C++ TEST-12 (ABSTRACTION)

Total points 50/50

STUDENT NAME *

VIVA

✓ 1. **What is abstraction in C++?** * 1/1

- (●) A) Showing essential details and hiding the background details ✓
- ( ) B) Combining data and functions
- ( ) C) Inheriting from another class
- ( ) D) Overloading operators

✓ 2. **Which of the following best describes abstraction?** * 1/1

- ( ) A) Data protection
- ( ) B) Data representation
- ( ) C) Data hiding
- (●) D) Simplifying complex systems ✓

✓ 3. **Abstraction focuses on:** * 1/1

- A) Implementation
- B) Functionality ✓
- C) Memory management
- D) Syntax

✓ 4. **Which OOP concept hides unnecessary details from the user?** * 1/1

- A) Inheritance
- B) Polymorphism
- C) Abstraction ✓
- D) Encapsulation

✓ 5. **Which of the following helps to achieve abstraction in C++?** * 1/1

- A) Classes
- B) Header files
- C) Abstract classes and functions ✓
- D) Constructors

✓ 6. **Which type of class cannot be instantiated in C++?** * 1/1

- A) Concrete class
- B) Abstract class ✓
- C) Derived class
- D) Static class

✓ 7. **A class that contains at least one pure virtual function is called:** * 1/1

- A) Static class
- B) Abstract class ✓
- C) Virtual class
- D) Derived class

✓ 8. **How is a pure virtual function declared in C++?** * 1/1

- A) virtual void func() = 0; ✓
- B) pure virtual func();
- C) virtual func() = pure;
- D) void func();

✓ 9. **Can objects of an abstract class be created?** * 1/1

- A) Yes
- B) No ✓

✓ 10. **Which of the following can be used to achieve abstraction?** * 1/1

- A) Abstract class
- B) Interface (pure virtual functions)
- C) Both A and B ✓
- D) None

✓ 11. **Abstraction is mainly used to:** * 1/1

- A) Hide implementation details ✓
- B) Optimize performance
- C) Access private data
- D) Increase code size

✓ 12. **Which function hides internal logic and exposes only required features?** *1/1

- A) Public function
- B) Private function
- C) Abstract function ✓
- D) Inline function

✓ 13. **Which access specifier is often used to hide implementation in abstraction?** *1/1

- ○ A) public
- ◉ B) private ✓
- ○ C) protected
- ○ D) static

✓ 14. **Abstraction is implemented using:** * 1/1

- ◉ A) Classes and objects ✓
- ○ B) Loops and functions
- ○ C) Arrays and pointers
- ○ D) Files and streams

✓ 15. **Which keyword is used to make a function virtual in C++?** * 1/1

- ○ A) abstract
- ◉ B) virtual ✓
- ○ C) override
- ○ D) base

✓ 16. **Abstraction allows focusing on:** * 1/1

- (●) A) What an object does ✓
- ( ) B) How it does
- ( ) C) Both A and B
- ( ) D) None

✓ 17. **Abstraction helps in:** * 1/1

- ( ) A) Reducing complexity
- ( ) B) Improving readability
- ( ) C) Making maintenance easier
- (●) D) All of the above ✓

✓ 18. **Which of the following best differentiates abstraction and encapsulation?** *1/1

- (●) A) Abstraction hides complexity, encapsulation hides data ✓
- ( ) B) Encapsulation hides complexity, abstraction hides data
- ( ) C) Both are same
- ( ) D) None

✓ 19. **Which function provides an abstract interface in a class?** * 1/1

- ○ A) Inline function
- ◉ B) Pure virtual function ✓
- ○ C) Constructor
- ○ D) Static function

✓ 20. **Abstract classes can contain:** * 1/1

- ○ A) Only pure virtual functions
- ○ B) Only data members
- ◉ C) Both normal and pure virtual functions ✓
- ○ D) Only constructors

✓ 21. **Which class can inherit from an abstract class?** * 1/1

- ◉ A) Any derived class ✓
- ○ B) Only virtual classes
- ○ C) Only friend classes
- ○ D) None

✓ 22. **Can an abstract class have a constructor in C++?** * 1/1

- ◉ A) Yes ✓
- ○ B) No

✓ 23. **Which keyword is used for abstract functions in C++?** * 1/1

- ○ A) abstract
- ◉ B) virtual ✓
- ○ C) pure
- ○ D) base

✓ 24. **What happens if a class inherits from an abstract class but doesn't override all pure virtual functions?** *1/1

- ◉ A) It remains abstract ✓
- ○ B) It becomes concrete
- ○ C) Compiler error
- ○ D) None

