OOPs MCQ on Constructors and Destructors

1. Which among the following is called first, automatically, whenever an object is created?
a) Class
b) Constructor
c) New
d) Trigger
View Answer
2. Which among the following is not a necessary condition for constructors?
a) Its name must be same as that of class
b) It must not have any return type
c) It must contain a definition body
d) It can contains arguments
View Answer
3. Which among the following is correct?
a) class student{ public: int student(){} };
b) class student{ public: void student (){} };
c) class student{ public: student{}{} };
d) class student{ public: student(){} };

4. In which access should a constructor be defined, so that object of the class can be created in any function?
a) Public
b) Protected
c) Private
d) Any access specifier will work
View Answer
5. How many types of constructors are available for use in general (with respect to parameters)?
a) 2
b) 3
c) 4
d) 5

6. If a programmer defines a class and defines a default value parameterized constructor inside it.
He has not defined any default constructor. And then he try to create the object without passing arguments, which among the following will be correct?
a) It will not create the object (as parameterized constructor is used)
b) It will create the object (as the default arguments are passed)
c) It will not create the object (as the default constructor is not defined)
d) It will create the object (as at least some constructor is defined)
View Answer
7. Default constructor must be defined, if parameterized constructor is defined and the object is to be created without arguments.
a) True

b) False
View Answer
8. If class C inherits class B. And B has inherited class A. Then while creating the object of class C, what will be the sequence of constructors getting called?
a) Constructor of C then B, finally of A
b) Constructor of A then C, finally of B
c) Constructor of C then A, finally B
d) Constructor of A then B, finally C
View Answer
9. In multiple inheritance, if class C inherits two classes A and B as follows, which class constructor will be called first?

```
class A{ };
class B{ };
class C: public A, public B{  };
```

a) A()
b) B()
c) C()
d) Can't be determined
View Answer
10. Which among the following is true for copy constructor?
a) The argument object is passed by reference
b) It can be defined with zero arguments
c) Used when an object is passed by value to a function
d) Used when a function returns an object
View Answer
11. If the object is passed by value to a copy constructor?
a) Only public members will be accessible to be copied
b) That will work normally
c) Compiler will give out of memory error
d) Data stored in data members won't be accessible
View Answer
12. Which object will be created first?

```
class student
{
    int marks;
};
student s1, s2, s3;
```

a) s1 then s2 then s3
b) s3 then s2 then s1
c) s2 then s3 then s1
d) all are created at same time
View Answer
13. Which among the following helps to create a temporary instance?
a) Implicit call to a default constructor
b) Explicit call to a copy constructor

c) Implicit call to a parameterized constructor
d) Explicit call to a constructor
View Answer
14. Which among the following is correct for the class defined below?

```
class student
{
    int marks;
    public: student(){}
    student(int x)
    {
         marks=x;
    }
};
main()
{
    student s1(100);
    student s2();
    student s3=100;
    return 0;
}
```

a) Object s3, syntax error
b) Only object s1 and s2 will be created
c) Program runs and all objects are created
d) Program will give compile time error

15. For constructor overloading, each constructor must differ in _____ and _____
a) Number of arguments and type of arguments
b) Number of arguments and return type
c) Return type and type of arguments
d) Return type and definition1. How many types of constructors are available, in general, in any language?
a) 2
b) 3
c) 4
d) 5
View Answer
2. Choose the correct option for the following code.

```
class student
{
    int marks;
}
student s1;
student s2=2;
```

a) Object s1 should be passed with argument
b) Object s2 should not be declared
c) Object s2 will not be created, but program runs
d) Program gives compile time error

3. Which constructor is called while assigning some object with another?
a) Default
b) Parameterized
c) Copy
d) Direct assignment is used4. It’s necessary to pass object by reference in copy constructor because _____
a) Constructor is not called in pass by reference
b) Constructor is called in pass by reference only
c) It passes the address of new constructor to be created
d) It passes the address of new object to be created
View Answer
5. Which specifier applies only to the constructors?
a) Public
b) Protected
c) Implicit
d) Explicit
View Answer
6. Which among the following is true?
a) Default constructor can’t be defined by the programmer
b) Default parameters constructor isn’t equivalent to the default constructor
c) Default constructor can be called explicitly
d) Default constructor is and always called implicitly only
View Answer
7. Which type of constructor can’t have a return type?
a) Default
b) Parameterized
c) Copy
d) Constructors don’t have a return type
View Answer
8. Why do we use static constructors?
a) To initialize the static members of class
b) To initialize all the members with static value
c) To delete the static members when not required
d) To clear all the static members initialized values
View Answer
9. When and how many times a static constructor is called?
a) Created at time of object destruction
b) Called at first time when an object is created and only one time
c) Called at first time when an object is created and called with every new object creation
d) Called whenever an object go out of scope
View Answer
10. Which among the following is true for static constructor?
a) Static constructors are called with every new object
b) Static constructors are used initialize data members to zero always
c) Static constructors can’t be parameterized constructors
d) Static constructors can be used to initialize the non-static members also
View Answer

11. Within a class, only one static constructor can be created.
a) True
b) False
View Answer
12. Default constructor initializes all data members as _____
a) All numeric member with some garbage values and string to random string
b) All numeric member with some garbage values and string to null
c) All numeric member with zero and strings to random value
d) All numeric member with zero and strings to null
View Answer
13. When is the static constructor called?
a) After the first instance is created
b) Before default constructor call of first instance
c) Before first instance is created
d) At time of creation of first instance
View Answer
14. If constructors of a class are defined in private access, then _____
a) The class can't be inherited
b) The class can be inherited
c) