Object Oriented Programming in C++
Question Bank

<u>PART-A Questions</u>

1. Distinguish & and && operators.

2. How does an enumerated statement differ from a *typedef* statement?

3. What are the various members of a class?

4. Who can access the protected members of a class?

5. How many constructors can a class have?

6. What is an intermediate base class?

7. What are the conditions are satisfied for casting operator function?

8. What is an abstract class?

9. Can you overload template function?

10. What are the two different types of exceptions?

11. The size of the data type double is _____

12. _____ mechanism associates the code and the data into single unit.

13. _____ and _____ does not have return type.

14. _____ and _____ operators are used to allocate memory.

15. Operators that cannot be overloaded are_____

16. List any 2 string handling functions.

17. fout.seekg(o,ios::beg) is used to do _____

18. Mention any 2 ios member functions to get formatted output data.

19. State the purpose of the preprocessor directive #include<iostream.h>

20. The _____ functions are invoked, when an exception is raised and the handle is not found

21. Define object.

22. What are manipulators?

23. _____variable are those variables, which retain its value between function calls.

24. Give an example for inline function.

25. The name of a constructor must be _____.

26. State the use of strlen( ) function.

27. What is a new operator?

28. What is Base class?

29. Define virtual function.

30. Mention the use of templates

31. When the operator able to perform three operands, it is called as _____

32. The insertion operator (<<) and extraction operator (>>) are member operator of the_____,_____ object.

33. _____ are used to define user access to the data member and member function of the class.

34. _____, _____ are member access operators.

35. _____,_____are free store operators and also referred as memory management operators

36. Operators that cannot be overloaded are_____

37. _____ class is one that is not used to create object.

38. The compiler does not support automatic type conversation for the _____

39. The ios class contains the member functions ____,____, ____, _____, ____.

40. Error handling functions are _____, _____, _____, _____.

41. State the difference between a structure and class in C++?

42. _____ and _____ operators are used for data assignment and comparison in C++.

43. Is main( ) a user defined function ?

44. Member access operators are _____ and _____.

45. Constructors are used for _____ data members in a class.

46. Can you overload all operators in C++?

47. When do you refer a class as pure virtual function?

48. Which class cannot create object?

49. The default action for terminate is to invoke _____ function in exception handling.

50. C++ supports _____ and _____exceptions.

51. Being able to create your own types is called _____.

52. Once a class has been written, created and debugged, it can be distributed to other programmers for use in their own programs. This is called _____.

53. Write a *for* loop that displays the number from 100 to 110.

54. The first line of a function definition is referred to as the _____.

55. The only technical difference between structures and classes in C++ is that _____.

56. An array name, used in the source file represents the _____ of the array.

57. An overloaded operator always requires one argument less than its number of operands. (True/False).

58. Write a statement that displays the address of the variable *testvar*.

59. A static function,

    a. should be called when an object is destroyed.
    b. is closely connected to an individual object of a class.
    c. can be called using the class name and function name.
    d. is used when a dummy object must be created.

