CS304 - Object Oriented Programming
Quiz No. 3, Feb 07, 2013

A template provides a convenient way to make a family of
Select correct option:
variables and data members
functions and classes
classes and exceptions
programs and algorithms

A class template may inherit from another class template.
Select correct option:
True
False

Target of a _____ function call is determined at run time.
Select correct option:
instance
virtual
operator
none of given

A class hierarchy
Select correct option:
shows the same relationships as an organization chart.
describes "has a" relationships.
describes "is a kind of" relationships.
shows the same relationships as a family tree.

Sender of the message does not need to know the exact class of receiver in_____.
Select correct option:
Abstraction
Polymorphism
Inheritance
none of the given

A function call is resolved at run-time in_____
Select correct option:
non-virtual member function
virtual member function
Both non-virtual member and virtual member function.
None of given

Adding a derived class to a base class requires fundamental changes to the base class.
Select correct option:
True
False

User can make virtual table explicitly.
Select correct option:
True
False

Binding means that target function for a call is selected at compile time.
Select correct option:
Static
Dynamic
Automatic
None of given

Target of a _____ function call is determined at run time.
Select correct option:
instance
virtual
operator
none of given

Which line will produce error. Class phone: private Transmit, private Receiver { } 1. int main() 2. { 3. phone obj; 4. Tranmit* obj1 = &obj; 5. Received obj2 = &obj; 6. }
Select correct option:
3rd line will produce error
4th line will produce error
3rd and 4th line will produce error.
5th line will produce error

Function overriding is done in context of,
Select correct option:
Single class
Single derived class
Single base class
Derived and base classes

Consider the code below, class class1{ public: void func1(); }; class class2 : protected class1 { }; Function func1 of class1 is _____ in class2,

Select correct option:
public
protected
private
none of the given options

the following statements: 1) int iArray[5]; 2) int *pArr = iArray;
Select correct option:
These statements will compile successfully
Error in first statement
Error in second statement
None of given options

Methodologies to the development of reusable software relate to_____.
Select correct option:
Structure programming
procedural programming
generic programming
None of the given

function template must have a parameter.
Select correct option:
True
False

The default inheritance mode is,
Select correct option:
Public inheritance
Protected inheritance
Private inheritance
None of these options

Two functions with same names, parameters and return type can exist in,
Select correct option:
Function overloading
Function overriding
Operator overloading
None of these options

Consider the code below, class c1{ }; class c2 : public c1 { }; class c3 : public c2 { }; Then c2 is,
Select correct option:

Direct base class of c3

Direct child class of c3
Direct base class of c1
None of these

Virtual functions allow you to
Select correct option:
create an array of type pointer-to-base class that can hold pointers to derived classes.
create functions that can never be accessed.
group objects of different classes so they can all be accessed by the same function code.
use the same function call to execute member functions of objects from different classes.

User can make virtual table explicitly.
Select correct option:
True
False

In order to define a class template the first line of definition must be:
Select correct option:
template <typename T>
typename <template T>
Template Class <ClassName>
Class <Template T>

Consider the following statements: 1) int iArray[5]; 2) int *pArr = iArray;
Select correct option:
These statements will compile successfully
Error in first statement
Error in second statement
None of given options

In c++ dynamic binding and polymorphism will be achieved when member function will be __ .
Select correct option:
private
public
virtual
inline

In type in depended function template should be use where code and behavior must be identical.
Select correct option:

True
False

Consider the code below, class class1{ protected: int i; }; class class2 : private class1 { };
Then int member i of class1 is _____ in class2,
Select correct option:

public
protected
private
none of the given options

In specialization we can,
Select correct option:
Replace child class with its base class
Replace base class with its child class (Not Sure)
Replace both child and base classes interchangeably
None of the given options

Consider the code below, class class1{ public: void func1(); }; class class2 : public class1 {
}; Function func1 of class1 is _____ in class2,
Select correct option:
public
protected
private
none of the given options

It is illegal to make objects of one class members of another class.
Select correct option:
True
False

An abstract class is useful when
Select correct option:
no classes should be derived from it.
there are multiple paths from one derived class to another.
no objects should be instantiated from its.
you want to defer the declaration of the class.

In resolution order compiler search firstly _____.
Select correct option:
Generic Template
Partial Specialization

Complete Specialization
Ordinary function

template<> class Vector{ void** p; //.... void*& operator[] ((int i); };
Select correct option:
This specialization can then be used as the common implimentation for all Vectors of pointers.
This spcialization can then be used as the all type implimentation for one type classes.
This specialization can then be used double type pointers.
This specialization should be used for Vectors of all type int types.

