

## 一、选择题

1 假定 int 类型变量占用两个字节，其有定义  $\text{int } x[10]=\{0, 2, 4\};$ ，则数组 x 在内存中所占字节数是( )。

A. 3B. 6C. 10D. 20

参考答案：D

参考解析：x 数组共有 10 个元素，在花括号内只提供 3 个初值，这表示只给前面的 3 个成员赋值，后 7 个元素的值为 0，所以，一共有 20 个字节。

2 有两个关系 R，S 如下：

R		
A	B	C
1	2	3
2	3	4
3	4	5

S	
A	B
a	3
b	0
c	2

由关系 R 通过运算得到关系 S，则所使用的运算为( )。

A. 选择 B. 投影 C. 插入 D. 连接

参考答案：B

参考解析：投影运算是指对于关系内的域指定可引入新的运算。本题中 s 是在原有关系 R 的内部进行的，是由 R 中原有的那些域的列所组成的关系。所以选择 B。

3 对于循环队列，下列叙述中正确的是( )。

- A. 队头指针是固定不变的
- B. 队头指针一定大于队尾指针
- C. 队头指针一定小于队尾指针

D. 队头指针可以大于队尾指针，也可以小于队尾指针

参考答案：D

参考解析：循环队列的队头指针与队尾指针都不是固定的，随着入队与出队操作要进行变化。因为是循环利用的队列结构所以队头指针有时可能大于队尾指针有时也可能小于队尾指针。

4 下列有关函数重载的叙述中，错误的是( )。

A. 函数重载就是用相同的函数名定义多个函数

B. 重载函数的参数列表必须不同

C. 重载函数的返回值类型必须不同

D. 重载函数的参数可以带有默认值

参考答案：C

参考解析：所谓函数重载是指同一个函数名可以对应着多个函数的实现。每种实现对应着一个函数体，这些函数的名字相同，但是函数参数的类型不同，还允许参数的个数不同，但不能只有函数的返回值类型不同而参数的个数和类型相同的重载。所以本题答案为c。

5 有三个关系 R S 和 T 如下。

R			S			T		
A	B	C	A	B	C	A	B	C
a	1	2	a	1	2	b	2	1
b	2	1	d	2	1	e	3	1
c	3	1						

则由关系 R 和 S 得到关系 T 的操作是( )。

A. 自然连接 B. 并 C. 交 D. 差

参考答案：D

参考解析：关系 T 中的元组是关系 R 中有而关系 s 中没有的元组的集合，即从关系 R 中除去与关系 s 中相同元组后得到的关系 T。所以做的是差的运算。

6 下面关于类和对象的描述中，错误的是( )。

- A. 类就是 C 语言中的结构体类型，对象就是 C 语言中的结构体变量
- B. 类和对象之间的关系是抽象和具体的关系
- C. 对象是类的实例，一个对象必须属于一个已知的类
- D. 类是具有共同行为的若干对象的统一描述体

参考答案：A

参考解析：在 c++语言中，类和结构体有本质区别，类中包含数据及操作函数，而结构体只包含有数据。

7 有如下程序：

```
#include

using namespace std;

Class test{
private:
    int a;
public:
    test( ) {Cout<<" ConstruCtor" <
    test(int A, {Cout<
    test(const test&_test) { a=test.a;
    Cout<<" Copy ConstruCtor" <
    test( ) {Cout<<" destruCtor" <
};
```

```
int main(    ){ test A(3); return 0;

}
```

执行这个程序的输出结果是( )。

A. 3B. ConstruCtor

destruCtorC. Copy ConstruCtor

destruCtorD. 3

destruCtor

参考答案: D

参考解析: 本题考查默认构造函数和带参数的构造函数以及析构函数, 本题中定义了一个对象 A(3), 对象带着参数, 所以执行带参数的构造函数, 输出 3, 然后执行析构函数, 输出 destructor。所以本题答案为 D。

8 有如下函数模板:

```
template
```

```
T east(U u){return u;}
```

其功能是将 U 类型数据转换为 T 类型数据。已知 i 为 int 型变量, 下列对模板函数 east 的调用中正确的是( )。

A. cast(i); B. cast<>(i); C. east(i); D. east(i);

参考答案: D

参考解析: 本题考查函数模板的基本运用, 属于基础知识。函数模板实际上是建立一个通用函数, 其函数类型和形参类型不具体指定, 用一个虚拟的类型代表, 这个通用函数就是函数模板。函数模板可以设定默认参数, 这样在调用模板函数时就可以省实参, 函数模板的实例就是一个函数的定义。所以本题答案为 D。