60. In console mode programs, the printer can be accessed using the predefined filename _____.

61. Write the major parts of a C++ program.

62. What is the purpose of <iostream.h>?

63. When do we use default arguments in a function?

64. Can inline function be recursive?

65. How do we invoke a constructor function?

66. What are objects?

67. When is friend function compulsory in overloading?

68. Which class is not used to create objects?

69. Are virtual functions used to create pointers to base classes?

70. What is the file mode for using binary files?

71. What does Object Oriented Programming (OOP) do that traditional languages like C, Pascal, COBOL and BASIC don't?

72. State the problems or weaknesses in the procedural/structural programming.

73. State the difference between the break and continue statements.

74. Write a statement that sets the *hrs* member of the *time2* structure variable equal to 11.

75.   What is the purpose of class declaration?

76.   What is the use of static class data?

77.   State two advantages of the operator overloading in C++.

78.   Define inheritance.

79.   Why are virtual functions needed?

80.   State the advantages of Streams

81.   Define: Object

82.   Write the relational operators in C++.

83.   What is enumeration?

84.   Define: inline functions

85.   What is constructor?

86.   What is an array?

87.   Define copy constructor.

88.   State the benefits of inheritance.

89.   What is exception?

90.   Define: Function Templates.

91.   What is encapsulation?

92.   What are escape sequences?

93.   Differentiate between while loop and do loop statements.

94.   How are enumerations declared in C++?

95.   What is static data member?

96.   How are the objects used as function argument?

97.   When is the deferencing operator ->* used?

98.   What is function overloading? Give an example.

99.   List out the operators that cannot be overloaded using Friend function.

100.   What are seekg( ) and seekp( ) functions?

101.   What are the basic concepts of OOS?

102.   What are objects?

103. What is function prototype?

104. What is a enumeration?

105. Define copy constructor.

106. What is a class?

107. How can an overloaded operator be invoked using member functions?

108. What is multiple inheritance?

109. What are stream errors?

110. Define pointers to member

111. What are objects?

112. What are manipulators?

113. What is a default argument?

114. What is an inline function?

115. How to access a class member?

116. When is the deferencing operator ->* used?

117. Define copy constructor.

118. List out the operators that cannot be overloaded.

119. What is meant by Abstract base class?

120. What are virtual functions?

121. What is Data Abstraction?

122. Give an example in C++ for polymorphism.

123. Delete is a _____ operator.

124. List some of the storage class specifiers.

125. What is a destructor?

126. Manipulators are operators which are used to _____ the data.

127. State the use of strcat() function.

128. Does C++ support multiple inheritance?

129. What is a copy constructor?

130. _____ pointer refers to an object that currently invokes a member function

131.   Which concept of OOP uses reusability?

132.   What do you mean by an Escape sequence?

133.   Mention the use of the break statement.

134.   What is meant by a default argument?

135.   _____ data or functions can only be accessed from within the class.

136.   const int SIZE=10; - Comment on the usage of this statement with respect to arrays.

137.   What are Abstract classes?

138.   Mention the functions of the new operator.

139.   _____ points to the object itself.

140.   Mention the functions which are used to set and examine the get pointer.

141.   In C++, function contained within a class is called _____.

142.   What header file must you #include with your program to use setw and endl?

143.   A relational operator is used to _____.

144.   The increment expression in *for* loop can decrement the loop variable. (True / False)

145.   What is the purpose of class specification declaration?

146.   A member function can always access the data _____.

147.   When you overload an arithmetic assignment operator it will _____.

148.   How many arguments are required in the definition of an overloaded unary operator?

149.   Name three-stream class commonly used for disk I/O.

150.   What is a file?

151.   What is the use of *this* pointer?

152.   What does the default assignment operator (=) do when applied to objects?

153.   Define nameless temporary object.

154.   A pointer to void can hold pointer to _____.

155.   What is the use of static class data?

156.   Write a statement that defines an array called manyBird that holds 50 objects of type bird.

157.   The expression 11 % (-3) evaluates to _____.

158. How does the statement #include <myheader.h> differs from #include "myheader.h"?

159. What is Object oriented programming?

160. What is use of constant member function?

PART B Questions

1. What do you understand about access specifier?

2. Which member functions are created automatically if they are not included in the class definition?

3. State the difference between a constructor and a destructor.

4. Describe the visibility of inherited members

5. What are the different ways for getting formatted output data?

6. Write down the various logical operators used in C++.

7. What is the purpose of access specifier?

8. How will you overload binary operators using friend function?

9. Draw the block diagram of the stream classes for input/output operations.

10. What is a template? Give an example.

11. What are Built in functions? Give any two examples.

12. State the use of Switch statement.

13. What is meant by array? Give an example.

14. List the operators which can not be overloaded in C++.

15. Write any four classes of file stream

16. List out the benefits of Object-oriented programming.

17. In which situation inline expansion may not work?

18. What is operator overloading? Write the syntax.

19. What is the various arithmetic operations can perform by C++ pointers?

20. Define function template.

21. How do you use the relational operators in C++?

22. Create an integer array with size 5 and get the input and display the same.

23. Distinguish the constructor and destructor.

24. Write notes on exception handling.

25. Define the visibility of inherited members

26. Are const better than # define? Justify.

27. What is meant by an inline function? When should a function be made inline?

28. Distinguish between private destructors and public destructors.

29. What is the characteristic feature of private inheritance? How does it differ from that of public inheritance?

30. Distinguish the term template class and class template.

31. How are data types in C++ classified?

32. Differentiate between call by value and call by reference.

33. Mention the operators that cannot be overloaded. Mention the operators that cannot be overloaded using friend function.

34. Illustrate the use of pointer to object.

35. What should be placed inside a try block?

36. Define whitespace. Give example.

37. What is the use of functions in C++?

38. Create a class that imitates part of the functionality of the basic data type int. Call the class Int (Note different spelling). The only data in this class in an int variable. Include member functions to initialize an Int to 0, to initialize it to an int value, to display it (it looks just like an int) and to add two Int values. Write a program that exercises this class by creating two initialized and one uninitialized Int values, adding these two initialized values and placing the response in the uninitialized value, and then displaying this result.

39. What do you mean by public and private inheritance?

40. What is the template? Give different ways with an example.

41. Give a short note on Polymorphism.

42. Explain about switch case statement in C++.

43. Write a brief description about constructor overloading.

44. What is a pointer variable?

45. Concisely describe about class templates.

46. What is the difference between class and structure?

47. What is static member function?

48. What is the ambiguity between default constructor and default argument constructor?

49. What does inheritance means in C++? What are the different forms of inheritance?

50. What are pure virtual functions? Write the syntax.

51. What is a scope resolution operator?

52. What is a default argument?

53. How to create an object and a class?

54. Explain overloading of unary and binary operator.

55. What are Friend functions? Write the syntax. Write some properties of friend.

56. What are free store operators (or) Memory management operators?

57. How are the member functions defined?

58. Define parameterized constructor.

59. Explain one class to another class conversion with an example.

60. What is static member function? Give example.

61. List out the various type conversions with an example.

62. Write about Message Passing.

63. When is inline function? Give an example.

64. What do you mean by Pure Virtual Function?

65. Mention the usage of seekg and tellg functions.

66. When should comments be used in C++?

67. What are inline functions? Give the syntax of an inline function.

68. What is a string constant? Give an example.

69. Write a small note on *protected* access specifier.

70. What is a Pure virtual function? Explain.

71. What is the difference between a character and a character string representation?

72.   Write the use of default arguments.

73.   What is copy constructor?

74.   Why is it necessary to overload an operator?

75.   Define template and its role.

76.   Write a simple C++ program to demonstrate the concept of "method overloading".

77.   Define an inline function called 'inchToCenti( )' that accepts inch as input parameter and converts it and return the value in centimeter. Write a simple C++ code segment that accepts the inch value as input from the user and displays the result.

78.   Can a function return an 'object' to the calling program? Justify your answer.

79.   Define operator overloading. What are the operators that cannot be overloaded in C++?

80.   Predict the output for the given below program. Justify your answer.

```
main( ) {
        if (-100)
                cout << "Have a nice day";
        else
                cout << "Good day";
}
```

Part C Questions

1.   Illustrate the basic concept and describe the Characteristics of OOP with an example.

2.   With suitable example discuss the various operators supported by C++.

3.   Write advantages of Inline function? Explain the concept of inline function with example. Describe the restriction of inline function.

4.   a.What do you understand about default arguments? What is the essentiality of the default arguments?

        b.Write a program to swap two variables using call by value and call by reference.

5.How do you handle String in C++? Explain.

6.   a.Write the special characteristics of constructor.

        b.What is dynamic constructor? Explain with an example.

7. Write a c++ program to perform the addition and subtraction of complex numbers using operator overloading.

8. Write a c++ program to perform a student information system using multilevel inheritance. The system should consists student personal details, academic record and extra curriculum.

9. Explain the concept of virtual function with suitable example.

10. How do you handle exception in C++? Explain.

11. How are the object oriented programming concepts superior than procedure oriented programming? Explain with an example.

12. What is the need of data type in programming language? Describe the standard data types used in C++ and type conversion concepts.

13. Illustrate the various methods of passing values with an example.

14. Write a c++ program to prepare a calendar for the given year.

15. Explain the different types of constructors with a complete example.

16. Write an interactive c++ program to perform the Library system using constructors, which covers issue, return, renewal, book details and status of the book.

17. a.Write the standard rules for overloading operators.

    b.Write a c++ program to overload the unary operator '- - '.