✓ 25. **Which of the following can instantiate an abstract class?** * 1/1

- ○ A) Reference
- ◉ B) Pointer ✓
- ○ C) Object
- ○ D) None

✓ 26. **Abstract classes are mainly used for:** * 1/1

- A) Code reusability
- B) Defining interfaces ✓
- C) Hiding data
- D) File handling

✓ 27. **Which of these cannot be part of an abstract class?** * 1/1

- A) Constructor
- B) Destructor
- C) Object ✓
- D) Virtual function

✓ 28. **The pure virtual function forces:** * 1/1

- A) Inheritance
- B) Implementation in derived class
- C) Both A and B ✓
- D) None

✓ 29. **Abstraction is mainly achieved at which level?** * 1/1

- (●) A) Design level ✓
- ( ) B) Implementation level
- ( ) C) Compilation level
- ( ) D) Runtime level

✓ 30. **In C++, pure virtual functions are used for:** * 1/1

- ( ) A) Abstraction
- ( ) B) Encapsulation
- ( ) C) Polymorphism
- (●) D) Both A and C ✓

✓ 31. **Which of the following can not be abstracted?** * 1/1

- ( ) A) Data
- ( ) B) Implementation
- ( ) C) Interfaces
- (●) D) Memory allocation ✓

✓ 32. **Abstraction hides the _____ details.** * 1/1

- (●) A) Implementation ✓
- ( ) B) Functional
- ( ) C) Logical
- ( ) D) Interface

✓ 33. **Which class provides only definition of functions without implementation?** *1/1

- (●) A) Abstract class ✓
- ( ) B) Concrete class
- ( ) C) Base class
- ( ) D) Static class

✓ 34. **Which concept supports partial abstraction?** * 1/1

- (●) A) Abstract class ✓
- ( ) B) Interface
- ( ) C) Multiple inheritance
- ( ) D) Template

✓ 35. **What provides 100% abstraction in C++?** * 1/1

- (●) A) Interface (all pure virtual functions) ✓
- ( ) B) Abstract class
- ( ) C) Constructor
- ( ) D) Static class

✓ 36. **Can abstract classes contain variables?** * 1/1

- (●) A) Yes ✓
- ( ) B) No

✓ 37. **Which of these can provide abstraction in C++ other than classes?** * 1/1

- ( ) A) Header files
- ( ) B) Namespaces
- (●) C) Both A and B ✓
- ( ) D) None

✓ 38. **Abstract classes can be inherited using which access specifier?** * 1/1

- ( ) A) public
- ( ) B) private
- ( ) C) protected
- (●) D) All of the above ✓

✓ 39. **Which of the following is *not* a benefit of abstraction?** * 1/1

- A) Reduces complexity
- B) Increases security
- C) Reduces code reuse ✓
- D) Simplifies maintenance

✓ 40. **Which concept allows working with abstract classes through derived objects?** *1/1

- A) Polymorphism ✓
- B) Inheritance
- C) Encapsulation
- D) Abstraction

✓ 41. **Which function is used to make an abstract interface in C++?** * 1/1

- A) virtual function
- B) pure virtual function ✓
- C) inline function
- D) friend function

✓ 42. **What is the output of creating an object of a class with a pure virtual function?** *1/1

- A) Program runs normally
- B) Compile-time error ✓
- C) Run-time error
- D) None

✓ 43. **What happens when a derived class overrides all pure virtual functions?** *1/1

- A) It becomes a concrete class ✓
- B) It remains abstract
- C) Causes an error
- D) None

✓ 44. **Abstract classes can have:** * 1/1

- A) Static members
- B) Constructors and destructors
- C) Normal member functions
- D) All of the above ✓

✓ 45. **Which of the following can't be virtual?** * 1/1

- (●) A) Constructor ✓
- ( ) B) Destructor
- ( ) C) Member function
- ( ) D) None

✓ 46. **What is the purpose of = 0 in a virtual function?** * 1/1

- (●) A) Declares it as pure virtual ✓
- ( ) B) Initializes it to zero
- ( ) C) Ends the function
- ( ) D) None

✓ 47. **Which of the following C++ features directly supports abstraction?** * 1/1

- ( ) A) Classes and objects
- ( ) B) Inheritance
- ( ) C) Virtual functions
- (●) D) All of the above ✓

✓ 48. **Which keyword is required to achieve runtime abstraction?** * 1/1

- A) static
- B) virtual ✓
- C) inline
- D) const

✓ 49. **What is another term for “abstraction through interfaces”?** * 1/1

- A) Pure abstraction
- B) Complete abstraction
- C) Interface abstraction
- D) All of the above ✓

✓ 50. **Abstraction improves software design by:** * 1/1

- A) Separating implementation from interface ✓
- B) Mixing data and functions
- C) Using global variables
- D) Avoiding inheritance