INTERVIEW QUESTIONS FOR OBJECT ORIENTED PROGRAMMING

Assignment 1

Submission date: 31/10/2021

1. Explain OOPS?
2. Explain an abstraction? Real life example.
3. Explain encapsulation? Real life example.
4. Explain the relationship among abstraction and encapsulation?
5. Explain polymorphism?
6. Explain Inheritance?
7. How composition is better than inheritance?
8. Which OOPS concept is used as a reuse mechanism?
9. Which OOPS concept exposes only the necessary information to the calling functions?
10. Explain a class? Create a class.
11. Using above created class, Write in brief abstraction and encapsulation
12. Explain difference among class and object?
13. Define access modifiers?
14. Explain an object? Create an object of above class.
15. Give real life examples of object.
16. Explain a Constructor.
17. Define the various types of constructors?
18. Whether static method can use nonstatic members?
19. Explain Destructor?
20. Explain an Inline function?
21. Explain a virtual function?
22. Explain a friend function?
23. Explain function overloading?
24. Explain a base class, sub class, super class?
25. Write in brief linking of base class, sub class and base object, sub object.
26. . Explain an abstract class?
27. Explain operator overloading?
28. Define different types of arguments? (Call by value/Call by reference)
29. Explain the super keyword?

30. Explain method overriding?

31. Difference among overloading and overriding?

32. Whether static method can use non-static members?

33. Explain a base class, sub class, super class?

34. Write in brief linking of base class, sub class and base object, sub object.

35. Explain an interface?

36. Explain exception handling?

37. Explain the difference among structure and a class?

38. Explain the default access modifier in a class?

39. Explain a pure virtual function?

40. Explain dynamic or run time polymorphism?

41. Do we require a parameter for constructors?

42. Explain static and dynamic binding?

43. How many instances can be created for an abstract class?

44. Explain the default access specifiers in a class definition?

45. Which OOPS concept is used as reuse mechanism?

46. Define the Benefits of Object Oriented Programming?

47. Explain method overloading?

48. Explain the difference among early binding and late binding?

49. Explain early binding? Give examples?

50. Explain loose coupling and tight coupling?

OBJECT ORIENTED PROGRAMMING
Assignment 2
Submission date: 31/10/2021

51. Give an example among tight coupling and loose coupling.

52. Write in brief abstract class.

53. Define the Benefits of oops over pop?

54. Explain Generalization and Specialization?

55. Write in brief Association, Aggregation and Composition?

56. Write in brief Object Composition vs. Inheritance.

57. Explain cohesion?

58. Explain "black-box-reuse" and "white-box-reuse"?

59. Explain "this"

60. Write in brief static member and member function.

61. How will you relate unrelated classes or how will you achieve polymorphism without using base class?

62. Explain Diamond problem?

63. Explain the solution for diamond problem?

64. Explain the need of abstract class?

65. Why can't we instantiate abstract class?

66. Can abstract class have constructors?

67. How many instances can be created for an abstract class?

68. Which keyword can be used for overloading?

69. Explain the default access specifiers in a class definition?

70. Define all the operators that cannot be overloaded?

71. Explain the difference among structure and a class?

72. Explain the default access modifier in a class?

73. Can you list out the different types of constructors?

74. Explain a friend function?

75. Explain a ternary operator?

76. Do We Require Parameter For Constructors?

77. Explain Sealed Modifiers?

78. Explain The Difference Among New And Override?

79. How Can We Call The Base Method Without Creating An Instance?

80. Define The Various Types Of Constructors?

81. Define Manipulators?

82. Can you give some examples of tokens?

83. Explain structured programming and its disadvantage?

84. Explain the advantage of C++ being a block-structured language?

85. Can Struct be inherited?

86. When to use interface over abstract class.

87. Explain a private constructor? Where will you use it?

88. Can you override private virtual methods?

89. Can you allow class to be inherited, but prevent from being over-ridden?

90. Why can't you specify accessibility modifiers for methods inside interface?

91. Can static members use non static members? Give reasons.

92. Define different ways a method can be overloaded?

93. Can we have an abstract class without having any abstract method?

94. Explain the default access modifier of a class?

95. Can function overriding be explained in same class?

96. Does function overloading depends on Return Type?

97. Define different ways to declare an array?

98. Can abstract class have a constructor?

99. Define rules of Function overloading and function overriding?

100.   Explain the concept of Pure Virtual Functions?