C++ Concepts Quiz Questions

Class and Object

1. Which of the following is true about a class in C++?
   a) It is a template for objects.
   b) It is an instance of an object.
   c) It allocates memory for objects.
   d) It cannot contain methods.
   Answer: a) It is a template for objects.

2. What does an object represent in C++?
   a) Blueprint for a class.
   b) Instance of a class.
   c) A set of random values.
   d) Only functions of a class.
   Answer: b) Instance of a class.

Constructor and Destructor

3. Which of the following is true about constructors?
   a) They are called explicitly.
   b) They have no return type.
   c) They can be inherited.
   d) They must be virtual.
   Answer: b) They have no return type.

4. Which type of constructor is called when an object is copied?
   a) Default Constructor
   b) Parameterized Constructor
   c) Copy Constructor
   d) Destructor
   Answer: c) Copy Constructor

5. Which of the following is correct about destructors?
   a) A class can have multiple destructors.
   b) A destructor has a return type.
   c) A destructor cannot be virtual.
   d) It is invoked automatically when the object goes out of scope.
   Answer: d) It is invoked automatically when the object goes out of scope.

Friend Function and Friend Class

6. A friend function is:
   a) A member of the class.
   b) Defined outside the class but has access to private members.
   c) Used to define a virtual function.
   d) A function that cannot access static members.
   Answer: b) Defined outside the class but has access to private members.

7. A friend class:
   a) Can inherit the private members of another class.
   b) Can access private and protected members of another class.
   c) Must be declared inside the public section of the class.
   d) Is used to implement polymorphism.
   Answer: b) Can access private and protected members of another class.

Inline Functions

8. Inline functions are expanded:
   a) During runtime.
   b) During compile time.
   c) During linking.
   d) Only for virtual functions.
   Answer: b) During compile time.

9. Which of the following is false about inline functions?
   a) They reduce function call overhead.
   b) They can be recursive.
   c) The inline keyword is just a suggestion to the compiler.
   d) Inline functions can increase the size of the binary file.
   Answer: b) They can be recursive.

Inheritance

10. Inheritance allows:
    a) Overriding static functions.
    b) Code reusability and extensibility.
    c) Accessing private data members of the base class directly.
    d) None of the above.
    Answer: b) Code reusability and extensibility.

11. Which of the following is NOT a type of inheritance in C++?
    a) Single
    b) Double
    c) Hierarchical
    d) Multiple
    Answer: b) Double

Polymorphism

12. Polymorphism in C++ means:
    a) Many classes in one program.
    b) Multiple functions with the same name.
    c) One name, many forms.
    d) None of the above.
    Answer: c) One name, many forms.

13. Runtime polymorphism is achieved by:
    a) Function overloading
    b) Function overriding
    c) Operator overloading
    d) All of the above
    Answer: b) Function overriding

Abstraction and Encapsulation

14. Abstraction focuses on:
    a) Implementation details.
    b) Hiding implementation and showing functionality.
    c) Encapsulating data.
    d) Both b and c.
    Answer: b) Hiding implementation and showing functionality.

15. Encapsulation ensures:
    a) The functions cannot modify private data.
    b) The data members are accessible by any function.
    c) Data hiding and security.
    d) None of the above.
    Answer: c) Data hiding and security.

Static Members and Static Functions

16. Static members of a class are:
    a) Shared among all objects of the class.
    b) Unique for each object.
    c) Always private.
    d) Cannot be modified.
    Answer: a) Shared among all objects of the class.

17. Which of the following is true about static functions?
    a) They can access both static and non-static members.
    b) They can only access static members.
    c) They are called using object names.
    d) None of the above.
    Answer: b) They can only access static members.

Exception Handling

18. Which keyword is used to throw an exception?
    a) throw
    b) try
    c) catch
    d) finally
    Answer: a) throw

19. The correct sequence of exception handling is:
    a) throw → finally → catch
    b) try → throw → catch
    c) catch → throw → finally
    d) None of the above.
    Answer: b) try → throw → catch

Templates

20. Templates are used to:
    a) Reduce code redundancy.
    b) Replace macros.
    c) Make the program faster.
    d) Only for arrays.
    Answer: a) Reduce code redundancy.

