1. Wrapping data and its related functionality into a single entity is known as \_\_\_\_\_\_\_\_\_\_\_\_\_  
a) Abstraction  
b) Encapsulation  
c) Polymorphism  
d) Modularity

2. How structures and classes in C++ differ?  
a) In Structures, members are public by default whereas, in Classes, they are private by default  
b) In Structures, members are private by default whereas, in Classes, they are public by default  
c) Structures by default hide every member whereas classes do not  
d) Structures cannot have private members whereas classes can have

3. What does polymorphism in OOPs mean?  
a) Concept of allowing overriding of functions  
b) Concept of hiding data  
c) Concept of keeping things in different modules/files  
d) Concept of wrapping things into a single unit

4. Which concept allows you to reuse the written code?  
a) Encapsulation  
b) Abstraction  
c) Inheritance  
d) Polymorphism

5. Which of the following explains Polymorphism?  
a)

int func(int, int);

float func1(float, float);

b)

int func(int);

int func(int);

c)

int func(float);

float func(int, int, char);

d)

int func();

int new\_func();

6. Which of the following shows multiple inheritances?  
a) A->B->C  
b) A->B; A->C  
c) A,B->C  
d) B->A

7. C++ is \_\_\_\_\_\_\_\_\_\_\_\_\_\_  
a) procedural programming language  
b) object oriented programming language  
c) functional programming language  
d) both procedural and object oriented programming language

8. What does modularity mean?  
a) Hiding part of program  
b) Subdividing program into small independent parts  
c) Overriding parts of program  
d) Wrapping things into single unit

9. Which of the following feature of OOPs is not used in the following C++ code?

class A

{

int i;

public:

void print(){cout<<"hello"<<i;}

}

class B: public A

{

int j;

public:

void assign(int a){j = a;}

}

a) Abstraction  
b) Encapsulation  
c) Inheritance  
d) Polymorphism

10. Which of the following is correct?  
a) Base class pointer object cannot point to a derived class object  
b) Derived class pointer object cannot point to a base class object  
c) A derived class cannot have pointer objects  
d) A base class cannot have pointer objects

11. Out of the following, which is not a member of the class?  
a) Static function  
b) Friend function  
c) Constant function  
d) Virtual function

12. What is the other name used for functions inside a class?  
a) Member variables  
b) Member functions  
c) Class functions  
d) Class variables

13. Which of the following cannot be a friend?  
a) Function  
b) Class  
c) Object  
d) Operator function

14. Why references are different from pointers?  
a) A reference cannot be made null  
b) A reference cannot be changed once initialized  
c) No extra operator is needed for dereferencing of a reference  
d) All of the mentioned

15. Which of the following is an abstract data type?  
a) int  
b) float  
c) class  
d) string

16. Which of the following cannot be used with the virtual keyword?  
a) Class  
b) Member functions  
c) Constructors  
d) Destructors  
View Answer

17. Which concept is used to implement late binding?  
a) Virtual functions  
b) Operator functions  
c) Constant functions  
d) Static functions

18. Which of the following is correct?  
a) C++ allows static type checking  
b) C++ allows dynamic type checking.  
c) C++ allows static member function to be of type const.  
d) C++ allows both static and dynamic type checking

19. Which members are inherited but are not accessible in any case?  
a) Private  
b) Public  
c) Protected  
d) Both private and protected

20. Which of the following is correct?  
a) Friend functions can access public members of a class  
b) Friend functions can access protected members of a class  
c) Friend functions can access private members of a class  
d) All of the mentioned

21. Which of the following is used to make an abstract class?  
a) By using virtual keyword in front of a class declaration  
b) By using an abstract keyword in front of a class declaration  
c) By declaring a virtual function in a class  
d) By declaring a pure virtual function in a class

22. What will be the output of the following C++ code?

#include <iostream>

using namespace std;

class A{

public:

A(){

cout<<"Constructor called**\n**";

}

~A(){

cout<<"Destructor called**\n**";

}

};

int main(int argc, char const \*argv[])

{

A \*a = new A[5];

delete a;

return 0;

}

a) “Constructor called” five times and then “Destructor called” five times  
b) “Constructor called” five times and then “Destructor called” once  
c) Error  
d) Segmentation fault

23. What is the correct syntax of declaring array of pointers of integers of size 10 in C++?  
a) int arr = new int[10];  
b) int \*\*arr = new int\*[10];  
c) int \*arr = new int[10];  
d) int \*arr = new int\*[10];

24. What happens if the following C++ statement is compiled and executed?

int \*ptr = NULL;

delete ptr;

a) The program compiled successfully but throws an error during run-time  
b) The program gives a compile-time error  
c) The program is not semantically correct  
d) The program is compiled and executed successfully