18. Discuss multiple inheritance with an example.

19. Illustrate the various stream classes that are used to handle the input/output and file in C++.

20. Write a program to sort the integers and strings using function templates.

21. What is object-oriented programming? How is it different from the procedure – oriented programming? Discuss.

22. Write a C++ program to evaluate the following function.

$$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + ...$$

23. Explain the different control constructs in C++.

24. Write a C++ program to find the volume of cube, cylinder and rectangular box using function overloading.

25. Explain how do you use objects as function arguments with suitable examples.

26. Discuss the role of constructors and destructors in a class. State the rules associated with them.

27. Explain operator overloading with two different examples.

28. Explain single inheritance and multiple inheritance with example.

29. Explain the features of istream and ostream classes.

30. Write notes on error handling in C++.

31. Write the features of object oriented programming and distinguish from procedure oriented programming.

32. What is a data type? Describe the standard data types used in C++. How do you obtain type conversation during the execution?

33. With example explain how do you pass the arguments in function.

34. a.Write a c++ program to read an integer number and display the same in words.
    b.Write notes about storage class and default arguments.

35. What are constructor and destructor? Classify the different types of constructor with example.

36. Write an interactive c++ program to perform the banking system. Which Includes deposit, withdraw, interest, balance query, and loan process. The account number and initial amount is initialized using constructor.

37. a.Write the standard rules for overloading operators.

    b.Illustrate overloading binary operators using friends with an example.
38. Discuss in detail about inheritance.

39. How do you handle your unformatted and formatted input and output data in c++.

40. What is template? With suitable example discuss class, function template.

41. With suitable example discuss the characteristics of OOP.

42. What is inline function? Write in which situation the inline function may not work. Write a C++ program to find the factorial of a given number using inline.

43. a.   Describe about storage class variables.

    b.   Create a class 'Student' which consists of name, roll no., sex and subject marks. Find the total marks and average of each student. Achieve this using Arrays.

44. a.What is the purpose of default arguments?
    b.Write a program to sort the set of strings using call by value and call by reference.
45. Explain the various string handling functions in C++ with suitable examples.

46. Discuss the various types of inheritance with an example.

47.   a.What do you understand about operator overloading? Explain.

b.Write a C++ program to perform the multiplication of complex numbers using operator overloading.

48.   How do you handle the errors in file operation?

49.   What is the purpose of virtual function? Explain.

50.   Write a C++ program to manage the student information using friend function and file concepts.   File 1   contains the personal data and file 2 contains academic data of the students.

51.   a.How can you specify a class?

b. Describe the different mechanisms of accessing data members and member functions in a class with a suitable example.

52.   Explain the basic concepts of object oriented programming.

53.   a.What are the various storage classes and give their scopes with illustrations.
       b.   What is inline function? What is its use? Discuss its limitations.

54.   Explain the following:
       a.   Function overloading

b.   Operator Overloading

c.   Friend and virtual function

55. What is operator overloading? Develop a C++ program to overload unary operator for processing the objects of a class called counter.

56. a. Write a C++ program to overload the increment operator with prefix and postfix forms.

b. Explain how the destructor is invoked?

57. a. What is inheritance? What are different types of inheritance? Discuss with examples

b. Show the memory allocation for static and non-static data and functions of an object.

58. a. Explain about overriding methods with an example.

b. Give the different levels access protection available in C++.

c. Write a C++ program to find if the given string is a Palindrome or not.

59. Write a C++ program to create a file namely "student" with two field name and age. Get any five student information and write it in file and close it. Again display the information in the console from the file.

60. a. Write a C++ program using function template for finding the maximum value in an array

    b. Describe the exception handling mechanism in C++ with an example.

61. Describe the various concepts of object oriented programming.

62. Describe with example the uses of enumeration data types.

63. Explain selection structure and also write a C++ program to print the following output

    <div align="center">
    1
    22
    333
    ….
    </div>

64. Write a function power() to raise a number m to a power n. The function takes a double value for m and int value for n and returns the result correctly. Use a default value of 2 for n to make the function to calculate squares when this argument is omitted. Write a function that gets the int values of m. Write a main to get m and n from the user.