21. Which is true about templates?
    a) They can only be used with integers.
    b) They are evaluated at runtime.
    c) They provide type safety.
    d) None of the above.
    Answer: c) They provide type safety.

Upcasting and Downcasting

22. Upcasting refers to:
    a) Converting a base class pointer to a derived class pointer.
    b) Converting a derived class pointer to a base class pointer.
    c) Casting a void pointer.
    d) Implicit conversion.
    Answer: b) Converting a derived class pointer to a base class pointer.

23. Downcasting requires:
    a) A virtual destructor.
    b) An explicit cast.
    c) A default constructor.
    d) None of the above.
    Answer: b) An explicit cast.

Class and Object

1. A class in C++ is defined using which keyword?
    a) struct
    b) class
    c) object
    d) template
    Answer: b) class

2. Which of the following is true about private members of a class?
    a) They can only be accessed within the same class.
    b) They can be accessed by any function.
    c) They are automatically inherited by derived classes as public.
    d) None of the above.
    Answer: a) They can only be accessed within the same class.

3. How is a class instantiated in C++?
    a) Using a `new` keyword.
    b) By declaring an object of the class type.
    c) By directly calling the class name.
    d) None of the above.
    Answer: b) By declaring an object of the class type.

Constructor and Destructor

4. Which of the following constructors initializes an object with predefined values?
   a) Default Constructor
   b) Parameterized Constructor
   c) Dynamic Constructor
   d) None of the above
   Answer: b) Parameterized Constructor

5. Can a destructor be overloaded?
   a) Yes, it can have multiple forms.
   b) No, a class can have only one destructor.
   c) Yes, but only if it's public.
   d) None of the above.
   Answer: b) No, a class can have only one destructor.

6. Which symbol is used to declare a destructor?
   a) `->`
   b) `~`
   c) `*`
   d) `#`
   Answer: b) `~`

Friend Function and Friend Class

7. What is the key benefit of friend functions?
   a) They improve code performance.
   b) They allow encapsulation to be broken when necessary.
   c) They simplify inheritance.
   d) None of the above.
   Answer: b) They allow encapsulation to be broken when necessary.

8. Can a friend function access private and protected members?
   a) Yes, if explicitly declared a friend.
   b) No, they can only access public members.
   c) Yes, but only in the same file.
   d) None of the above.
   Answer: a) Yes, if explicitly declared a friend.

Inline Functions

9. What is the main advantage of using inline functions?
   a) Increased readability.
   b) Reduced function call overhead.
   c) Faster compilation.
   d) None of the above.
   Answer: b) Reduced function call overhead.

10. When does the compiler ignore the `inline` keyword?
    a) When the function is recursive.
    b) When the function is declared outside the class.
    c) When the function is too large.
    d) Both a and c.
    Answer: d) Both a and c.

Inheritance

11. How is private inheritance declared in C++?
    a) `class Derived : public Base`
    b) `class Derived : private Base`
    c) `class Derived : protected Base`
    d) `class Derived : Base`
    Answer: b) `class Derived : private Base`

12. Which of the following is true for protected members in inheritance?
    a) They become private in the derived class.
    b) They can be accessed by any function in the derived class.
    c) They are inherited as protected in derived classes.
    d) None of the above.
    Answer: c) They are inherited as protected in derived classes.

13. Multiple inheritance allows:
    a) A derived class to inherit from multiple base classes.
    b) A base class to inherit from multiple derived classes.
    c) Only one base class with multiple derived classes.
    d) None of the above.
    Answer: a) A derived class to inherit from multiple base classes.

