1. What is operator overloading in C++?
   1. Creating new operators in C++
   2. Using operators to perform different operations on different data types
   3. **Redefining operators to work with user-defined data types**
   4. Using operators with user-defined precedence
2. Which of the following operators cannot be overloaded in C++?
   1. +
   2. &&
   3. %
   4. **::**
3. What is the correct syntax for overloading the assignment operator in a class?
   1. **operator =**
   2. operator :=
   3. operator ()
   4. operator \*=
4. When overloading the subscript operator [], which type of function should be used?
   1. **Unary member function**
   2. Binary member function
   3. Ternary member function
   4. Non-member function
5. What is the purpose of the "friend" keyword in operator overloading?
   1. It makes the overloaded operator a member function.
   2. **It allows access to private members of the class.**
   3. It defines a new operator for a class.
   4. It creates a new instance of the class.
6. What is the precedence of the overloaded << operator?
   1. **Same as the built-in << operator**
   2. Higher than the built-in << operator
   3. Lower than the built-in << operator
   4. It depends on the specific implementation
7. Which operator is used to overload the function call operator?
   1. **()**
   2. []
   3. ->
   4. {}
8. What is a binary operator?
   1. Operator that performs its action on a single operand
   2. **Operator that performs its action on two operand**
   3. Operator that performs its action on three operand
   4. Operator that performs its action on any number of operands
9. Which is the correct example of a binary operator?
   1. ++
   2. —
   3. Dereferencing operator(\*)
   4. **+**
10. Which is the correct example of a unary operator?
    1. &
    2. ==
    3. **--**
    4. /
11. Which is called ternary operator?
    1. **?:**
    2. &&
    3. |||
    4. ===
12. Which is the correct statement about operator overloading?
    1. Only arithmetic operators can be overloaded
    2. Only non-arithmetic operators can be overloaded
    3. Precedence of operators are changed after overlaoding
    4. **Associativity and precedence of operators does not change**
13. What is Type Casting?
    1. Converting a file from one type to another
    2. Creating new arrays
    3. Deleting the variable from memory
    4. **Converting a variable from one type to another**
14. Who initiates explicit type casting?
    1. **The programmer**
    2. The compiler
    3. The microcontroller
    4. The assembler
15. How can we restrict dynamic allocation of objects of a class using new?
16. By overloading new operator
17. By making an empty private new operator.
18. **By making an empty private new and new[] operators**
19. By overloading new operator and new[] operators
20. Which of the following operators cannot be overloaded?
21. . (Member Access or Dot operator)
22. ?: (Ternary or Conditional Operator )
23. :: (Scope Resolution Operator)
24. .\* (Pointer-to-member Operator )
25. **All of the above**
26. Which of the following operators are overloaded by default by the compiler in every user defined classes even if user has not written?

1) Comparison Operator ( == )

2) Assignment Operator ( = )

1. Both 1 and 2
2. Only 1
3. **Only 2**
4. None of the two
5. Which of the following operators should be preferred to overload as a global function rather than a member method?
6. Postfix ++
7. Comparison Operator
8. **Insertion Operator <<**
9. Prefix++
10. How does C++ compiler differs between overloaded postfix and prefix operators?
11. C++ doesn't allow both operators to be overloaded in a class
12. **A postfix ++ has a dummy parameter**
13. A prefix ++ has a dummy parameter
14. By making prefix ++ as a global function and postfix as a member function
15. Which of the following operator functions cannot be global, i.e., must be a member function.
16. new
17. delete
18. **Conversion Operator**
19. All of the above
20. \_\_\_\_\_\_ is not an operator overloaded by the C++ language.
21. <<.
22. +.
23. **pow**
24. >>.
25. We can overload almost all the C++ operators except the following.

i) Class member operator (.,.\*) ii) Assignment operator (=)

iii) Scope resolution operator (::) iv) Conditional operator (?:)

1. i, ii and iii only
2. ii, iii and iv only
3. **i, iii and iv only**
4. All i, ii, iii and iv
5. Which of the following is the correct order involves in the process of operator overloading.

i) Define the operator function to implement the required operations.

ii) Create a class that defines the data type that is to be used in the overloading operation.

iii) Declare the operator function op() in the public part of the class.

1. 1-i, 2-ii, 3-iii
2. **1-ii, 2-iii, 3-i**
3. 1-ii, 2-i, 2-iii
4. 1-iii, 2-ii, 3-i
5. To perform the conversion from any other data type or class to a class type, a ………….. should be used in the destination class.
6. casting operator
7. **constructor**
8. not applicable
9. operator function
10. The general form of an overloaded casting operator function usually referred to as a ……………..
11. casting function
12. operator function
13. **conversion function**
14. overloaded function
15. When using ………………., overloaded through a member function, the left-hand operand must be an object of the relevant class.
16. Unary operators
17. **Binary operators**
18. Arithmetic operators
19. Function operator
20. Operator overloading is done with the help of a special function called ……………, which describes the special task of an operator.
21. overloading function
22. special task function
23. detail function
24. **operator function**
25. The compiler does not support automatic type conversions for the ………….. data type.
26. basic
27. **user-defined**
28. class
29. automatic
30. Which of the following keyword is used to overload operators in C++?
31. overload
32. **operator**
33. operate
34. op
35. While overloading binary operators using member function, it requires \_\_\_ argument/s.
36. Zero
37. **One**
38. Two
39. Three
40. In case of operator overloading, operator function must be \_\_\_\_\_\_ .

1. Static member functions

2. Non- static member functions

3. Friend Functions

1. Only 2
2. Only 1, 3
3. **Only 2 , 3**
4. All 1 , 2, 3
5. Scope resolution operator is used\_\_\_\_\_\_ .
6. to resolve the scope of global variables only
7. to resolve the scope of functions of the classes only
8. **to resolve scope of global variables as well as functions of the classes**
9. None of these