一、概念题

**1．类和对象有什么区别和联系？**

类是一种复杂的数据类型，它是将不同类型的数据和与这些数据相关的操作封装在一起的集合体。类是对某一类对象的抽象，而对象是某一种类的实例。

1. **什么是类的实现？**

将类所有未编写函数体的成员函数在类体外全部编写出来。

1. **this指针的概念是什么？**

类中所有的成员函数（静态成员函数除外）都隐含了第一个参数，这个隐含的第一个参数就是this指针，在成员函数的实现代码中，所有涉及对类的数据成员的操作都隐含为对this指针所指对象的操作,它的值是当前被调用的成员函数所在的对象的起始地址。

1. **为什么要引入构造函数和析构函数？**

构造函数的作用是为类对象的数据成员赋初值，构造函数在定义类对象时由系统自动调用；在一个对象死亡或者说退出生存期时，系统会自动调用析构函数，因此可以在析构函数定义中，设置语句释放该对象所占用的一些资源。

1. **什么时候需要自定义拷贝构造函数？若程序员没有定义拷贝构造函数，则编译器自动生成一个缺省的拷贝构造函数，它可能会产生什么问题？**

当有用一个已经存在对象创建一个同类型的新对象的需求时。当对象含有指针数据成员，并用它初始化同类型的另一个对象时，默认的拷贝构造函数只能将该对象的数据成员复制给另一个对象，而不能将该对象中指针所指向的内存单元也复制过去。这样，就可能出现同一内存单元释放两次，导致程序运行出错。

1. **什么是堆对象？创建和回收堆对象的方法是什么？**

堆区用来存放在程序运行期间，根据需要随时建立的变量（对象），建立在堆区的对象称为堆对象，当堆对象不再使用时，应予以删除，回收所占用的动态内存。创建和回收堆对象的方法是使用new和delete运算符。

1. **为什么需要静态数据成员？静态数据成员的定义和初始化方法是什么？**

定义静态数据成员是为了同一个类的不同对象之间共享公共数据成员；用关键字static可以把数据成员定义成静态数据成员；在定义的类被使用前，要对其中的静态数据成员进行初始化，初始化时不必添加关键字static。

1. **什么是静态成员函数？静态成员函数和非静态成员函数有什么区别？**

用关键字static可以把这成员函数定义成静态成员函数。

⑴ 静态成员函数只能访问类的静态数据成员，不能访问类的非静态数据成员。因为静态成员函数是类的成员，不涉及具体的对象，访问静态成员函数常用类名引导。

⑵ 非静态成员函数可以访问类的静态数据成员，也可以访问非静态数据成员。因为类的静态成员总是存在的。

⑶ 静态成员函数和非静态成员函数，最主要的差别是非静态成员函数隐含了第一个参数this指针，静态成员函数不含this指针。

1. **结构struct和类class有什么异同？**

struct和class都可以定义类，但是缺省访问权限说明时，struct的成员是公有的，而class的成员是私有的。在C++中，struct可被class代替。

1. **比较类的三种继承方式public、protected、private之间的区别。**

⑴采用public公有派生，基类成员的访问权限在派生类中保持不变，即基类所有的公有或保护成员在派生类中仍为公有或保护成员。public派生最常用，可以在派生类的成员函数中访问基类的非私有成员，可通过派生类的对象直接访问基类的公有成员。

⑵采用private私有派生，基类所有的公有和保护成员在派生类中都成为私有成员，只允许在派生类的成员函数中访问基类的非私有成员。private派生很少使用。

⑶采用protected保护派生，基类所有的公有和保护成员在派生类中都成为保护成员，只允许在派生类的成员函数和该派生类的派生类的成员函数中访问基类的非私有成员。

1. **派生类构造函数和析构函数执行的次序是怎样的？**

派生类构造函数的执行次序：首先，调用基类构造函数，调用顺序按照它们被继承时声明的基类名顺序执行；其次，调用内嵌对象构造函数，调用次序为各个对象在派生类内声明的顺序；最后，执行派生类构造函数体中的内容。

派生类析构函数执行过程与构造函数执行过程相反。即当派生类对象的生存期结束时，首先调用派生类的析构函数，然后调用内嵌对象的析构函数，再调用基类的析构函数。

1. **什么叫虚基类？有什么作用？**

在多重继承中，如果多条继承路径上有一个公共的基类，则在这些路径的汇合点上的派生类会产生来自不同路径的公共基类的多个拷贝，如果用virtual把公共基类定义成虚基类，则只会保留公共基类的一个拷贝。引进虚基类的目的是为了解决二义性问题，使得公共基类在它的派生类对象中只产生一个基类子对象。