65. How can you define member functions? Explain with example.

66. What are constructors? How can you overload them? Write a C++ program to explain copy constructor.

67. Write a C++ program to do complex number manipulation using operator overloading.

68. Briefly explain multilevel inheritance with example? Explain pointers to pointer.

69. a. What are the merits and demerits of friend function?

    b. Write a static function to calculate variance and standard deviation of N numbers.

70. a. Describe the various approaches by which we can detect the end-of-file condition successfully.

    b. What are the ways we can handle errors during file operations?

71. Briefly explain the major characteristics of Object Oriented Programming.

72. Explain in detail about the following:

    a. Preprocessor directives
    b. Header files
    c. Comments

73. Explain about condition and looping statements in detail.

74. Explain the function classification with the passing the arguments.

75. Explain about Multiple constructors with an example and list the special characteristics of the constructors.

76. Explain the following with examples.

    a. Pointers and functions.
    b. Pointers to objects.
    c. Pointer to pointer.

77. Explain about the operator overloading.

78. Explain overloading Unary and Binary Operator.

79. Explain how exception handling is achieved in C++. Give 5 different exception handling mechanism and explain the working of them.

80. a. What are file modes? Describe various file mode options available in C++.

    b. Explain the various file stream classes needed for file manipulations in C++.

81. Elucidate about the salient features of object oriented programming.

82. Give explanation about different data types and operators in C++.

83. Explain about the control statements in C++.

84. Describe about various types of functions in C++. Write a program to calculate factorial of n number using C++.

85. Furnish the detail description about Classes, object and access specifier. Write a program for maintaining employee details using classes.

86. Provide the detailed description about structures and classes. Create a structure for student fee details which contain Register number, Name of the student, Tuition fee, auxiliary fee and access student fee details using structures.

87. Describe about inheritance.

88. Illustrate about arrays. Write a program to calculate the multiplication of 3*3 matrix using array.

89. Provide detail description about virtual functions.

90. Give detail description about files in C++. Write a program for read, write, update and display operations for the personal details using file.

91. Explain the features required for object oriented language with examples.

92. Give a brief account on the following:

    a.   Tokens
    b.   Keywords
    c.   Rules for naming the identifiers in C++.
    d.   Scope Resolution operators
    e.   Symbolic Constants

93. What is function overloading? Write a program in C++ to find the maximum of two integer and two real numbers using function overloading.

94. Explain default and constant argument. Give examples.

95. Explain different types of constructors with example code segments and the rules associated with constructors.

96. a. What are strings? Are they standard or derived data types? Write an interactive program to   check whether a given string is a palindrome or not.

    b. What are command line arguments? Explain with an example.

97.   Write a C++ program to overload the operator to add two complex numbers.

98. Explain different types of inheritance with suitable examples and discuss about the visibility modes in inheritance.

99. Can a constructor be virtual? Justify your answer. What is the use of virtual destructor? List down the basic rules for virtual functions.

100.Write a program to create a file called emp.dat with employee number, name, Basic Pay, deductions and allowances as record fields. Open a file, read the record, calculate the salary and write it back to the same file.

101.   What are the Features of OOPS and how are they implemented in C++?

102.   What are the operators available in C++?

103.   Describe about inline functions.

104.   Illustrate Function Overloading with an example program.

105.   Explain about Operator Overloading with an example program.

106.   Write in detail about Multiple constructors (constructor overloading).

107.   Explain about Type conversions.

108.   Discuss the different types of inheritance.

109.   Explain with an example, the concept of virtual functions.

110.   Write in detail the fundamentals of Exception Handling.

111.   What are the features of OOP's and how are they implemented in C++?

112.   Discuss about the types of operators with example.

113.   Explain about the control structures in C++ with example.

114.   a.   Explain about inline function.
        b.   Explain Function Overloading.

115.   a.   Discuss about the class and objects in C++ with example.
        b.   Write a program in C++ to concatenate two strings.

116.   Elucidate the types of constructors with example.

117.   Explain operator overloading with the implementation of Complex numbers and its numeric operations addition, subtraction, multiplication and division.

118.   Explain inheritance.

119.   Describe the following with example.

     a. Virtual function
     b. Static function
     c. Friend function

120.   Discuss about the templates and exception with example.

121.   a.   Compare Object Oriented Paradigm with Structured approach in detail.
        b.   List out logical and relational operators and their functions with examples.

