

QUESTION BANK – OOP

QUESTION

1. Differentiate between procedural programming and object oriented programming language.
2. Explain features/characteristics of Object Oriented Programming.
3. Enlist applications where Object Oriented Programming can be used.
4. Why Object Oriented Programming is required? Justify your answer.
5. How C++ variables are declared?
6. What is object oriented and object based language?
7. Write program to demonstrate scope resolution operator and constant keyword.
8. Write short note on:
 - a. Function Call by value
 - b. Default argument in functions
 - c. Constants
 - d. Data types in C++
 - e. Operators in C++
 - f. Inline functions
 - g. Prototype of Function
 - h. I/O Manipulators
9. Write short note on:
 - a. Class
 - b. Object
 - c. Public, Private and Protected access modifiers
 - d. Scope resolution operator
 - e. This pointer
 - f. Class as ADT and code reuse
 - g. Characteristics of friend function
 - h. Name spaces
 - i. Types of constructor
 - j. Destructor
10. Write a program to demonstrate various constructors and destructor for time class.
11. What is meant by abstraction and encapsulation?

QUESTION BANK – OOP

12. Explain array of objects using example.
13. Write a program to demonstrate use of inline function to find factorial of a number.
14. Write a program to create use defined manipulator for currency symbol.
15. Differentiate between early binding and late binding.
16. What is inheritance? Which are various types of inheritance? Explain single inheritance with example.
17. What is the need of operator overloading?
18. What is the difference between operator overloading as member function and using friend function?
19. How unary and binary operators are overloaded using member function?
20. What is type casting? What is difference between implicit and explicit type casting?
21. Which are various type conversion methods? Explain any one with example.
22. Write short note on:
 - Virtual base class
 - Virtual function
 - Pure virtual function and abstract class
 - Virtual destructor
 - Class and object
 - Static members
 - Friend class
 - Base and derived class
 - Polymorphism
23. What do you mean by operator overloading? How unary and binary operators are overload? Enlist operators which can't be overload. Which are the various rules for operator overloading?
24. Which are various types of inheritance?
25. What do you mean by function overriding? What is the difference between function overloading and function overriding?
26. Explain friend keyword with respect to function and class .
27. Explain static keyword with respect to variable, function and class with example.

QUESTION BANK – OOP

28. Write a program to demonstrate friend function to access member variables of two different classes.
29. What is use of virtual keyword in hybrid inheritance? Explain with example.
30. Explain explicit and mutable keyword with exampld.
31. Explain inheritance and various access specifier used in inheritance. How access specifier affects inheritance of members of a class?
32. Write short note on smart pointer and shared pointer.
33. Explain pointer, array of pointer, pointer to object, pointer to derived class array of pointer to object.
34. Which operators can't be overloaded using friend function? How unary and binary operators can be overloaded using friend function?
35. Write a program to demonstrate basic to class type conversion.
36. Write a program to demonstrate class to basic type conversion.
37. Write a program to overload "+" operator for concatenation of two strings, ">>" for reverse of two strings and "<<" to capitalize a string.
38. Write a program to overload unary minus operator and binary addition operator for time class.
39. Write a program to demonstrate multiple inheritance.
40. Write a program to demonstrate hybrid inheritance.
41. Write a program to demonstrate hierarchical inheritance.
42. Write a program to demonstrate use of virtual base class.
43. Write a program to demonstrate virtual function.
44. Write a program to demonstrate pure virtual function.
45. Write a program to demonstrate friend class.
46. Write a program to demonstrate friend function.
47. What is the need of generic programming? How generic programming is implemented?
48. What do you mean by function template?
49. What do you mean by class template?
50. Write a program to demonstrate function template for swap function
51. Write a program to demonstrate class template for stack class
52. Write a function template to find minimum of two numbers.

QUESTION BANK – OOP

53. Write a class template to demonstrate generic vector class
54. Write a class template to demonstrate generic queue class.
55. Differentiate between function template, function overloading and function overriding.
56. How to overload function templates? Explain with example.
57. Describe type template parameters and non-type template parameters
58. How templates works with inheritance? Demonstrate using example.
59. Which are different types of exception? What do you mean by exception handling?
60. How exception handling works in C++? Which are the keywords available in C++ for exception handling?
61. Demonstrate exception handling using multiple catch block. How single catch block can handle all exceptions?
62. Explain throw and rethrow with example.
63. How use can define its own exception? Demonstrate using example.
64. How constructor and destructors with exception handling?
65. Write short note on standard input, output and error stream.
66. How various stream errors can be handled?
67. Which are various classes used for reading and writing a file?
68. Write a program to demonstrate reading and writing in a file.
69. Which are various file opening modes? Explain any 3 with example
70. How to read and write user defined variable or class object in C++ file?
71. What is difference between sequential access file and random access file? Which are various file manipulators in C++ file handling?
72. Write a program to overload << and >> operator in C++.
73. Write short note on command line arguments. Explain command line arguments with example.
74. What is early binding vs late binding?
75. What do you mean by Standard Template Library (STL) in C++? Which are various components of STL?
76. Enlist different containers, iterators and algorithms available in STL? Explain any two of container, algorithm and iterator.
77. What do you mean by iterator? Explain different iterators

QUESTION BANK – OOP

78. Explain stack, queue and deque STL.
79. Explain vector and list STL
80. Explain map, set STL.
81. Write a program to demonstrate stack STL
82. Write a program to demonstrate queue STL
83. Write a program to demonstrate deque STL
84. Write a program to demonstrate list STL
85. Write a program to demonstrate vector STL
86. Write a program to demonstrate set STL
87. Write a program to demonstrate map STL
88. Write a program to demonstrate quick sort and heap sort algorithm using STL.
89. Write a program to search a number in a list using binary search using STL.
90. Write a program to find minimum and maximum number in a list.
91. Which STL algorithms are used for following operations in STL
 - a. find minimum and maximum element in a list
 - b. perform binary search on a vector
 - c. sort vector using quick sort
 - d. sort list using heap sort
 - e. sort using minmax algorithm

Prof. Sonali K.