1. **什么是多态性，在C++中是如何实现多态的？**

多态是指同样的消息被不同类型的对象接收时导致完全不同的行为，是对类的特定成员函数的再抽象。C++支持的多态有多种类型，重载(包括函数重载和运算符重载)和虚函数是其中主要的方式。

1. **什么是抽象类，其有何作用？抽象类的派生类是否一定要给出纯虚函数的实现？**

含有纯虚函数的类称为抽象类。抽象类的主要作用是通过它为一个类族建立一个公共的接口，使它们能够更有效地发挥多态特性。抽象类声明了一组派生类共同操作接口的通用语义，而接口的完整实现，即纯虚函数的函数体，要由派生类自己给出，抽象类只能作为基类被继承使用。抽象类的派生类不一定要给出纯虚函数的实现，没有给出纯虚函数的实现的派生类仍然还是抽象类。

1. **在C++中能否声明虚构造函数，为什么？能否声明虚析构函数，有什么作用？**

在C++中不能声明虚构造函数。多态是不同的对象对同一消息有不同的行为特性，虚函数作为运行过程中多态的基础，主要是针对对象的，而构造函数是在对象产生之前运行的，因此虚构造函数是没有童义的。

在C++中可以声明虚析构函数。析构函数的功能是在该类对象消亡之前进行一些必要的清理工作，如果一个类的析构函数是虚函数，那么，由它派生而来的所有子类的析构函数也是虚函数。析构函数设置为虚函数之后，在使用指针引用时可以动态联编，实现运行时的多态，保证使用基类的指针就能够调用适当的析构函数对不同的对象进行清理工作。

　2． 简述成员函数、全局函数和友元函数的差别。  
　　解答要点：以下几点必须说清楚：  
　　成员函数是在类内部定义的，作用域在类的内部，成员函数可以访问类的数据成员（公有、保护和私有数据成员），可以调用该类的其它成员函数（公有、保护和私有成员函数），可以调用全局函数。如果友元函数是另一个类的公有成员函数，则该类的成员函数也只能通过那个类的对象调用，不能调用那个类的保护和私有成员函数。非本类成员函数（其它类成员函数或全局函数）可以通过该类的对象访问该类的公有数据成员和调用该类的的公有成员函数。  
　　不是在类中定义的成员函数都是全局函数。  
　　如果某一个函数（全局函数或类的成员函数）定义为另一个类的友元函数，需要在那个类中用friend关键字声明，友元函数并不是类的成员，它的定义自然是在那个类的外面。

　3． 简述结构化的程序设计、面向对象的程序设计的基本思想。  
　　解答要点：结构化的程序设计将数据和对数据的操作分离，程序是由一个个的函数组成的，面向对象的程序设计将数据和操作封装在一起，程序是由一个个对象组成的，对象之间通过接口进行通信，它能够较好地支持程序代码的复用。

　4． 结构struct和类class有什么异同？  
　　解答要点：struct和class都可以定义类，但是缺省访问权限说明时，struct的成员是公有的，而class的成员是私有的。在C++中，struct可被class代替。

5．简述变量的生存期和变量作用域有什么区别？

答：变量的生存期是指程序运行后，变量占有内存的时间；变量作用域是指变量声明之后，在程序正文中有效的那部分区域。

6．什么叫形式参数？什么叫实际参数？C++函数参数有什么不同的传递方式？

答：参数是调用函数与被调用函数之间交换数据的通道。函数定义首部的参数称为形式参数，调用函数时使用的参数称为实际参数。C++有三种参数传递机制：值传递（值调用）；指针传递（地址调用）；引用传递（引用调用）。

7．若程序员没有定义拷贝构造函数，则编译器自动生成一个缺省的拷贝构造函数，它可能会产生什么问题？

答：当对象含有指针数据成员，并用它初始化同类型的另一个对象时，缺省的拷贝构造函数只能将该对象的数据成员复制给另一个对象，而不能将该对象中指针所指向的内存单元也复制过去。这样，就可能出现同一内存单元释放两次，导致程序运行出错。

