重载

D. 允许函数设置缺省参数

答案：B

6. 类型修饰符 `unsigned` 修饰  (1) 类型是错误的。

(1) A. `long int`

B. `int`

C. `float`

D. `char`

答案：C

7. 下列十六进制的整型常数表示中  (1) 是错误的。

(1) A. `5fx`

B. `0xaf`

C. `0x7c`

D. `0Xed`

答案 :A

8. 下列变量名中,  是合法的。

- A. C+A  
C. double

- B. byte-size  
D. X

答案 :D

9. 在 `int a[5] = {1, 3, 5}`; 中, 数组元素 `a[1]` 的值是 。

- A. 0  
C. 1

- B. 3  
D. 2

答案 :B

10. 在 `int a[][3] = {{1}, {3, 2}, {4, 5, 6}, {0}}`; 中 `a[2][2]` 的值是 。

- A. 0  
C. 6

- B. 5  
D. 2

答案 :C

11. 下列给字符数组进行初始化中,  是正确的。

- A. `char s1[] = "abcd"`;  
C. `char s3[][3] = {'a', 'x', 'y'}`;

- B. `char s2[3] = "xyz"`;  
D. `char s4[2][3] = {"xyz", "mnp"}`;

答案 :A

12. 在定义 `int a=3, *p=&a`; 后, `*p` 的值是 。

- A. 无意义  
C. 3

- B. 变量 a 的地址值  
D. 变量 p 的地址值

答案 :C

13. 对于 `int *pa[5]`; 的描述,  是正确的。

- A. `pa` 是一个具有 5 个元素的指针数组, 每个元素是一个 `int` 型指针  
B. `pa` 是一个指向数组的指针, 所指向的数组是 5 个 `int` 型元素  
C. `pa` 是一个指向某数组中第 5 个元素的指针, 该元素是 `int` 型变量  
D. `pa[5]` 表示某个数组的第 5 个元素的值

答案 :A

14. 下列关于指针的运算中,  是非法的。

- A. 两个指针在一定条件下, 可以相加。  
B. 两个指针在一定条件下, 可以进行相等或不等的运算;  
C. 一个指针可以加上两个整数之差;  
D. 可以用一个空指针赋值给某个指针;

答案 :A

15. 指针可以用来表示数组元素, 下列表示中  是错误的。

已知: `int a[3][7]`;

- A. `*(&a[0][0]+2)`  
C. `*(*a+1)`

- B. `*(a+1)[5]`  
D. `*(*a+3)`

答案 :D

16. 下列表示引用的方法中,  是正确的。

已知: int m=10;

- A. int &z;
- B. int &y=10;
- C. int &x=m;
- D. float &t=&m;

答案 :C

17. 预处理命令中都是以  开头的。

- A. \*
- B. #
- C. ;
- D. /

答案 :B

18. 文件包含命令中被包含的文件的扩展名 。

- A. 不一定是.h
- B. 不能用.h
- C. 必须是.c
- D. 必须为.h

答案 :A

19. 下列  是语句

- A. ;
- B. a=17
- C. x+y
- D. cout<<"\n"

答案 :A

20. 下列 for 循环中 a=1 语句执行的次数为 。

```
for (int i(0),x=0;! x&& i<=5;i++) a=1;
```

- A. 5
- B. 6
- C. 1
- D. 无限

答案 :B

21. 下列 while 循环的次数是 。

```
while (int i=0) i--;
```

- A. 5
- B. 1
- C. 0
- D. 无限

答案 :C

22. 下列 do-while 循环的次数为 。

```
已知: int i(5);  
do  
{  
    i--;  
} while (i!=0);
```

- A. 1
- B. 5
- C. 0
- D. 无限

答案 :D

23. 下列 for 循环的循环体执行次数为 。

```
for (int i(0),j(10);i=j=10;i++,j--)
```

- A. 无限
- B. 1
- C. 10
- D. 0

答案 :A

24. 下述关于循环体的描述中,  是错误的。

- A. 循环体中不能出现 goto 语句
- B. 循环体中还可以出现循环语句
- C. 循环体中可以出现 break 语句和 continue 语句
- D. 循环体中可以出现开关语句

答案 :A

25. 下述关于 goto 语句的描述中,  是正确的。

- A. goto 语句可在一个文件中随意转向
- B. goto 语句后面要跟上一个它所转向的语句
- C. goto 语句可以同时转向多条语句
- D. goto 语句可以从一个循环体内转到循环体外

答案 :D

26. 下述关于 break 语句的描述中,  是不正确的。

- A. break 语句可用于循环体内, 它将退出该重循环
- B. break 语句可用于开关语句中, 它将退出开关语句
- C. break 语句可用于 if 体内, 它将退出该 if 语句
- D. break 语句在一个循环体内可以出现多次

答案 :C

27. 下列关于开关语句的描述中,  是正确的。

- A. 开关语句中 case 子句后面的表达式可以是整型表达式
- B. 开关语句中每个语句序列中必须有 break 语句
- C. 开关语句中 default 子句只能放在最后
- D. 开关语句中 default 子句可以没有, 也可有一个