9 下面程序的运行结果是( )。

```
#include
```

```

void fun(int *a, int *B. {

int x=*a;

*a=*b;*b=X;

cout<<*a<<*b<<" " :

void main() {

int X=1, Y=2; fun(&x, &y);

cout<

}

```

A. 12 12B. 12 21C. 21 12D. 21 21

参考答案: D

参考解析: 本题考查的是指针和引用, 函数 fun 完成把 \*a 和 \*b 互换的操作。

10 在 C++语言中, 打开一个文件就是将这个文件与一个( )建立关联。

A. 流 B. 类 C. 结构 D. 对象

参考答案: A

参考解析: C++的 V0 系统是通过一种称为流的机制来实现文件和控制台的 I/O 操作的。简单地说, 在 C++中, 输入/输出操作是通过流来完成的。而文件流以磁盘文件及其他可按文件方式进行管理的外部设备为输入/输出对象。每个文件流都应当与一个打开的文件相联系, 关闭文件就是取消这种联系。

11 下列语句中, 错误的是( )。

A. const int buffer: 256;

B. const int temp;

C. const double\*point;

D. `const double*rt=new double(5.5);`

参考答案: B

参考解析: `constexpr` 是一个 C++ 语言的关键字, 它限定一个变量不允许被改变。使用 `const` 在一定程度上可以提高程序的安全性和可靠性。限定符声明变量只能被读, 必须初始化; , B 选项中没有初始化, C 选项为指向常量的指针, 定义时初始化与否均可, 如果是 `const*double point` 形式, 则为常指针, 必须在定义时进行初始化, A、c、D 都正确。

12 下列程序的输出结果为( )。

```
#include

void main()

{

char*a[]={ " hello", " the", " world" };

char** pa = a;

pa++;

cout<<*a<

}

A. helloB. theC. worldD. hellotheworld
```

参考答案: B

参考解析: 本题主要考查的是指针数组和指向指针的指针之间的关系, 其中 `a` 是指针数组, `pa` 是指向指针数组行的指针, 所以 `pa` 自加 1 相当于指向下一行。

13 关于运算符重载, 下列表述中正确的是( )。

A. C++ 已有的任何运算符都可以重载

B. 运算符函数的返回类型不能声明为基本数据类型

C. 在类型转换符函数的定义中不需要声明返回类型

D. 可以通过运算符重载来创建 C++ 中原来没有的运算符

参考答案: C

参考解析: 重载运算符的规则如下: ①c++不允许用户自己定义新的运算符, 只能对已有的 c++运算符进行重载; ②c++不能重载的运算符只有 5 个; ③重载不能改变运算符运算对象的个数; ④重载不能改变运算符的优先级和结合性; ⑤重载运算符的函数不能有默认的参数; ⑥重载的运算符必须和用户定义的自定义类型的对象一起使用. 至少应有一个是类对象, 即不允许参数全部是 c++的标准类型。故本题答案为 c。

14 可以用 P.a 的形式访问派生类对象 P 的基类成员 a, 其中 a 是( )。

A. 私有继承的公有成员

B. 公有继承的私有成员

C. 公有继承的保护成员

D. 公有继承的公有成员

参考答案: D

参考解析: 只有公有继承的公有成员才能被派生类的对象直接访问, 其他的情况都会因为访问权限问题产生错误。

15 在下列原型所示的 C++ 函数中, 按“传值”方式传递参数的是( )。

A. void f1(int X);

B. void t2(int \*f);

C. void f3(const int \*X);

D. void f4(int &x);

参考答案: A

参考解析：函数参数的 3 种传递方式：①将变量名作为形参和实参，即传值方式；②传递变量指针；③引用形参。其中 B、C 选项都是传递变量指针，D 选项为引用形参，所以本题答案为 A。

16 以下关键字不能用来声明类的访问权限的是( )。

A. public B. static C. protected D. private

参考答案：B

参考解析：类的成员访问权限有 3 种：public(公用)、protected(保护)、private(私有)。

17 下面程序的运行结果为( )。

```
#include <iostream>
using namespace std;

class A {
public:
    int num;
    A(int i) {num=i;}
    A(A&a) {num=a.num++;} void print() {cout<<num<<endl;}
};

void main() {
    A a(1), b(A); a.print();
    b.print();
}
```

A. 11 B. 10 C. 2 D. 22

参考答案：C

参考解析：本题考查考生对拷贝构造函数的掌握。因为++运算是右结合的，所以在使用 a 对 b 赋值时，b 的 num 变为 1，而 a 的 num 变为 2(先赋值再自加)。

18 下面是重载为非成员函数的运算符的函数原型，其中错误的是( )。



- A. Fraction operator+(Fraction, Fraction);
- B. Fraction operator-(Fraction);
- C. Fraction&operator=(Fraction&, Fraction);
- D. Fraction&operator+=(Fraction&, Fraction);

参考答案: C

参考解析: 本题考查四种运算符作为非成员函数的重载的使用方法, 根据重载规则, 本题答案为 c, c 项正确原型为: Fraction operator=(Fraction&).