Polymorphism

14. Which of the following is an example of compile-time polymorphism?
    a) Function overloading
    b) Virtual functions
    c) Dynamic method invocation
    d) None of the above
    Answer: a) Function overloading

15. Dynamic polymorphism is implemented using:
    a) Virtual functions
    b) Overloaded operators
    c) Static members
    d) None of the above
    Answer: a) Virtual functions

16. A pure virtual function is declared by:
    a) Appending `= 0` in the declaration.
    b) Using the keyword `virtual`.
    c) Adding `abstract` before the function name.
    d) Declaring the function inside private access.
    Answer: a) Appending `= 0` in the declaration.

Abstraction and Encapsulation

17. What is achieved by encapsulation?
    a) Restricting access to some data.
    b) Increasing memory efficiency.
    c) Improving the execution time.
    d) None of the above.
    Answer: a) Restricting access to some data.

18. Which of the following best defines abstraction?
    a) Showing essential features while hiding details.
    b) Defining multiple classes.
    c) Using function pointers.
    d) None of the above.
    Answer: a) Showing essential features while hiding details.

Static Members and Static Functions

19. What is true about static data members?
    a) They are initialized to zero by default.
    b) They must be defined outside the class.
    c) They are shared among all objects.
    d) All of the above.
    Answer: d) All of the above.

20. Static functions:
    a) Can modify non-static members.
    b) Do not require an object to be called.
    c) Must return a value.
    d) Are always private.
    Answer: b) Do not require an object to be called.

Exception Handling

21. Which of the following blocks must be present for exception handling?
    a) try
    b) catch
    c) both try and catch
    d) finally
    Answer: c) both try and catch

22. What is the role of the `catch` block?
    a) Execute code when no exception is thrown.
    b) Handle exceptions thrown by the `try` block.
    c) Re-throw an exception.
    d) None of the above.
    Answer: b) Handle exceptions thrown by the `try` block.

Templates

23. Class templates are primarily used for:
    a) Allowing multiple inheritance.
    b) Writing generic classes.
    c) Optimizing runtime execution.
    d) None of the above.
    Answer: b) Writing generic classes.

24. Which keyword is used to define a template?
    a) template
    b) typename
    c) generic
    d) function
    Answer: a) template

Upcasting and Downcasting

25. What is upcasting?
    a) Converting a derived object to a base class reference.
    b) Converting a base class pointer to a derived object.
    c) Implicitly changing the data type.
    d) None of the above.
    Answer: a) Converting a derived object to a base class reference.

26. Which of the following is required for safe downcasting?
    a) Static cast
    b) Dynamic cast
    c) Explicit destructor
    d) None of the above.
    Answer: b) Dynamic cast

Class and Object

1. The size of an empty class in C++ is:
   a) 0 bytes
   b) 1 byte
   c) Depends on the compiler
   d) Undefined
   Answer: b) 1 byte

2. Which of the following can be declared inside a class?
   a) Functions
   b) Data members
   c) Constructor and Destructor
   d) All of the above
   Answer: d) All of the above

3. Which of the following is true about an object?
   a) It is a real-world entity.
   b) It encapsulates both data and functions.
   c) It is an instance of a class.
   d) All of the above
   Answer: d) All of the above

Constructor and Destructor

4. What happens if no constructor is explicitly defined for a class?
   a) The class cannot be instantiated.
   b) A default constructor is provided by the compiler.
   c) An error occurs during compilation.
   d) None of the above.
   Answer: b) A default constructor is provided by the compiler.

5. Which of the following statements is true?
   a) A constructor can be virtual.
   b) A destructor can be overloaded.
   c) A destructor can be virtual.
   d) None of the above.
   Answer: c) A destructor can be virtual.

6. Destructors are used primarily for:
   a) Allocating memory.
   b) Deleting objects.
   c) Releasing resources like memory and files.
   d) Modifying data members.
   Answer: c) Releasing resources like memory and files.

Friend Function and Friend Class

7. Can a friend function access private static members of a class?
   a) Yes
   b) No
   c) Only if declared public
   d) Only in the same class
   Answer: a) Yes

8. A friend class can be declared:
   a) Anywhere in the program.
   b) Only inside the private section of the class.
   c) Inside the public or private section of the class.
   d) None of the above.
   Answer: c) Inside the public or private section of the class.

