C++ Multiple Choice Questions

Q1. Number of keywords available in C++

1. 32
2. 27
3. 31
4. 63

Answer: 4 (63 keywords: 32+31)

Q2. Which language is purely OOP?

1. Smalltalk
2. CPP
3. Simula
4. Java

Answer: 1 (Smalltalk) - Note: Simula is the first OOP language

---

Q3. First OOP Language in 1960

1. Smalltalk
2. CPP
3. Simula
4. Java

Answer: 3 (Simula)

Q4. Function having same name but differs in arguments

Function having same name but differs either in different number of arguments or type of arguments or order of arguments such process of writing function is called function _____

1. overloading

2. overriding
3. both 1 and 2
4. none of above

Answer: 1 (Function Overloading - same name but different signature)

Q5. Operator used with cin (>>)

1. extraction
2. insertion
3. in
4. out

Answer: 1 (Extraction operator for cin object of istream class)

Q6. Operator used with cout (<<)

1. extraction
2. insertion
3. in
4. out

Answer: 2 (Insertion operator for cout object of ostream class)

Q7. Values stored in data members of object

The values stored in data members of the object called as _____ of object.

1. state
2. behavior
3. identity
4. none of above

Answer: 1 (State - Object has state, behaviour and unique identity)

Q8. What is decided by member functions?

1. state
2. behavior
3. identity
4. none of above

Answer: 2 (Behavior)

Q9. Default arguments order

1. right to left order
2. left to right order
3. depends on compiler
4. none of above

Answer: 1 (Right to left - to avoid comma separator confusion)

Q10. Size of empty class object

1. 1 byte
2. 8 byte
3. 8 bits
4. 4 bytes
5. both 1 and 3

Answer: 5 (1 byte or 8 bits)

Q11. Inline function replacement exceptions

Every function may not be replaced by compiler, rather it avoids replacement in certain cases like ____ may not be replaced

1. function containing switch, loop
2. recursion

3. both 1 and 2
4. none of above

Answer: 3 (Function having loop, switch, recursion)

Q12. First POP Language

1. FORTRON
2. PASCAL
3. C
4. NONE OF ABOVE

Answer: 1 (FORTRON)

Q13. First object based language

1. Ada
2. visual basic
3. module2
4. none of above

Answer: 1 (Ada)

Q14. Data types added by C++

C++ supports all data types provided by C language and C++ adds data types

1. bool
2. wchar_t
3. both 1 and 2
4. none of above

Answer: 3 (wchar_t, bool)

Q15. Compile time polymorphism achieved by

1. function overloading
2. operator overloading
3. function overriding
4. both 1 and 2

Answer: 4 (Function overloading, operator overloading)

Q16. Removal of small object doesn't affect big object

1. association
2. aggregation
3. containment
4. none of above

Answer: 1 (Association - loose coupling)

Q17. Removal of small object affects big object

1. association
2. aggregation
3. containment
4. none of above

Answer: 2 (Aggregation - tight coupling)

Q18. Default inheritance mode in C++

1. private
2. protected
3. public
4. none of above

Answer: 1 (Private)

Q19. Function called based on object type, not pointer type

1. virtual function
2. static function
3. const function
4. global function

Answer: 1 (Virtual function)

Q20. Class with at least one pure virtual function

1. abstract class
2. concrete class
3. both 1 and 2
4. none of above

Answer: 1 (Abstract class)

Q21. Storing derived class object address in base class pointer

1. up casting
2. down casting
3. object slicing
4. none of above

Answer: 1 (Upcasting)

Q22. Storing base class object address in derived class pointer

1. up casting
2. down casting
3. object slicing
4. none of above

Answer: 2 (Downcasting)

Q23. Assigning derived class object to base class object

When we assign derived class object to the base class object at that time base class portion which is available in derived class object is assign to the base class object.

