

指向了一个派生类的对象,那么它在调用 func()时,会调用派生类的 func()函数。 7.以下关于构造函数和析构函数描述错误的是: .构造函数可以重载,析构函数不能重载 B. 析构函数可以带或不带参数 C.类的构造函数和析构函数不能被它的派生类继承。 D.类中如果没有定义复制构造函数,则编译器将自动生成一个默认的 8.下面对静态数据成员的描述,正确的是: A.对于一个类的某个静态数据成员,该类的不同对象可以有不同的值**少** ▶.静态数据成员不能通过类的对象调用 C.静态数据成员可以通过初始化列表来初始化 D.静态数据成员可以定义为 private 9.类 TM 和 类 TN 是两个独立的类, 那么类 TM 中哪种形式的成员变量一定是不 允许的 A: TM *a; B: TM &a; C: TM a; D: TN a; 10.下列情况中,不会调用拷贝构造函数的是 A. 用一个对象去初始化同一类的另一个新对象时 B. 将类的一个对象赋值给该类的另一个对象时

3

- C. 函数的形参是类的对象,调用函数进行形参和实参结合时
- D. 函数的返回值是类的对象, 函数执行返回调用时

14.下列哪个运算符可以被重载 A. -> 3. . ?: sizeof



- ₹2.下列关于 STL 的说法错误的是
- A、vector 支持随机访问,在尾部增删有比较好的性能。
- B、list 在内存上是不连续的,不支持随机访问,在两端增删有比较好的性能。
- 🕰 deque 支持随机访问,在两端增删元素有比较好的性能
- D、std::copy 函数可以作用于 vector、list、deque 和 priority_queu



13.以下 STL 中的函数模板哪个可以作用于 set

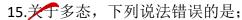
- A. find
- B. random shuffle
- C. sort
- D. 都不行



14. 关于构造函数,错误的说法是

- A. 构造函数中不可以访问类的纯 virtual 函数
- B. 每个类都有无参数的构造函数
- C. 构造函数可以是类的 private 成员
- D. 每个类都有复制构造函数





- **A**. 通过派生类指针访问了基类成员变量,不一定是多态
- B. 通过派生类引用访问了基类成员函数,不一定是多态
- C. 通过"基类指针->函数名()"不能调用派生类的私有虚函数
- D. 在类的构造函数中调用虚函数,不可能是多态





16类模板 T 实例化为两个模板类 A 和 B,则

A A 的友元函数可能也是 B 的友元函数

- B. A 的静态成员变量和 B 的静态成员变量可能在程序中位于同一地址
- C. 可以把 A 的对象赋值给 B 的对象
- D. A 的成员函数中,可以访问 B 的私有成员





17.在程序中,希望定义输出文件流 ofstream 来完成以二进制格式输出到文件 output.txt 尾部,保留 output.txt 的现有数据的工作。应当采用的最正确的打开文件的方式是:

- A. ofstream outFile("output.txt",ios::out);
- B. ofstream outFile("output.txt",ios::app);
- C. ofstream outFile("output.txt",ios::out|ios::binary);
- D. ofstream outFile("output.txt",ios::app|ios::binary);



18.以下关于 C++语言中虚函数的描述错误的是:

- A. 构造函数不可以是虚函数
- B. 析构函数不可以是纯虚函数
- C. 构造函数中不应该调用纯虚函数
- 10. struct 中不可以定义虚函数





- 19.下列说法错误的是
- A、静态成员函数可以为虚函数
- B、每个类可以有多个构造函数但只有一个析构函数
- C、多态的函数调用语句中,函数一定是虚函数
- D 通过"基类指针->成员函数名"的方式调用成员函数,不一定是多态



20.有模板如下:

template << font="">class T>
int func(T op,int n)
{

return op(n);

} 以下 4 项中有几项不可能成为调用 func 模板时的第一个实参? 1) int 类型的变量 2) 对象 3) 函数名 4) 函数指针 A. 0 B. 1

C. 2

D. 3