In private inheritance derived class pointer can be assigned to base class pointer in,
Select correct option:
Main function
In derived class member and friend functions
In base class member and friend functions
None of the given options

Which statement will be true for concrete class?
it implements an virtual concept.
it can be instantiated
it cannot be instantiated
none of given

Target of a _____ function call is determined at run time.
Select correct option:
instance
virtual
operator
none of given

The Specialization pattern after the name says that this specialization is to be used for every___.
Select correct option:
data types
meta types
virtual types
pointers type

c++ dynamic binding and polymorphism will be achieved when member function will be __.
Select correct option:

private
public
virtual
inline

Consider the code below, class class1{ protected: void func1(); }; class class2 : public class1 { }; Function func1 of class1 is _____ in class2,
Select correct option:
public
protected
private
none of the given options

Consider the code below, class class1{ protected: int i; }; class class2 : protected class1 { }; Then int member i of class1 is _____ in class2,
Select correct option:
public
protected
private
none of the given options

Consider the code below, class class1{ private: void func1(); }; class class2 : private class1 { }; Function func1 of class1 is _____ in class2,
Select correct option:
public
protected
private
none of the given options

Target of a _____ function call is determined at run time.
Select correct option:
instance
virtual
operator
none of given

Consider the following statements: 1) int iArray[5]; 2) int *pArr = iArray;
Select correct option:
These statements will compile successfully
Error in first statement
Error in second statement
None of given options

Consider the code below, class class1{ private: int i; }; class class2 : private class1 { };
Then int member i of class1 is _____ in class2,
Select correct option:
public
protected
private
none of the given options

If there is a pointer, p, to objects of a base class, and it contains the address of an object
of a derived class, and both classes contain a virtual member function, ding(), then the
statement p->ding(); will cause the version of ding() in the ___ class to be executed.
Select correct option:
base
derived
virtual
implemented

A class template may inherit from another class template.
Select correct option:
True
False

Derived class can inherit from public base class as well as private and protected base
classes
Select correct option:
True
False

Two functions with same names, parameters and return type can exist in,
Select correct option:
Function overloading
Function overriding
Operator overloading
None of these options

Consider the code below, class class1{ private: int i; }; class class2 : public class1 { }; Then
int member i of class1 is _____ in class2,

Select correct option:
public
protected
private
none of the given options

Target of a _____ function call is determined at run time.
Select correct option:
instance
virtual
operator
none of given

A class template may inherit from another class template.
Select correct option:
True
False

A function call is resolved at run-time in_____
Select correct option:
non-virtual member function
virtual member function
Both non-virtual member and virtual member function.
None of given
hello

A class hierarchy
Select correct option:
shows the same relationships as an organization chart.
describes "has a" relationships.
describes "is a kind of" relationships.
shows the same relationships as a family tree.

Consider the code below, class class1{ public: int i; }; class class2 : public class1 { }; Then int member i of class1 is _____ in class2,
Select correct option:
public
protected
private
none of the given options

Consider the code below, class c1{ }; class c2 : public c1 { }; class c3 : public c2 { }; Then c1 is,
Select correct option:
Direct base class of c3
Direct child class of c3
Direct base class of c2
Direct child class of c2

A class can inherit from more then one class is called.
Select correct option:
Simple inheritance
Multiple inheritances
Single inheritance
Double inheritance

template<> class Vector{ void** p; //.... void*& operator[] ((int i); };
Select correct option:

This specialization can then be used as the common implimentation for all Vectors of pointers.
This spcialization can then be used as the all type implimentation for one type classes.
This specialization can then be used double type pointers.
This specialization should be used for Vectors of all type int types.

Consider the code below, class class1{ public: int i; }; class class2 : protected class1 { };
Then int member i of class1 is _____ in class2,
Select correct option:
public
protected
private
none of the given options

Consider the code below, class class1{ private: void func1(); }; class class2 : public class1 { }; Function func1 of class1 is _____ in class2,
Select correct option:
public
protected
private
none of the given options

Templates automatically create different versions of a function, depending on user input.
Select correct option:
True
False

_____ Binding means that target function for a call is selected at run time
Select correct option:
Automatic
Dynamic
Static
Dramatic

When we create objects, then space is allocated to:
Member function
Access specifier
Data member
None of given

There is only one form of copy constructor.
True
False

Which of the following features of OOP is used to deal with only relevant details?
Abstraction
Information hiding
Object

_____ Binding means that targets function for a call is selected at compile time.
Static
Dynamic
Automatic
None of given

A Class hierarchy
Shows the same relationships as an organization chart
Describes "has a" relationships.
Describes "is a kind of" relationships.
Shows the same relationships as a family tree

In C++, we declare a function virtual by preceding the function header with keyword
"Inline"
True
False

It is illegal to make objects of one class members of another class.
True
False

In Resolution order compiler search firstly_____.
Generic Template
Partial Specification
Complete Specification
Ordinary function

Derived class can inherit from public base class as well as private and protected base classes
True
False

Which line will produce error. Class phone: Private Transmit, private Receiver { } 1.int main () 2. { 3.phone obj; 4.Tranmit*obj1 = &obj; 5.Received obj2 = &obj; 6.}

3rd line will produce error
4th line will produce error
3rd and 4th line will produce error.
5th line will produce error.

Methodologies to the development of reusable software relate to _____.
Structure programming
Procedural programming
Generic programming
None of the given

A template argument is preceded by the keyword_____.
Vector
Class
Template
Type*

Friends are used exactly the same for template and non-template classes.
True
False

A function template must have a parameter
True
False

Child class can call constructor of its,
Direct base class
Indirect base class
Both direct and indirect base classes
None of these.

Which statement will be true for concrete class?
It implements an virtual concept.

It can be instantiated
It cannot be instantiated
None of given

A class D can be derived from a class C, which is derived froma class B, which is derived from a class A
True
False

Adding a derived class to a base class requires fundamental changes to the base class.
True
False

A Class or class template can have member _____ that are themselves templates.

Variable
Function
Objects
None of given

Which will be the Primary task or tasks of generic programming?
Categorize the abstractions in a domain into concepts
Implement generic algorithms based on the concepts
Build concrete models of the concepts
All of given

The default inheritance mode is,
Public inheritance
Protected Inheritance
Private Inheritance
None of these options

If there is a pointer, p, to objects of a base class, and it contains the address of an object of a derived class, and both classes contain a virtual member function, ding(), then the statement p->ding(); will cause the version of ding() in the _____class to be executed.
Base
Derived
Virtual
Implemented

A class template _____
Facilitates reuse of class

Does not facilitate reuse of class

Sender of the message does not need to know the exact class of receiver in_____.
Abstraction
Polymorphism
Inheritance
none of the given

Friend Functions of a class are _____ members of that class.
Select correct option:
Public
Private ---
Protected
None of the given options.

Consider the code below, class class1{ protected: int i; }; class class2 : protected class1 {
}; Then int member i of class1 is _____ in class2,
Select correct option:
 public
 protected
 private
 none of the given options

A Child class can call constructor of its parent class through,
Select correct option:
Its constructor initialization list
Its constructor body
Both from its constructor initialization list or body --------
Can not call the constructor of its parent class

We can access private members of the class from outside the class using _____
operator with its object pointer.
Select correct option:
->
.
&
None of the given options

A function call is resolved at run-time in_____
Select correct option:
non-virtual member function

virtual member function
Both non-virtual member and virtual member function.
None of given

It is sometimes useful to specify a class from which no objects will ever be created.
Select correct option:
True
False

Consider the code below, class class1{ public: void func1(); }; class class2 : private class1 { }; Function func1 of class1 is _____ in class2,
Select correct option:
public
protected
private
none of the given options

Consider the code below, class class1{ public: int i; }; class class2 : protected class1 { };
Then int member i of class1 is _____ in class2,
Select correct option:
public
 protected
 private
 none of the given options

Static casting is,
Select correct option:
 Implicit way of calling base class functions from derived class
Explicit way of calling base class functions from derived class
None of these
Both of these

Consider the code below, class class1{ protected: void func1(); }; class class2 : public class1 { }; Function func1 of class1 is _____ in class2,
Select correct option:
public
protected
private
none of the given options

Consider the code below, class class1{ public: void func1(); }; class class2 : public class1 { }; Function func1 of class1 is _____ in class2
Public

Protected
Private
none of the given options

Consider the code below, class class1{ protected: int i; }; class class2 : public class1 { };
Then int member i of class1 is _____ in class2,
public
protected
private
none of the given options

In Private Inheritance the public members of base class become _____ in derived class.
Public
Private
Protected
None of the given options.

Consider the following two lines of code written for a class Student, 1. Student sobj1; 2. Student sobj2(sobj1); In line No.2 what constructor of Student class will be called,
Default constructor of Student class
Copy constructor of student class
Both default and copy constructer of Student class
No constructor will be called