1. What is a class?

Ans. A class is a template/skeleton for creating an object which consists of data member and member functions.

2. The class members are by default …….
   (a) Public
   (b) Private
   (c) Protected
   (d) Inherited
   Ans. Private
3. Which of the following is false about classes
   (a) Permits the data to be hidden from other classes
   (b) It contains data member and member functions
   (c) It brings together all aspects of an entity in one place
   (d) All of the above.
   Ans. (c)
4. What is the difference between the class and structure in c++?

Ans. The data members in class are by default private whereas the data members are public by default in case of a structure.

5. What is an object?

Ans. An object is an entity that has properties for validations, methods for functionality and events for depicting the change of the state.

6. Which of the following helps you to find out the dynamic type of an object when you have only pointer or a reference to the base type?
   (a) Object type
   (b) Abstract data type
   (c) Reference type
   (d) Runtime type Identification
   Ans. (d) Runtime type Identification
   Explanation: RTTI, Runtime type Identification is the type which finds the type of an object and converts the type of a pointer or reference (i.e. dynamic typing).
7. What is the size of an object of an empty class?
   (a) 0
   (b) Undefined
   (c) 1
   (d) Garbage value
   Ans. (c) 1

Explanation: C++ allows us a feature where we can create an object of an empty class. At the time of creating the object some bytes of memory has been initialized. The minimum amount of memory created is 1 byte. But sometimes it may change depending on the compiler.

8. Which of these means exposing the essential data of an object and hiding the background details.
   (a) Inheritance
   (b) Encapsulation
   (c) Data abstraction
   (d) Polymorphism
   Ans.(c) Data abstraction
   Explanation: Data abstraction is the process of exposing the essential data of an object to the outside of the world and hiding the low level data of that object.
9. Which operator is used to create a memory space of specified type and returns the starting address to the pointer variable.
   (a) malloc
   (b) New
   (c) Calloc
   (d) None of the above
   Ans. (b) new operator.
   Explanation: int *p = new int[10]; In this statement we can say that, new operator creates an array of 10 integers and allocates 20 bytes of memory space & returns the starting position or starting memory location to the pointer variable p. 10 is being assigned to the pointer variable.
10. What is the significance of endl in c++?
    (a) Creates a new line
    (b) Creates space between two words
    (c) Removes a line
    (d) None
    Ans. (a) creates a new line.
    Explanation: endl has the same effect as that of the "\n" in 'C'.
11. Private members can be accessed
    (a) Within the class
    (b) Outside the class
    (c) Both within and outside the class
    (d) None of the above
    Ans. (a) within the class
    Explanation: Private members can be accessed only from within the class and they cannot be accessed from outside of the class.

12. The data members in the class are usually declared as private and the member functions are declared as …….
    (a) public
    (b) private
    (c) protected
    (d) inherited
    Ans. (b) private
13. The operator :: is known as:
    (a) Conditional operator
    (b) Extraction operator
    (c) Scope resolution operator
    (d) Field width operator
    Ans. (c) Scope resolution operator
14. What is the difference between a member function and a normal function?

Ans. The difference between a member function and a normal function is that a member function uses a membership 'identity' label in the header to indicate the class to which it belongs.

15. Can a member function call another member function directly?
    (a) Partially yes
    (b) No
    (c) Partially no
    (d) Yes
    Ans. (d) Yes
    Explanation: A member function can call a non member function directly, without using the dot operator.
16. What is the name of the function which is defined inside the class
    (a) Inline function
    (b) normal function
    (c) data function
    (d) none
    Ans. (a) Inline function
17. What is the advantage of Inline function?

Ans. The main advantage of the inline function is that it eliminates the code duplication as a result of which a lot of memory space is being saved.

18. Identify the error in the following program

```
#include <iostream>
using namespace std;
class room
```

```
{
  int width,height;
  void setvalue(int w, int h)
  {
            width = w;
            height = h;
  }
};
int main()
{
  room r1;
  r1.width = 12;
  return 0;
}
```

Ans.r1.width = 12;

Explanation: room :: width is not accessible in the function main().

19. Can member functions be nested?
    (a) Can't predict
    (b) Yes
    (c) No
    (d) May be
    Ans. (b) Yes
    Explanation: A member function can be called by using its name inside another member function of the same class and this is known as nesting of member functions.
20. What are the four default functions available for each object?
    (a) Copy constructor, Inline function, constructor,
    (b) Constructor, copy constructor, Inline function,
    (c) Inline function, default constructor, destructor, constructor
    (d) Constructor, destructor, copy constructor, default constructor
    Ans. (d)
    Explanation: Each c++ object has constructor, destructor, copy constructor, default constructor as default functions.
21. Static data members are also known as …….