1. up casting
2. down casting
3. object slicing
4. none of above

Answer: 3 (Object slicing)

Q24. Pointer pointing to unavailable memory

1. dangling pointer
2. null pointer
3. huge pointer
4. far pointer

Answer: 1 (Dangling pointer)

Q25. Called automatically when object is created

1. mutator
2. constructor
3. destructor
4. copy constructor

Answer: 2 (Constructor)

Q26. True statement about abstract class

1. An abstract class can be instantiated using new operator

2. An abstract class is designed only to be inherited by other classes
3. An abstract class can not have data members and member function declarations
4. abstract class can not have constructor and destructor

Answer: 2

Q27. Function invoked when object goes out of scope

1. static
2. friend
3. exception handler
4. destructor
5. constructor

Answer: 4 (Destructor)

Q28. Mechanism to acquire properties of another class

1. encapsulation
2. data hiding
3. abstraction
4. inheritance

Answer: 4 (Inheritance)

Q29. Derived class inherits from multiple base classes

1. multilevel inheritance
2. single inheritance
3. multiple inheritance
4. hybrid inheritance
5. hierarchical inheritance

Answer: 3 (Multiple inheritance)

Q30. One base class, multiple derived classes

1. multilevel inheritance
2. single inheritance
3. multiple inheritance
4. hybrid inheritance
5. hierarchical inheritance

Answer: 5 (Hierarchical inheritance)

Q31. Single inheritance with multiple levels

1. multilevel inheritance
2. single inheritance
3. multiple inheritance
4. hybrid inheritance
5. hierarchical inheritance

Answer: 1 (Multilevel inheritance)

Q32. One base class, one derived class

1. multilevel inheritance
2. single inheritance
3. multiple inheritance
4. hybrid inheritance
5. hierarchical inheritance

Answer: 2 (Single inheritance)

Q33. Incorrect statement about static member function

1. static member function can be called by object of that class

2. static member function can be called without creating object of that class ie by class name only
3. static member function can be called by non static member function
4. static function can not access only static data member

Answer: 4

Q34. Not a key component of OOPs

1. inheritance
2. polymorphism
3. encapsulation
4. virtualization

Answer: 4 (Virtualization)

Q35. Class defined in another class

1. nested class
2. inheritance
3. encapsulation
4. containship

Answer: 1 (Nested class)

Q36. Keyword to refer current object

1. this
2. static
3. friend
4. abstract
5. const

Answer: 1 (this)

Q37. OOP concept: Multiple roles in life

Statement: I have many roles in life teacher, employee, student, cricket player and many more.

1. abstraction
2. polymorphism
3. data hiding
4. composition
5. inheritance

Answer: 2 (Polymorphism)

Q39. Code output prediction

```cpp
#include<iostream>
using namespace std;
int main(int argc, char *argv[], char *envp[])
{
   int a=5;
   int &b=a;
   int c=10;
   b=c;
   cout<<a<<" "<<b<<endl;
   c=20;
   cout<<a<<" "<<b<<endl;
   return 0;
}
```

Options:

1. 10 10 / 20 20
2. 10 5 / 20 20
3. 5 10 / 20 20
4. 10 10 / 10 10

Answer: 4 (10 10 / 10 10)

Q40. Member function declared in base, redefined in derived

1. constructor
2. destructor
3. static function
4. friend function
5. virtual function

Answer: 5 (Virtual function)

Q41. Every non-const member function is a

1. constructor
2. destructor
3. mutator
4. friend

Answer: 3 (Mutator)

Q42. Class is a

1. build in type
2. user define type
3. reference type
4. primitive type

Answer: 2 (User defined datatype)

Q43. Not true about destructor

1. it is a member function
2. it is used to finalize object
3. it does not have any return value
4. it does not have any parameter

5. it accept class object as parameter

Answer: 5

Q44. True statements about destructor

1. it is a member function
2. it is used to finalize object
3. it does not have any return value
4. it does not have any parameter
5. all of above

Answer: 5 (All of above)

Q45. Correct pure virtual function declaration

1. virtual void calculate();
2. virtual void calculate()=0;
3. void calculate()=0;
4. virtual calculate();

Answer: 2 (virtual void calculate()=0;)

Q46. To eliminate side effects of macro

1. inline function
2. static function
3. abstract class
4. virtual function
5. pure virtual function

Answer: 1 (Inline function)

Q47. C++ developed by

1. Alan Kay
2. Bjarne Stroustrup
3. James Gosling
4. Brian Kernighan

Answer: 2 (Bjarne Stroustrup)

