C++ TEST-20 (FRIEND FUNCTION)

Total points 50/50

STUDENT NAME *

VIVA

✓ 1. **What is a friend function in C++?** * 1/1

- A) A member function of a class
- B) A function that can access private and protected members of a class ✓
- C) A constructor
- D) A virtual function

✓ 2. **Which keyword is used to declare a friend function?** * 1/1

- A) protected
- B) friend ✓
- C) public
- D) extern

✓ 3. **A friend function is declared:** * 1/1

- A) Inside the class
- B) Outside the class only
- (✓) C) Inside the class but defined outside
- D) Inside main()

✓ 4. **Can a friend function access private members of a class?** * 1/1

- (✓) A) Yes
- B) No

✓ 5. **A friend function is:** * 1/1

- A) A member of the class
- (✓) B) Not a member of the class

✓ 6. **Can a friend function be called using the object of the class?** * 1/1

- A) Yes
- (✓) B) No

✓ 7. **How many classes can have the same friend function?** * 1/1

- ○ A) Only one
- ○ B) Two
- ◉ C) As many as needed ✓
- ○ D) Only derived classes

✓ 8. **Which of the following statements is true about friend functions?** * 1/1

- ○ A) They are in the public section only
- ◉ B) They can be declared in any section (public, private, protected) ✓

✓ 9. **Can a friend function be private?** * 1/1

- ◉ A) Yes ✓
- ○ B) No

✓ 10. **A friend function is invoked using:** * 1/1

- ○ A) Object name
- ○ B) Class name
- ◉ C) Function name directly ✓

✓ 11. **Which of the following can be a friend function?** * 1/1

- ○ A) Normal function
- ○ B) Member function of another class
- ◉ C) Both A and B ✓

✓ 12. **Can a constructor be a friend function?** * 1/1

- ○ A) Yes
- ◉ B) No ✓

✓ 13. **Can a destructor be a friend function?** * 1/1

- ○ A) Yes
- ◉ B) No ✓

✓ 14. **Can we have a friend class in C++?** * 1/1

- ◉ A) Yes ✓
- ○ B) No

✓ 15. **Friend functions are used to:** * 1/1

- ○ A) Protect data
- ◉ B) Share access between classes ✓
- ○ C) Restrict inheritance

✓ 16. **Friend functions violate which OOP principle?** * 1/1

- ○ A) Abstraction
- ◉ B) Encapsulation ✓
- ○ C) Inheritance

✓ 17. **What type of arguments does a friend function use to access class members?** *1/1

- ○ A) Pass by reference
- ○ B) Pass by pointer or object
- ◉ C) Both A and B ✓

✓ 18. **Can a friend function be overloaded?** * 1/1

- ◉ A) Yes ✓
- ○ B) No

✓ 19. **Can a friend function be virtual?** * 1/1

- ○ A) Yes
- ◉ B) No ✓

✓ 20. **Can a friend function be inline?** * 1/1

- (●) A) Yes ✓
- ( ) B) No

✓ 21. **What access specifier can a friend function be declared under?** * 1/1

- ( ) A) Only public
- ( ) B) Only private
- (●) C) Any section ✓

✓ 22. **Can a friend function be a member of another class?** * 1/1

- (●) A) Yes ✓
- ( ) B) No

✓ 23. **How many friend functions can a class have?** * 1/1

- ( ) A) Only one
- ( ) B) Two
- (●) C) Unlimited ✓

✓ 24. **Which is true about friend functions?** * 1/1

- ( ) A) They are inherited
- (●) B) They are not inherited ✓

✓ 25. **Which is false about friend functions?** * 1/1

- A) They can access private data
- B) They can be called with class name
- C) They are called using objects ✓

✓ 26. **Can a friend function access static members of a class?** * 1/1

- A) Yes ✓
- B) No

✓ 27. **Can a friend function access non-static members of a class?** * 1/1

- A) Yes ✓
- B) No

✓ 28. **What is the correct syntax to declare a friend function?** * 1/1

- A) friend void fun(); ✓
- B) void friend fun();
- C) friend: void fun();
- D) private friend fun();

✓ 29. **What happens if we don't define a friend function declared in class?** * 1/1

- A) Compiler error
- B) Linker error ✓

✓ 30. **Can a class have both friend functions and friend classes?** * 1/1

- A) Yes ✓
- B) No

✓ 31. **Friend function helps mainly in:** * 1/1

- A) Polymorphism
- B) Data sharing between unrelated classes ✓
- C) Abstraction

✓ 32. **Can a friend function be a template?** * 1/1

- A) Yes ✓
- B) No

✓ 33. **When are friend functions most commonly used?** * 1/1

- A) Operator overloading ✓
- B) Inheritance
- C) Abstract classes

✓ 34. **Can a friend function of one class be a member of another?** * 1/1

- (●) A) Yes ✓
- ( ) B) No

✓ 35. **Friend functions are declared in the class definition but:** * 1/1

- (●) A) Defined outside ✓
- ( ) B) Defined inside

✓ 36. **The friend function definition must include which argument?** * 1/1

- ( ) A) Pointer to class
- (●) B) Object of class ✓

✓ 37. **The keyword friend should appear:** * 1/1

- (●) A) In the declaration only ✓
- ( ) B) In the definition only
- ( ) C) Both declaration and definition

✓ 38. **Can two classes be declared as friends to each other?** * 1/1

- (●) A) Yes ✓
- ( ) B) No

✓ 39. **Friend function is useful when:** * 1/1

- A) Classes are related through inheritance
- B) Classes are unrelated but need to share data ✓

✓ 40. **Can a friend function access protected members?** * 1/1

- A) Yes ✓
- B) No

✓ 41. **Can a friend function be recursive?** * 1/1

- A) Yes ✓
- B) No

✓ 42. **Which of these cannot be declared as friend?** * 1/1

- A) Function
- B) Class
- C) Object ✓

✓ 43. **Which statement is false about friend functions?** * 1/1

- A) They increase coupling
- B) They support encapsulation ✓
- C) They can access private data

✓ 44. **Can a friend function be used to overload operators?** * 1/1

- (●) A) Yes ✓
- ( ) B) No

✓ **45. A friend function declaration is placed:** * 1/1

- (●) A) In the class that grants friendship ✓
- ( ) B) In both classes

✓ **46. What happens if a friend function tries to access a non-existent member?** *1/1

- (●) A) Compiler error ✓
- ( ) B) Runtime error

✓ **47. A friend function can be part of:** * 1/1

- ( ) A) Namespace
- ( ) B) Another class
- ( ) C) Global scope
- (●) D) All of the above ✓

✓ **48. Can a friend function be declared in multiple classes?** * 1/1

- (●) A) Yes ✓
- ( ) B) No

✓ **49. Friendship in C++ is:** * 1/1

- (●) A) One-way ✓
- ( ) B) Two-way automatically

✓ **50. Which of the following best describes a friend function?** * 1/1

- (●) A) It is a non-member function with special access privileges ✓
- ( ) B) It is a private function
- ( ) C) It is inherited by derived classes