1. What is hybrid inheritance?
   1. A type of inheritance that involves multiple base classes and multiple derived classes.
   2. A type of inheritance where a class can inherit from both another class and an interface.
   3. **A combination of multiple inheritance and single inheritance.**
   4. A type of inheritance that only involves one base class and multiple derived classes.
2. In C++, how is hierarchical inheritance achieved?
   1. By inheriting from multiple base classes.
   2. **By inheriting from a single base class.**
   3. By inheriting from more than one derived class.
   4. By inheriting from multiple unrelated classes.
3. Which inheritance type is prone to the "diamond problem" in C++?
   1. Hybrid inheritance
   2. Hierarchical inheritance
   3. **Multiple inheritance**
   4. Single inheritance
4. The "diamond problem" in C++ arises due to:
   1. Inheriting from multiple base classes.
   2. Using virtual functions in the base class.
   3. Having more than one derived class.
   4. **Ambiguity in method resolution with multiple inheritance.**
5. In hybrid inheritance, a class can inherit from:
   1. Only one base class and one derived class.
   2. **Two or more unrelated base classes.**
   3. One base class and multiple derived classes.
   4. Only two base classes.
6. Which access specifier is used by default in C++ for a class declared using class keyword?
   1. public
   2. **private**
   3. protected
   4. friend
7. In C++, which keyword is used to inherit publicly from a base class?
   1. **public**
   2. inherit
   3. extends
   4. using
8. When using hierarchical inheritance, how many classes are derived from a single base class?
   1. One
   2. Two
   3. Three
   4. **It depends on the design.**
9. What is the primary advantage of using hierarchical inheritance?
   1. **Code reusability**
   2. Method overriding
   3. Data encapsulation
   4. Multiple inheritance
10. In C++, when a class inherits privately from a base class, which members of the base class are accessible in the derived class?
    1. Public members only
    2. Protected members only
    3. **Private members only**
    4. All members
11. The ambiguity problem in multiple inheritance occurs when:
    1. Two or more base classes have the same member variables.
    2. **Two or more base classes have the same member functions.**
    3. A derived class has multiple constructors.
    4. The derived class is inherited privately from the base class.
12. Which keyword is used to resolve ambiguity in multiple inheritance?
    1. super
    2. **base**
    3. this
    4. virtual
13. Which access specifier in C++ allows the derived class to access the public members of the base class?
    1. public
    2. private
    3. **protected**
    4. friend
14. What is the main advantage of multiple inheritance in C++?
    1. Improved code readability
    2. Easier class hierarchy management
    3. **Enhanced code reusability**
    4. Reduced memory consumption
15. When does ambiguity occur in hierarchical inheritance?
    1. When two or more derived classes have the same method names.
    2. **When two or more base classes have the same method names.**
    3. When a class inherits from multiple base classes.
    4. When a class is derived from a single base class.
16. How can ambiguity be resolved in multiple inheritance?
    1. By using the friend keyword
    2. By using the virtual keyword
    3. **By using the scope resolution operator (::)**
    4. By using the using keyword
17. Which inheritance type is used to model "is-a" relationships in C++?
    1. Multiple inheritance
    2. Hierarchical inheritance
    3. Hybrid inheritance
    4. **Single inheritance**
18. Which type of inheritance allows a class to inherit from multiple base classes?
    1. Single inheritance
    2. **Multiple inheritance**
    3. Hierarchical inheritance
    4. Hybrid inheritance
19. What is the main drawback of multiple inheritance in C++?
    1. **It increases code complexity.**
    2. It leads to memory leaks.
    3. It is not supported by the C++ standard.
    4. It can only be used with abstract classes.
20. In C++, can a class inherit from multiple base classes with the same name member function?
    1. Yes, as long as the member functions have the same return type.
    2. Yes, as long as the member functions have different return types.
    3. **No, it will cause ambiguity.**
    4. No, it is not allowed by the C++ compiler.
21. How can the diamond problem in C++ be resolved?
    1. **By using the virtual keyword when inheriting classes.**
    2. By avoiding multiple inheritance altogether.
    3. By using the scope resolution operator (::) to specify which version of the function to use.
    4. By using the friend keyword to grant access to the base class's members.
22. How many specifiers are used to derive a class?
    1. 1
    2. 2
    3. **3**
    4. 4
23. What is a virtual function in C++?
    1. Any member function of a class
    2. All functions that are derived from the base class
    3. All the members that are accessing base class data members
    4. **All the functions which are declared in the base class and is re-defined/overridden by the derived class**
24. Which is the correct syntax of declaring a virtual function?
    1. **virtual int func();**
    2. virtual int func(){};
    3. inline virtual func();
    4. inline virtual func(){};
25. Which statement is incorrect about virtual function.
    1. They are used to achieve runtime polymorphism
    2. **They are used to hide objects**
    3. Each virtual function declaration starts with the virtual keyword
    4. All of the mentioned
26. The concept of deciding which function to invoke during runtime is called \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
    1. late binding
    2. dynamic linkage
    3. static binding
    4. **both late binding and dynamic linkage**
27. What is a pure virtual function?
    1. A virtual function defined inside the base class
    2. **A virtual function that has no definition relative to the base class**
    3. A virtual function that is defined inside the derived class
    4. Any function that is made virtual
28. What is the order of Destructors call when the object of derived class B is declared, provided class B is derived from class A?
    1. Destructor of A followed by B
    2. **Destructor of B followed by A**
    3. Destructor of A only
    4. Destructor of B only
29. Which concept of OOPs is shown by Virtual Functions?
    1. Inheritance
    2. Encapsulation
    3. **Polymorphism**
    4. Abstraction
30. Pick the correct statement.
    1. Virtual function can have different names in the base and derived class
    2. Virtual function cannot be applied in Multiple Inheritance classes
    3. **Virtual function are different in definitions only**
    4. Virtual function does early bindings