Ans. Class variables.

Explanation: Static data members are also known as class variables because they are associated with the class itself rather than with any class object.

22. What is the value of the static member variable when the first object of a class is created?

    (a) Undefined

    (b) 1

    (c) 0

    (d) Garbage value

    Ans. (c) 0

    Explanation: By default the static member variable is 0.

23. What is the output of the following program ?

```
#include <iostream>
using namespace std;
class Foo
{
 public:
 Foo()
 {
            ++i;
            cout << "Number:" << i;
           }
 private:
           static int i;
};
int Foo::i = 0;       // allocate memory for numFoos, and initialize it
int main()
{
           Foo f1;
           Foo f2;
           Foo f3;
           return 0;
}
```

    (a) Number:1 Number:1 Number:1

    (b) Number:1 Number:2 Number:3

    (c) Number:0 Number:1 Number:2

    (d) Number:0 Number:0 Number:0

    Ans.(b) Number:1 Number:2 Number:3

Explanation: variable i is a static variable and hence at the time of object creation only one copy of the member is created and is common to the entire class. It is shared by all the objects of class irrespective of how many objects created.

24. What is the scope of the static data members?
    (a) Inside the class
    (b) Outside the class
    (c) Both (a) and (b)
    (d) None of the above.
    Ans. (b) Outside the class
    Explanation: The scope of the static data members is specified outside the class because the static data members are stored separately in the memory rather than as a part of the object.
25. Static methods can use non static variables from its class.
    (a) True
    (b) False
    Ans. (b) false
    Explanation: Non-static variables allocate memory only when object of the class is created, but static methods can be invoked without creating object. So static methods are not allowed to use non-static variables.
26. 'this' pointer is only accessible in the static member class
    (a) True
    (b) False
    Ans. (b) false
27. 'this' pointer points to ……

Ans. 'this' pointer points to the object for which the member function is called.

28. Which operator is used to access a member when we use pointers to both the object and the member?
    (a) >>
    (b) ->*
    (c) *
    (d) ::
    Ans. (b) ->*
    Explanation: Dereferencing operator ->* or .* is used to access a member when we use pointers to both the object and the member.
29. When a constructor is called?
    (a) After the object is created
    (b) At the time of object creation
    (c) At the run time
    (d) At the compile time

Ans. (b) At the time of object creation.

30. Does constructor have a return type?

    (a) Yes

    (b) No

    (c) Can't say

    Ans. (b) No

    Explanation: Constructor does not have a return type not even void.

31. A constructor which is defined inside the class is known as

    (a) Copy constructor

    (b) Default constructor

    (c) Inline constructor

    (d) Parameterized constructor

    Ans. (c) Inline constructor

32. What is a copy constructor?

Ans. When a constructor takes the argument which is the type of same class itself then this kind of constructor is called copy constructor. We can also define it as passing the reference of an object as an argument to the constructor. In this copy constructor, the parameter of the constructor can be of any type except the class to which it belongs.

33. The destructors are invoked by ……

Ans. The destructors are invoked by the compiler upon the exit of the program.

34. Constructors, like other member functions, can be declared anywhere in the class

    (a) True

    (b) False

    Ans. (a) True

35. Allocation of memory to objects at the time of their construction is known as …….

    (a) Dynamic memory allocation

    (b) pointers

    (c) dynamic construction

    (d) static construction

    Ans. (c) Dynamic construction

36. Which operator is used to signify the namespace?

    (a) conditional operator

    (b) ternary operator

    (c) scope operator

    (d) none of the mentioned

    Ans. (c)

37. Identify the correct statement.

    (a) Namespace is used to group class, objects and functions.

    (b) Namespace is used to mark the beginning of the program.

    (c) Namespace is used to separate the class, objects.

    (d) None of the above

    Ans. (a)

    Explanation: Namespace allow you to group class, objects and functions. It is used to divide the global scope into the sub-scopes.

38. What is the use of Namespace?

    (a) To encapsulate the data

    (b) To structure a program into logical units.

    (c) Both a and b

    (d) none of the mentioned

    Ans. (b)

    Explanation: The main aim of the namespace is to understand the logical units of the program and to make the program so robust.

39. What is the general syntax for accessing the namespace variable?

    (a) namespaceid::operator

    (b) namespace,operator

    (c) namespace#operator

    (d) none of the mentioned

    Ans. (a)

40. What is the output of this program?

```
#include <iostream>
using namespace std;
namespace first
{
   int var = 5;
}
namespace second
{
   double var = 3.1416;
}
int main ()
{
   int a;
```

```
    a = first::var + second::var;
    cout << a;
    return 0;
}
```

(a) 8.31416

(b) 8

(c) 9

(d) compile time error

Ans. (b)

Explanation: As we are getting two variables from namespace variable and we are adding that.