122.   a.   Discuss the merits and demerits of Object Orientation in detail.
        b.   Write a C++ program to generate a Fibonacci series using Recursion.

123.   Explain the various control statements available in C++ with suitable examples.

124   Explain overloading of functions by illustrating overloading concepts to add time in the format hh:mm:tt.

125.   What is use of Constructor? Give general syntax of constructor and explain default and copy constructor with a sample. How to perform constructor overloading?

126.   Explain about the various String Manipulation functions in C++ with an example for each.

127.   Write a C++ program to add and multiply two distance objects with feet and inch fields by overloading the addition and multiplication operators.

128.   Write a C++ program to calculate the area of various polygons using Virtual functions.

129.   Write a program using class templates to create and manage a generic array which can store int, float or string. Write member functions for receiving and printing array values. Using main program, test the template class by creating instances with the three data types.

130.   Illustrate the usage of open, close, read and write operations performed in a file using an example program.

131.   Write a detailed note on concepts of OOP.

132.   Explain in detail about different data types and operators used in C++.

133. Write a C++ program to pass a structure by value and by reference.

134. Explain function overloading with a C++ program to perform addition.

135. Write a detailed note on constructors and its types. Explain with a C++ program.

136. a. Illustrate returning object from functions through a C++ program by adding two complex numbers.

    b. Explain static class data with an example C++ program.

137. a. Write a C++ program to overload '+' operator to perform of addition of two distances in feet and inches.

    b. Write a C++ program to overload the unary operator.

138. Explain in detail about multiple inheritance with a detailed example program.

139. Explain the concept of Virtual functions with an example C++ program.

140. Write a detailed note on Class templates with an example C++ program.

141. Briefly discuss about different data types with examples.

142. Distinguish between the following terms:

    a. Cin
    b. Cout
    c. Comments
    d. Escape sequence
    e. Type conversion

143. Explain the following statements in detail with examples:

    a. *for* statement
    b. *do while* statement
    c. *switch* statement
    d. *if else* statement

144. a. What do you mean by overloading of functions? When do we use this concept?

    b. Explain inline function with an example.

145. Write a C++ program to merge two sorted arrays consisting of integers.

146. Explain the following in detail:

    a. Static class data
    b. Const and classes
    c. Structure and class

147. Explain the syntax of binary operator overloading. How many arguments are required in the definition of an overloaded binary operator?

148.　a.　Illustrate how constructors are implemented when the classes are inherited.

　　　b.　Illustrate the implementation of both multilevel and multiple inheritance.

149.　Write a class template to represent a generic vector including member functions to perform the following tasks

　　a. To create the vector
　　b. To modify the value of a given element
　　c. To multiply by a scalar value

150.　Explain the various file stream classes needed for file manipulations.

151.　Define Object Oriented paradigm. Explain the various characteristics of OO paradigm.

152.　Illustrate with necessary example, the various conditional and looping statements used in C++.

153.　Explain in detail, the scope and storage class of local, static and global variable.

154.　Differentiate function overloading from function overriding with suitable example.

155.　Create a class Matrix with data member 10 x 10 array. A constructor should allow the programmer to specify the actual dimensions of matrix. The main( ) program that operates on matrix class might look like.

```
Matrix m1(3, 4); // Matrix m1 of size 3 x 4
int temp = 1234;
m1.putElement(2, 3, temp); // insert 'temp' at 2nd row, 3rd column
temp    = m1.getElement(3, 3); // retrieve element from 3rd row, 3rd column
```

156.　Explain the concept of exception with examples and also specify how exception specification can be performed.

157.　What is inheritance? Differentiate public inheritance from private inheritance. Write a C++ program to define a base class Father with attributes name, age. Inherit another class Son from the base class with specialized attributes name and age. Write member functions to read and print the details of members in both classes. Using a pointer to the class Father, invoke the function that prints the Father class's attributes and also print Son class's attributes

158.　Explain in detail, type conversion between objects of different classes with necessary example.

159.　Define a class Person with a set of data members and member methods of your own. Write a C++ program to write and read the instance of user-defined class Person to and from a file.

200.　Write a C++ program to create Generic Linked List class using template that stores user-defined data type and primitive data type.