Inline Functions

9. What is a limitation of inline functions?
   a) They must be defined inside the class.
   b) They cannot be recursive.
   c) They can only have a single line of code.
   d) None of the above.
   Answer: b) They cannot be recursive.

10. Inline functions can be declared:
    a) Only in the header file.
    b) Only in the source file.
    c) In both header and source files.
    d) None of the above.
    Answer: a) Only in the header file.

Inheritance

11. Which inheritance type is used to derive a class from multiple base classes?
    a) Multiple inheritance
    b) Hierarchical inheritance
    c) Multilevel inheritance
    d) None of the above
    Answer: a) Multiple inheritance

12. When a base class function is redefined in the derived class, it is known as:
    a) Overloading
    b) Overriding
    c) Overwriting
    d) None of the above
    Answer: b) Overriding

13. Which of the following access specifiers restricts the inheritance to private members only?
    a) private
    b) protected
    c) public
    d) None of the above
    Answer: a) private

Polymorphism

14. What is achieved by operator overloading in C++?
    a) Runtime polymorphism
    b) Compile-time polymorphism
    c) Data abstraction
    d) None of the above
    Answer: b) Compile-time polymorphism

15. Which of the following functions are eligible for overriding?
    a) Static functions
    b) Virtual functions
    c) Inline functions
    d) None of the above
    Answer: b) Virtual functions

16. What is the purpose of the `virtual` keyword in C++?
    a) To declare abstract functions.
    b) To ensure runtime polymorphism.
    c) To allow multiple inheritance.
    d) None of the above.
    Answer: b) To ensure runtime polymorphism.

Abstraction and Encapsulation

17. Abstract classes are used:
    a) To create instances.
    b) As base classes for derived classes.
    c) To define concrete methods.
    d) None of the above.
    Answer: b) As base classes for derived classes.

18. Encapsulation ensures:
    a) Data hiding.
    b) Secure communication between objects.
    c) Reduced dependency between objects.
    d) All of the above.
    Answer: d) All of the above.

Static Members and Static Functions

19. Which of the following is not true for static members?
    a) They are shared among all objects.
    b) They are initialized only once.
    c) They cannot be modified after initialization.
    d) They are accessed using the class name.
    Answer: c) They cannot be modified after initialization.

20. Static functions:
    a) Are specific to a single object.
    b) Do not have access to non-static members.
    c) Must return a value.
    d) Cannot be called outside the class.
    Answer: b) Do not have access to non-static members.

Exception Handling

21. Which keyword is used to define an exception handler?
    a) `try`
    b) `catch`
    c) `throw`
    d) `finally`
    Answer: b) `catch`

22. How can you rethrow an exception?
    a) Using `throw` inside a catch block.
    b) Using `try` inside a catch block.
    c) Using `finally` inside a catch block.
    d) None of the above.
    Answer: a) Using `throw` inside a catch block.

23. What happens if an exception is not caught?
    a) The program continues execution.
    b) The program terminates abnormally.
    c) The compiler generates a warning.
    d) None of the above.
    Answer: b) The program terminates abnormally.

Templates

24. Which type of template allows using a class as a template argument?
    a) Function template
    b) Class template
    c) Type template
    d) None of the above
    Answer: b) Class template

25. A class template is instantiated:
    a) During compilation.
    b) During linking.
    c) At runtime.
    d) None of the above.
    Answer: a) During compilation.

Upcasting and Downcasting

26. Upcasting is always:
    a) Safe without explicit casting.
    b) Unsafe and requires explicit casting.
    c) Dependent on the type of inheritance.
    d) None of the above.
    Answer: a) Safe without explicit casting.

27. Downcasting in C++:
    a) Converts a base class pointer to a derived class pointer.
    b) Always requires explicit casting.
    c) May throw an exception in case of invalid cast.
    d) All of the above.
    Answer: d) All of the above.