19 下列语句中错误的是( )。

- A. Const int a;
- B. Const int a=10;
- C. Const int \* point=0;
- D. Const int \* point=new int(10);

参考答案: A

参考解析: const 是一个 c++ 语言的关键字, 它限定一个变量不允许被改变。使用 const 在一定程度上可以提高程序的安全性和可靠性。限定符声明变量照能被读, 必须初始化。A 选项没有初始化, B、C、D 选项都正确。

20 下列关于函数的描述中, 错误的是( )。

- A. 函数可以没有返回值
- B. 函数可以没有参数
- C. 函数可以是一个类的成员
- D. 函数不能被定义为模板

参考答案: D

参考解析：本题考查函数的基础知识，函数可以没有返回值，没有参数，也可以是一个类的成员，函数可以定义为模板，即模板函数。

22 有三个关系 R、S 和 T 如下：

R			S			T		
A	B	C	A	B	C	A	B	C
a	1	2	a	1	2	c	3	1
b	2	1	b	2	1			
c	3	1						

则由关系 R 和 S 得到关系 T 的操作是( )。

A. 自然连接 B. 差 C. 交 D. 并

参考答案：B

参考解析：关系 T 中的元组是 R 关系中有而 S 关系中没有的元组的集合，所以进行的是差的运算。

23 有如下程序：

```
#include

using namespace std;

class VAC{
public:
    int f()const{return 3;}
    int f(){return 1;}
};

int main()
{
    VAC v1;

    const VAC v2;
```

```
cout<

return 0;

}
```

执行这个程序的输出结果是( )。

A. 53B. 35C. 55D. 33

参考答案: A

参考解析: 本题考查常对象及常成员函数及构造函数, 常对象只能调用常成员函数, 所以本题中, 对象 v1 为普通对象, 执行 v1.f() 后, 会执行默认构造函数, 输出 5。对象 v2 为常对象, 执行 v2.f() 后, 会执行常构造函数, 输出 3。

24 一间宿舍可住多个学生, 则实体宿舍和学生之间的联系是( )。

A. 一对一 B. 一对多 C. 多对一 D. 多对多

参考答案: B

参考解析: 因为一间宿舍可以住多个学生即多个学生住在一个宿舍中, 但一个学生只能住一间宿舍, 所以实体宿舍和学生之间是一对多的关系。

25 下列关于二叉树的叙述中, 正确的是( )。

- A. 叶子结点总是比度为 2 的结点多一个
- B. 叶子结点总是比度为 2 的结点多一个
- C. 叶子结点数是度为 2 的结点数的两倍
- D. 度为 2 的结点数是度为 1 的结点数的两倍

参考答案: B

参考解析: 根据二叉树的基本性质 3: 在任意一棵: 二叉树中, 度为 0 的叶子结点总是比度为 2 的结点多一个。所以选择 B。

26 有以下语句, 则对 a 数组元素的引用不正确的是 ( $0 \leq i \leq 9$ ) ( )。

```
int a[10]={0, 1, 2, 3, 4, 5, 6, 7, 8, 9}, *P=a;
```

A. a[P-a] B. \*(&a[i]) C. P[i] D. \*(\*a+i)

参考答案: D

参考解析: 选项 D 第一层括号中为数组 a 中第 i 项元素的值, 外面再加指针运算符没有意义。

27 下列程序的运行结果为( )。

```
#include
```

```
void print(double A.
```

```
{
```

```
cout<<++a;
```

```
}
```

```
void print(int a,int B.
```

```
{
```

```
cout<<
```

```
}
```

```
void main()
```

```
{
```

```
print(1, 2);
```

```
cout<<" ";
```

```
print(3, 4);
```

```
}
```

