

类的静态成员函数 4.以下说法错误的是()

个类的 静态成员变量一共就一份,为该类所有对象共享

如果不经转换,则不能把非常量指针赋值给常量指针,也不能把常量指针 赋值给非常量指针

如果A是B的友元类,那么A的成员函数可以访问B的私有成员

友元类之间的关系既不能传递也不能继承

5.下列关于运算符重载的描述中,正确的是()

运算符重载为成员函数时,若参数表中无参数,重载的是一元(单目)运算



一元(单目)运算符只能作为成员函数重载



运算符重载可以改变运算符的优先级



重载自增运算符时,前置形式比后置形式多了一个 int 类型的参数

6.以下关于构造函数的说法不正确的是()



构造函数不可以是 private 类型

对 const 成员、引用类型成员以及没有默认构造函数的成员对象,必须通 初始化列表进行初始化



构造函数可以被重载

除复制构造函数以外,单个形参的构造函数定义了从形参类型到该类类型 一个隐式转换



7.以下关于静态成员的说法正确的是()

静态成员函数中可以使用 this 指针



非静态成员函数中不可以调用静态成员函数



静态成员变量不可以定义为 const 类型



静态成员变量可以在非静态成员函数中访问



8.以下关于容器迭代器操作,合法的是()



. list< int > I(10); list< int >::iteratorit=I.begin()+2;

queue< int > q; sort(q.begin(), q.end());



ector< int > v(5); if (v.begin() < v.end()){cout << "Yes!" << endl;}



set< int > s; int set_size=s.end()-s.begin();



9.任意一个类,析构函数的个数最多是()



不限个数

- B. 1
- C. 2
- D. 3
- 10.下列关于 this 指针的说法,哪个是正确的:
 - this 指针一定指向常量型数据

this 指向的数据不可更改

静态成员函数中也可以使用 this 指针

this 指针本身可直接作为成员函数的返回值

- **11**.下列语句中,将函数 int sum(int x, int y)正确重载的是()
- A. float sum(int x, int y);
- int sum(int a, int b);
- C. float sum(float x, float y);
- D. double sum(int y, int x);
- 12.下列的各类函数中,哪个不是类的成员函数:
- A. 构造函数
- 8. 析构函数
- C. 友元函数
- D. 复制构造函数
- 13.下面对模版的声明,正确的是()
- A. template < T >
- B. template < class T1, T2 >
- C. template < class T1, class T2 >

D. template < class T1; class T2 >

14.下列关于引用的说法,错误的是()

声明引用时,必须同时对其进行初始化

引用只能引用变量

引用不是变量,因此本身不占存储单元

可以通过常引用去修改其引用的内容

15.哪种情况下,一定不会调用类 A 的复制构造函数

函数返回时,函数的返回值是类 A 的对象

调用函数时,函数的某个参数是类 A 的对象

将类A的一个对象赋值给类A的另一个对象

用类 A 的一个对象去初始化的类 A 的另一个对象

16.若基类和派生类均定义了同名的虚函数,则以下情况不可能发生的是()

使用派生类指针调用该虚函数,实际调用了在基类中定义的虚函数

使用派生类指针调用该虚函数,实际调用了在派生类中定义的虚函数

使用基类指针调用该虚函数,实际调用了在基类中定义的虚函数

D. 使用基类指针调用该虚函数,实际调用了在派生类中定义的虚函数

17.下列说法错误的是()

抽象类既可以派生出抽象类,也可以派生出非抽象类

В. 纯虚函数没有函数体

C. 凡是包含虚函数的类都是抽象类

抽象类的构造函数或析构函数内部不能调用自己的纯虚函数 D.

18.下列关于函数模板和类模板的说法,错误的是()













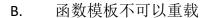






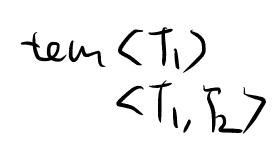






C. 同一个类模板可能生成不同的类

D. 类模板的成员函数也可以是函数模板



19.关于以下函数,有几种可以访问 protected 属性的成员?

类的成员函数

2/ 类的友员函数

) 派生类的成员函数

- A. 0
- B. 1
- C. 2
- D. 3

20.对于有 n 个元素的以下容器, 大多数情况下, 哪种操作速度最快?

在 deque 头部添加一个元素

在 set 中查找元素

在 vector 头部添加一个元素

在 map 中删除元素