Output:

41. _____keyword is used to access the variable in namespace?

Ans. using

42. How many access specifiers are there in c++?

(a) 1

(b) 2

(c) 3

(d) 4

Ans. (c)

Explanation: There are three access specifiers in c++. They are public, private and protected.

43. What of the following describes protected access specifier?

(a) The variable is visible only outside inside the block

(b) The variable is visible everywhere

(c) The variable is visible to its block and to its derived class

(d) None of the mentioned

Ans. (c)

44. To which of the following access specifiers are applicable?

(a) Member data

(b) Functions

(c) Both a & b

(d) None of the mentioned

Ans.(c)

Explanation: The access specifiers can be applicable to the member data and functions because they need to be accessed outside the block.

45. What is the output of this program?

```
#include <iostream>
```

```
using namespace std;
class Cat
{
   public:
   int age;
   int weight;
};
int main()
{
   Cat f;
   f.age = 56;
   cout << "Gates is " ;
   cout << f.age << " years old.\n";
}
```

(a) Gates is

(b) Gates is 56 years old

(c) Error

(d) None of the mentioned

Ans. (b)

Explanation: In this program, We passed the value from main function to class and returning it to the main and then printing it.

46. What is the use of private member function?

Ans . The data or some functions may require certain functions to be hidden from outside calls. A private member function can only be called by another function that is a member of its class.

47. Can we invoke a private function using the dot operator?

Ans. No.

Explanation: A private member function can only be called by another function that is a member of its class.

48. Which class is used to design the base class?

(a) abstract class

(b) derived class

(c) base class

(d) None of the mentioned

Ans. (a)

49. Which is used to create a pure virtual function?

(a) $

(b) =0

(c) &

(d) !

Ans. (b)

Explanation: For making a method as pure virtual function, We have to append '=0' to the class or method.

50. Which of the following is also called as abstract class?

(a) virtual function

(b) pure virtual function

(c) derived class

(d) none of the mentioned

Ans. (b)

Explanation: Classes that contain at least one pure virtual function are called as abstract base classes.

51. Pick out the correct option.

(a) We cannot make an instance of an abstract base class

(b) We can make an instance of an abstract base class

(c) Both a & b

(d) None of the mentioned

Ans. (a)

52. Where does the abstract class is used?

(a) base class only

(b) derived class

(c) both a & b

(d) None of the mentioned

Ans. (a)

53. What is ADT?

Ans. Abstract Data Type is an encapsulation of set data objects and set of operations on those data objects or that includes only the data representation on specific data type that performs operations on that type.

54. Where we can implement the ADT?

(a) Predefined data types

(b) User defined data types

(c) Both a & b

(d) None of the above

Ans. (c)

55. A type definition that allows program units to declare variables of the type but hides the representation of these _____.

Ans. Variables

56. What is information hiding?

Ans. Information hiding refers to the characteristic that an encapsulation such as an ADT "hides" the internal structure of the data from the user of the ADT.

57. Information hiding can be implemented by _____.

Ans. Access labels.

58. Which rule will not affect the friend function?

    (a) private and protected members of a class cannot be accessed from outside

    (b) private and protected member can be accessed anywhere

    (c) both a & b

    (d) None of the mentioned

    Ans. (a)

    Explanation: Friend is used to access private and protected members of a class from outside the same class.

59. Which keyword is used to declare the friend function?

    (a) Friend

    (b) Classfriend

    (c) Myfriend

    (d) None of the above

    Ans. (a)

60. What is the syntax of friend function?

    (a) friend class1 Class2;

    (b) friend class;

    (c) friend class

    (d) None of the mentioned

    Ans. (a)

    Explanation: In option a, the class2 is the friend of class1 and it can access all the private and protected members of class1.

61. What is the output of following program

```
#include <iostream>
 using namespace std;
 class sample;
 class sample1
{
       int width, height;
       public:
       int area ()
      {
          return (width * height);}
```

```
        void convert (sample a);
      };
class sample
{
    private:
   int side;
    public:
   void set_side (int a)
   {
       side = a;
   }
  friend class sample1;
};
void sample1::convert (sample a)
{
   width = a.side;
    height = a.side;
}
int main ()
{
    sample sqr;
    sample1 rect;
    sqr.set_side(6);
    rect.convert(sqr);
     cout << rect.area();
     return 0;
}
```

(a) 24

(b) 35

(c) 16

(d) 36

Ans. (d)

Explanation: In this program, we are using the friend for the class and calculating the area of the square.