A. 1.2 34B. 2.2 34C. 1.2 43D. 2.2 43

参考答案: D

参考解析: 本题考查的是对重载函数的掌握, 因为++a 是右结合的, 所以 a 先自加 1 再输出, 即为 2.2, 之后调用函数 print(int a, int b) 输出 h 和 a, 即 43。

点击查看试题笔记 (1)28 存在定义 int a[10], x, \*pa; , 若 pa=&a[0], 下列的( ) 选项和其他 3 个选项不是等价的。

A. x=\*pa; B. x=\*(a+1); C. x=\*(pa+1) D. x=a[1];

参考答案: A

参考解析: 本题考查的是数组和指向数组的指针之间的关系, 选项 A 中 x 是被数组 a 的第一个元素赋值, 其余的都是被数组 a 的第二个元素赋值。

29 下列叙述中正确的是( )。

- A. 循环队列有队头和队尾两个指针, 因此, 循环队列是非线性结构
- B. 在循环队列中, 只需要队头指针就能反映队列中元素的动态变化情况
- C. 在循环队列中, 只需要队尾指针就能反映队列中元素的动态变化情况
- D. 循环队列中元素的个数是由队头指针和队尾指针共同决定

参考答案: D

参考解析: 循环队列有队头和队尾两个指针, 但是循环队列仍是线性结构的, 所以 A 错误; 在循环队列中需要队头指针与队尾指针来共同反映队列中元素的动态变化情况, 所以 B 与 C 错误。

30 通常拷贝构造函数的参数是( )。

- A. 某个对象名
- B. 某个对象的成员名
- C. 某个对象的引用名

D. 某个对象的指针名

参考答案: C

参考解析: 复制初始化构造函数的作用是用一个已知的对象来初始化一个被创建的同类的对象, 该函数只有一个参数, 并且是对同类的某个对象的引用。

31 如果在表达式 $++x/y$ 中, “++”是作为友元函数重载的, “/”是作为成员函数重载的, 则该表达式还可为( )。

A.  $(\text{operator}++(x)).\text{operator}/(Y)$

B.  $(\text{operator}++(0)).\text{operator}/(Y)$

C.  $\text{operator}/((\text{operator}++(x, 0)), Y)$

D.  $\text{operator}/((\text{operator}++(0)), Y)$

参考答案: A

参考解析: c++中用成员函数重载 $++X$ 为:  $x.\text{operator}++()$ , 用友元函数重载 $++x$ 为:  $\text{operator}++(x)$ , 用成员函数重载 $x/y$ 为:  $x.\text{operator}/(Y)$ , 用友元函数重载 $x/y$ 为:  $\text{operator}/(X, Y)$ 。

32 一个函数功能不太复杂但要求被频繁调用, 应选用( )。

A. 内联函数 B. 重载函数 C. 递归函数 D. 嵌套函数

参考答案: A

参考解析: 本题考查的是内联函数的使用, 它的引入是为了, 解决程序中函数调用的效率问题, 在编译时, 编译器将程序中出现的内联函数的调用表达式用内联函数的函数体来替换。

33 数据库应用系统中的核心问题是( )。

A. 数据库设计

B. 数据库系统设计

C. 数据库维护

D. 数据库管理员培训

参考答案: A

参考解析: 数据库应用系统中的核心问题是数据库的设计。

34 有如下程序: Class Base{ public:

int data; };

Class Derived1: public Base{};

Class Derived2: protected Base{};

int main( ) {

Derived1 d1; Derived2 d2; d1.data=0; //①

d2.data=0; //② return 0;

}

下列关于程序编译结果的描述中, 正确的是( )。

A. ①②皆无编译错误

B. ①有编译错误, ②无编译错误

C. ①无编译错误, ②有编译错误

D. ①②皆有编译错误

参考答案: C

参考解析: 本题考查公用继承和保护继承对基类成员的访问属性。在公用继承中, 基类的公用成员和保护成员在派生类中保持原有访问属性, 其私有成员仍为基类私有、稿. 受保护继承中, 基类的公用成员和保护成员在派生类中成了保护成员, 其私有成员仍为基类私有。本题中 Derived1 公用继承 Base, 所以①编译正确, Derived2 保护继承 Base, 所以②编译不 正确。

35 下列关于基类和派生类关系的叙述中, 正确的是( )。

- A. 每个类最多只能有一个直接基类
- B. 派生类中的成员可以访问基类中的任何成员
- C. 基类的构造函数必须在派生类的构造函数体中调用
- D. 派生类除了继承基类的成员，还可以定义新的成员

