

1. In the statement `class car : public vehicle`, which is the base class?
a. Car **b. vehicle** c. public d. class
2. _____ allows us to create new classes based on existing classes.
a. Polymorphism **b. Inheritance** c. Function overloading d. The copy constructor
3. When you derive a class from an existing class, you _____ add new data and functions.
a. may never b. may sometimes **c. may** d. None of these
4. _____ members of a base class are never accessible to a derived class.
a. Public **b. Private** c. Protected d. a, b, and c e. None of these
5. The _____ constructor is called before the _____ constructor.
a. base, derived b. derived, base c. public, private d. private, public
6. A _____ of a base class expects to be overridden in a derived class.
a. constructor function b. destructor function c. static function **d. virtual function**
7. The term _____ means the ability to take many forms.
a. inheritance **b. polymorphism** c. member function d. encapsulation
8. _____ to a base class may be assigned the address of a derived class object.
a. Access specifiers b. Static members c. Private members **d. Pointers**
9. The base class access specification determines how _____ members in the base class may be accessed by derived classes.
a. Public b. Private c. Protected **d. a, b, and c** e. None of these
10. When a derived class has two or more base classes, the situation is known as _____.
a. multiple inheritance b. polymorphism c. encapsulation d. access specification
11. Arguments are passed to the base class destructor function by the _____ class _____ function.
a. derived, constructor b. derived, destructor c. base, constructor
d. base, destructor **e. None of these**
12. In an inheritance situation, the new class that you create from an existing class is known as the _____.
a. derived class b. inheritee c. child class **d. a and c** e. None of these
13. The base class's _____ affects the way its members are inherited by the derived class.
a. name b. return data type **c. access specification** d. All of these
14. Protected members of a base class are like _____, but they may be accessed by derived classes.
a. constructor functions b. static members **c. private members** d. public members

15. The following statement: `class Car : private Vehicle`
allows the _____ members of the Car class to access _____ members of the Vehicle class.
- a. private, private b. public, private c. protected, private
d. public, protected e. None of these
16. The _____ destructor is called before the _____ destructor.
- a. base, derived b. derived, base c. public, private d. private, public
17. _____ is commonly used to extend a class, or to give it additional capabilities.
- a. Inheritance b. Privacy c. The constructor d. The destructor
18. When member functions behave differently, depending upon which object performed the call, this is an example of _____.
- a. chaos theory b. virtual insubordination c. polymorphism d. encapsulation
19. A virtual function is a member function that expects to be _____ in a derived class.
- a. ignored b. called frequently c. overridden d. private e. None of these
20. A virtual function is declared by placing the keyword _____ in front of the return type in the base class's function declaration.
- a. virtual b. private c. public d. protected e. None of these
21. In the following statement: `class Car : protected Vehicle`
which is the derived class?
- a. Car b. Vehicle c. protected d. cannot be determined e. None of these
22. In the following statement: `class car : protected vehicle`
what is being protected?
- a. derived class functions b. base class members
c. derived class data d. future inherited classes
23. Polymorphism is when _____ in a class hierarchy perform differently, depending upon which object performs the call.
- a. base class constructors b. member functions
c. derived class destructors d. derived class constructors
24. _____ functions are dynamically bound by the compiler.
- a. Constructor b. Destructor c. Static d. Virtual e. None of these

TRUE/FALSE

25. **True/False:** In an inheritance situation, you may not pass arguments to a base class constructor.
26. **True/False:** More than one class may be derived from a base class.

27. **True/False:** A derived class may become a base class, if another class is derived from it.
28. **True/False:** The base class access specification can be viewed as a filter that base class members must pass through when becoming inherited members of a derived class.
29. **True/False:** When arguments must be passed to the base class constructor, they are passed from the derived class constructor's header line.
30. **True/False:** A member function of a derived class may not have the same name as a member function of a base class.
31. **True/False:** Static binding occurs when the compiler binds a function call with the function call that resides in the same class as the call itself.
32. **True/False:** Pointers to a base class may be assigned the address of a derived class object.
33. **True/False:** A derived class may not have any classes derived from it.