十、 简答题

1、类的公有成员和私有成员有何区别？

C++中的类里的成员函数是可以是私有函数或公有函数。两者的区别是，私有函数只可以在类里面调用，而公有函数其他类也可以调用

私有成员只可以在本类中使用不能在其他类中使用，而公有成员都可以使用

2、引入类的静态成员有何意义？

3、什么是多态性？它能带来什么样的好处？

1.   什么是多态  
多态是C++中的一个重要的基础，可以这样说，不掌握多态就是C++的门外汉。然而长期以来，C++社群对于多态的内涵和外延一直争论不休。大有只见树木不见森林之势。多态到底是怎么回事呢？说实在的，我觉的多态这个名字起的不怎么好（或是译的不怎么好）。要是我给起名的话，我就给它定一个这样的名字--“调用’同名函数’却会因上下文不同会有不同的实现的一种机制”。这个名字长是长了点儿，可是比“多态”清楚多了。看这个长的定义，我们可以从中找出多态的三个重要的部分。一是“相同函数名”，二是“依据上下文”，三是“实现却不同”。嘿，还是个顺口溜呢。我们且把它们叫做多态三要素吧。  
2.   多态带来的好处  
多态带来两个明显的好处：一是不用记大量的函数名了，二是它会依据调用时的上下文来确定实现。确定实现的过程由C++本身完成另外还有一个不明显但却很重要的好处是：带来了面向对象的编程。

4、运算符重载能带来什么好处？

5、怎样理解类与对象的含义？类与对象的关系是什么？

类是现实世界或思维世界中的实体在计算机中的反映，它将数据以及这些数据上的操作封装在一起。

对象是具有类类型的[变量](http://baike.baidu.com/view/296689.htm" \t "_blank)。

类是对象的抽象，而对象是类的具体实例。类是抽象的，不占用内存，而对象是具体的，占用[存储空间](http://baike.baidu.com/view/1375172.htm" \t "_blank)。类是用于创建对象的蓝图，它是一个定义包括在特定类型的对象中的方法和[变量](http://baike.baidu.com/view/296689.htm" \t "_blank)的软件模板。

6、为什么要在基类中定义虚函数？

7、引入友元的目的是什么？

友元(成员)函数特点：定义在类外部，但是需要在类体内进行说明；说明时在前面加上关键字friend class A{public: ...friend (B::)double GetDistance(A start,A stop); .....}; 友元(成员)函数作用：提高程序的运行效率，可以访问类中的保护和私有成员；

友元类特点：class A{...... friend class B; ......};

友元类作用：能够允许一个类中的所有成员函数都能够访问另一个类中的私有成员

具体来说：为了使其他类的成员函数直接访问该类的私有变量即：允许外面的类或函数去访问类的私有变量和保护变量，从而使两个类共享同一函数

优点：能够提高效率，表达简单、清晰

缺点：友元函数破环了封装机制，尽量不使用成员函数，除非不得已的情况下才使用友

元函数。

8、什么是运算符重载？是否所有运算符都能进行重载？

运算作用在复数类对象上应该完成什么功能这就是运算符重载。

重载函数存在于类的继承，是指对你类已有的函数重新定义。重载要注意函数的返回类型及参数表要完全相同。

9、析构函数有什么作用？

解答要点：  
　　对象销毁时，需要调用析构函数。在多态调用时，是用基类的指针访问派生类的对象。如果析构函数是非虚函数，则基类指针只能访问基类的析构函数，而不能访问派生类的析构函数，导致派生类对象销毁时，没有调用派生类的析构函数，只是调用了基类的析构函数。如果把析构函数定义成虚函数，则可克服这个问题。

10、拷贝构造函数在哪几种情况下调用？

解答要点：  
用一个对象初始化另一个对象时  
当用对象作为函数参数传递时  
当函数返回对象时

11、函数重载与函数覆盖有什么不同，它们与多态有什么关系？

解答要点：  
　　函数重载是指函数名相同，而函数的参数个数或类型不同；覆盖是指在派生类中成员函数与基类成员函数的函数名、参数个数、类型与返回值均相同；C++中正是通过虚函数的覆盖，实现多态的功能。

12、C++继承是如何工作的？

解答要点：  
　　继承使得派生类能够使用基类的公有和保护成员，从而实现代码的复用，派生类可以增加成员，也可以隐藏和覆盖基类的成员。对于公有继承，基类成员的访问权限在派生类保持不变。