参考答案：D

参考解析：本题考查派生类与基类的关系，派生类可以有多个直接基类，而派生类有三种继承方式，在私有继承中，派生类的成员并不能访问基类的成员。如果基类中定义了缺省构造函数或根本没有定义任何一个构造函数时，在派生类构造函数的定义中可以省对基类构造函数的调用。所以本题答案为D。

36 结构化程序所要求的基本结构不包括( )。

- A. 顺序结构
- B. GOTO 跳转
- C. 选择(分支)结构
- D. 重复(循环)结构

参考答案：B

参考解析：1966 年 Boehm 和 Jacopini 证明了程序设计语言仅仅使用顺序、选择和重复三种基本控制结构就足以表达出各种其他形式结构的程序设计方法。

37 将 E—R 图转换为关系模式时，实体和联系都可以表示为( )。

- A. 属性 B. 键 C. 关系 D. 域

参考答案：C

参考解析：从 E—R 图到关系模式的转换是比较直接的，实体与联系都可以表示成关系，E—R 图中属性也可以转换成关系的属性。

38 下面是一个模板声明的开始部分：



templatedouble...由此可知( )。

- A. 这可能是一个函数模板的声明
- B. 这可能是一个类模板的声明
- C. 这既可能是一个函数模板的声明，也可能是一个类模板的声明
- D. 这肯定是一个错误的模板声明

参考答案：A

参考解析：本题考查函数模板和类模板的基础知识，从 templatedouble 这里就可以看出这是一个函数模板的声明。

39 有如下程序：

```
#include

using namespace std;

class AA{

    int k;

protected:

    int n;

    void setK(int k){this->k=k;}

public:

    void setN(int n){this->n=n;}

};

class BB: public AA{/*类体略*/>

;

};
```

```
int main()

}

BB X;

A n=1;//1

B setN(2);//2

C k=3;//3

D setK(4);//4

return 0;

}
```

在标注号码的四条语句中正确的是( )

A. 1B. 2C. 3D. 4

参考答案: B

参考解析: 本题考查公用继承对基类成员的访问属性。在公用继承中, 基类的公用成员和保护成员在派生类中保持原有访问属性, 其私有成员仍为基类私有。所以本题中 n、k 和 setN 变为了派生类的保护成员, 不能直接使用, 即 1、3、4 错误, 所以 B 选项正确。

均有如下头文件:

```
int f1();

static int f2();

class Mx{

public:

int f3();
```

```
static int f4();
```

```
};
```

在所描述的函数中，具有隐含的 this 指针的是( )。

A. f1B. f2C. f3D. f4

参考答案：C

参考解析：只有类的非静态成员函数才隐含 this 指针。其作用域是类内部，当类的非静态成员函数中访问类的非静态成员时，编译器会自动将对象本身的地址作为一个隐含参数传递给函数，而不必一定写上 this。所以本题答案为 C。

## 二、基本操作题

41 请使用 VC6 或使用【答题】菜单打开考生文件夹 proj1 下的工程 proj1。该工程中包含程序文件 main.cpp，其中有类 Door（“门”）和主函数 main 的定义。程序中位于每个“//ERROR\*\*\*\*found \*\*\*\*”之后的一行语句有错误，请加以改正。改正后程序的输出结果应为：

打开 503 号门…门是锁着的，打不开。

打开 503 号门的锁，锁开了。

打开 503 号门…门打开了。

打开 503 号门…门是开着的，无须再开门。

锁上 503 号门…先关门…门锁上了。

注意：只修改每个“//ERROR\*\*\*\*\*found \*\*\*\*\*”下的那一行，不要改动程序中的其他内容。

```
#include
```

```
using namespace std;
```

```
class Door{
```

```
int Bum;//门号
```

```

bool closed;//true 表示门关着

bool locked: //true 表示门锁着

public:

Door(int nHm) {

//ERROR*****found*****

nun=this ->ann:

closed=locked=true:

}

bool isClosed()const {return closed;}

//门关着时返回 true, 否则返回 false

bool isOpened()const {return !closed;}

//门开着时返回 true, 否则返回 false

bool isLocked()const {return locked;}

//门锁着时返回 true, 否则返回 false

bool isUnlocked()const {return !locked;}

//门未锁时返回 true, 否则返回 false

void open() { //开门

cout<<" 门是开着的, 无须再开门。" ;

//ERROR*****found*****

if(closed)

cout<<" 门是开着的, 无须再开门。" ;