答案 :D

28. 下列关于条件语句的描述中,  是错误的。

- A. if 语句中可以有多个 else if 子句
- B. if 语句中 if 体内不能是开关语句
- C. if 语句中只有一个 else 子句
- D. if 语句中 if 体内可以是循环语句

答案 :B

29. 当一个函数无返回值时, 定义它时函数的类型应是 。

- A. void
- B. 任意
- C. int
- D. 无

答案 :A

30. 在函数说明时, 下列  项是不必要的。

- A. 返回时表达式
- B. 函数参数类型和名字
- C. 函数名字
- D. 函数的类型

答案 :A

31. 函数返回值类型与函数返回值表达式的类型的描述中,  是错误的。
- A. 函数返回值类型决定了返回值表达式的类型
  - B. 函数返回值的类型是在定义函数时确定的, 在函数调用时是不能改变的,
  - C. 函数返回值表达式类型与函数返回值类型不同时, 表达式类型应转换成函数返回值类型
  - D. 函数返回值的类型就是返回值表达式的类型

答案 :D

32. 在一个被调用函数中, 关于 return 语句的描述,  是错误的。
- A. 被调用函数中如果有返回值, 就一定要有 return 语句
  - B. 被调用函数中可以使用多个 return 语句
  - C. 被调用函数中, 一个 return 语句可以返回多个值给调用函数
  - D. 被调用函数中可以不用 return 语句

答案 :C

33. 下列的  是引用调用。
- A. 形参是引用, 实参是变量
  - B. 形参和实参都是变量
  - C. 形参是数组名, 实参是数组名
  - D. 形参是指针, 实参是地址值

答案 :A

34. 在传值调用中, 要求 。
- A. 形参和实参类型任意, 个数相等
  - B. 实参和形参对应的类型一致, 个数相等
  - C. 实参和形参都是变量
  - D. 实参和形参对应的类型一致, 个数任意

答案 :B

35. 下列标识符中,  不是局部变量。
- A. 外部 static 类
  - B. auto 类
  - C. register 类
  - D. 函数形参

答案 :A

36. 在一个函数中, 要求通过函数来实现一种不太复杂的功能, 并且要求加快执行速度, 选用 。
- A. 内联函数
  - B. 重载函数
  - C. 嵌套调用
  - D. 递归调用

答案 :A

37. 求自然数 1~10 之和。

```
#include<iostream.h>
```

```
void main()
```

```
{
```

```
    int i(1), sum(0);
```

```
do {
    sum += i;
    (1);
}while(i <= 10);
cout << "sum=" << sum << endl;
}
```

- (1) A.  $i += sum$                                       B.  $i += i$   
 C.  $sum = i + 1$                                       D.  $i ++$

**答案 :** (1) D

38. 求自然数 1~10 之和。

```
#include <iostream.h>
void main()
{
    int i, sum=0;
    for(i=0; (1); i++)
        (2);
    cout << "sum=" << sum << endl;
}
```

- (1) A.  $i <= 10$                                       B.  $i < 10$   
 C.  $i = 10$     D.  $sum <= 10$   
 (2) A.  $sum += i$                                       B.  $i += i$   
 C.  $sum = i + 1$                                       D.  $i = sum$

**答案 :** (1) A    (2) A

39. 内联函数计算圆的面积,程序输出结果是 12.56。

```
#include <iostream.h>
inline double CalArea(double radius)                 //内联函数,计算圆的面积
{
    return 3.14 * radius * radius;
}
int main()
{
    double r(2.0);                                    //r 是圆的半径
    double area;
    area = (1);                                        //调用内联函数求圆的面积
    cout << area << endl;
    return 0;
}
```



(1) A. CalArea(r)  
C. CalArea

B. CalArea(area)  
D. 3.14 \* r \* r

答案 : (1) A

40. 读入三个浮点数,将整数部分和小数部分分别输出。

```
#include <iostream.h>
```

```
void splitfloat(float x, int *intpart, float *fracpart)
```

```
{  
    *intpart=int(x);           //取 x 的整数部分  
    *fracpart=;       //取 x 的小数部分  
}
```

```
void main()
```

```
{  
    int i,n;  
    float x,f;  
    cout<<"Enter 3 float point numbers:"<<endl;  
    for(i=0;i<3;i++)  
    {  
        cin>>x;  
        ;           //变量地址作为实参  
        cout<<"Integer Part="<<n<<" Fraction Part="<<f<<endl;  
    }  
}
```

(1) A. float(x)

B. x - \*intpart

C. x - INT(x)

D. \*intpart

(2) A. splitfloat(x,&n,&f)

B. splitfloat()

C. splitfloat(x,n,f)

D. splitfloat(x,\*n,\*f)

答案 : (1) B (2) A

41. 函数将华氏温度(F)转换为摄氏温度( $C = F - 32 * 5/9$ )。

```
#include <iostream.h>
```

```
float Convert(float);
```

```
int main()
```

```
{  
    float TempFer;  
    float TempCel;  
    cout << "Please enter the temperature in Fahrenheit: ";  
    ;
```