Q48. C++ invented in year

1. 1972
2. 1979
3. 1983
4. 1998

Answer: 2 (1979)

Q49. Properly defined structure

1. struct {int a;}
2. struct a_struct {int a;}
3. struct a_struct int a;
4. struct a_struct {int a;};

Answer: 4 (struct a_struct {int a;};)

Q50. Private and public are known as

1. Accessors
2. Access Specifier
3. visibility Manipulator
4. Manipulator

Answer: 2 (Access Specifier)

Q51. Function called without arguments

1. void add(int x, int y=0)
2. void add(int=0)
3. void add(int x=0, int y=0)
4. void add(char c)

Answer: 3 (void add(int x=0, int y=0))

Q52. Valid class declaration

1. class A { int x; };
2. class B { }
3. public class A { }
4. object A { int x; };

Answer: 1 (class A { int x; };)

Q53. Default access for class members in C++

1. protected
2. private
3. public
4. public & protected

Answer: 2 (Private)

Q54. How constructors differ from other functions

1. Constructor has the same name as the class itself
2. Constructors do not return anything
3. Constructors are automatically called when an object is created
4. All of the mentioned

Answer: 4 (All of the mentioned)

Q55. 'this' pointer is

1. nonconstant & externally
2. constant & externally
3. constant & internally
4. nonconstant & internally

Answer: 3 (constant & internally)

Q56. Destructor description

1. A special function that is called to free the resources, acquired by the object
2. A special function that is called to delete the class
3. A special function that is called anytime to delete an object
4. A special function that is called to delete all the objects of a class

Answer: 1

Q57. Syntax for accessing namespace variable

1. namespace::variable_name
2. namespace,variable_name
3. namespace#variable_name
4. namespace$variable_name

Answer: 1 (namespace::variable_name)

Q58. Syntax for destructor of class A

1. A(){}
2. ~A(){}
3. A::A(){}

4. ~A(){};

Answer: 2 (~A(){})

Q59. Keyword to access variable in namespace

1. using
2. dynamic
3. const
4. static

Answer: 1 (using)

Q60. Standard namespace in C++

1. global namespace
2. std namespace
3. default namespace
4. system namespace

Answer: 2 (std namespace)

Q61. Default value passing in C++

1. call by value
2. call by reference
3. call by address
4. All of above

Answer: 1 (Call by value)

Q62. How constants are declared

1. const keyword

2. #define preprocessor
3. both const keyword and #define preprocessor
4. $define

Answer: 1 (const keyword)

Q63. Keyword to modify non-const data in const function

Inside constant member function, if we want to modify state of non constant data member then we should use _____ keyword?

1. static
2. immutable
3. mutable
4. mutator

Answer: 3 (mutable)

Q64. Syntax for defining static data members

1. dataType className :: memberName = value;
2. dataType className : memberName = value;
3. dataType className . memberName = value;
4. dataType className -> memberName =value

Answer: 1 (dataType className :: memberName = value;)

Q65. Operator that cannot be overloaded

1. =
2. []
3. ()
4. ? :

Answer: 4 (? : ternary operator)

Q66. OOP feature for code reusability

1. Polymorphism
2. Abstraction
3. Encapsulation
4. Inheritance

Answer: 4 (Inheritance)

Q67. Output of code with reference to literal

```cpp
#include<iostream>
using namespace std;
int main()
{
   int &a=5;
   cout<<a<<endl;
   return 0;
}
```

1. 5
2. segmentation fault
3. Runtime error
4. compile time error

Answer: 4 (Compile time error)

Q68. Not compile time polymorphism

1. Function Overloading
2. Operator Overloading
3. Function Overriding
4. Template

Answer: 3 (Function Overriding)

Q69. Operators allowed to overload using member function

1. =
2. [ ]
3. ->
4. ( )
5. all of above

Answer: 5 (All of above)

Q70. Output of reference declaration code

```cpp
#include<iostream>
using namespace std;
int main(void)
{
   int &num;
   int a=5;
   &num=a;
   cout<<num;
   return 0;
}
```

1. 5
2. Segmentation fault
3. Runtime error
4. Compile time error

Answer: 4 (Compile time error)

Source: Sunbeam Infotech - CPP Multiple Choice Questions by Rahul Kale