```

```
else if(!locked)

cout<<" 门是锁着的，打不开。";

else{

closed=false;

eout<<" 门打开了。";

}

}

void close() { //关门

cout<

if(closed)

cout<<" 门是关着的，无须再关门。";

else{

closed=true;

cout<<" 门关上了。";

}

}

//ERROR*****found*****

void lock() const { //锁门

cout<

if(!locked)

cout<<" 门是锁着的，无须再锁门。";
```

```
else{

    if(!closed){

        cout<<" 先关门..." ;

        closed=true;

    }

    locked=true;

    cout<<" 门锁上了。" ;

}

}

wild unlock() { //开锁

    cout<

    if(!locked)

        cout<<" 门没有上锁，无须再开锁" ;

    else{

        locked=false;

        cout<<" 锁开了。" ;

    }

};

int main() {

    Door door(503);

    door.open();
```

```

door.unlock();

door.open();

door.open();

floor.lock();

return 0;

}

```

参考解析:

```

(1) this->num=num;

(2) if(!closed)

(3) void lock() {

```

### 三、简单应用题

42 请使用 VC6 或使用【答题】菜单打开考生文件夹 proj2 下的工程 proj2，该工程中包含一个程序文件 main.cpp，其中有类 CPolygon(“多边形”)、CRectangle(“矩形”)、CTriangle(“三角形”)的定义。请在横线处填写适当的代码并删除横线，以实现上述类定义。该程序的正确输出结果应为：。注意：只能在横线处填写适当的代码，不要改动程序中的其他内容，也不要删除或移动“//\*\*\*\*\*found\*\*\*\*\*”。 #include

```

{tout<<——<

int area(void){return(width *height);}

class CTriangle: public CPolygon{

int length;//三角形一边长 int height;//该边上的高

public:

CTriangle(int l, int h): length(l), height(h) {}

//*****found*****

```

```

int area(void) {return(——)/2;}

};

int main() {

CRectangle rect(4, 5); CTriangle trgl(4, 5);

//*****found***** _____ *ppoly1, * ppoly2; ppoly1=&rect;

ppoly2=&trgl;

ppoly1->printarea(); ppoly2->printarea(); retun 0;

```

参考解析:

- (1) virtual int area(void)=0;
- (2) area()
- (3) length\*height
- (4) CPolygon

#### 四、综合应用题

43. 请使用 VC6 或使用【答题】菜单, 打开考生文件夹 proj3 下的工程 proj3, 其中声明的 DataList 类, 是一个用于表示数据表的类。sort 成员函数的功能是将当前数据表中的元素升序排列。请编写这个 sort 函数。程序的正确输出应为: 。

排序前: 7, 1, 3, 11, 6, 9, 12, 10, 8, 4, 5, 2 排序后: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

要求:

补充编制的内容写在“//\*\*\*\*\*33\*\*\*\*\*”与“//\*\*\*\*\*666\*\*\*\*\*”两行之间。不得修

改程序的其他部分。



注意：程序最后将结果输出到文件。ut.dat 中。输出函数 writeToFile 已经编译为。obj 文件，并且在本程序

调用。

```
//DataList.h

#include using namespace std; Class DataList{//数据表类

int len;

double*d: public:

DataList(int len, double data[]=NULL);。DataList() {delete[]d;}

int length(Constt return len;}//数据表长度(即数据元素的个数) double
getElement(int i)Const{return d[i].}

void sort();//数据表排序

void show()Const;{//显示数据表}

void writeToFile(Char*, Const DataList?);//main.Cpp

#include"DataList.h"

DataList::DataList(int len, double data[]): len(len) { d=new double[len];
for(int i=0;i
d[i]=(data==NULL?0.0: data[i]);
}

void DataList:: sort() {//数据表排序//*****333*****

//*****666*****}

void DataList:: show()Const{//显示数据表

for(int i=0;i
```

```

}

int main() {

double S[]={7, 1, 3, 11, 6, 9, 12, 10, 8, 4, 5, 2}; DataList list(12, S);

Cout<<" 排序前: " ; list.show();

list.sort();

Cout<

writeToFile(" ", list); return 0;

}

```

参考解析:

```

for(int i=0;i
for(int j=i;j
//从 i+1 处遍历数组 d
if(d[i]>d[j])
//d[j] 和 d[i] 比较大小, 如果大于, 就 d[i] 和 d[j] 值交换
{
int temp=d[i];
//把临时整型变量 temp 赋值为 d[i]
d[i]=d[j]; //把 d[j] 赋值给 d[i]
d[j]=temp; //把 temp 值赋给 d[j]
}
}

```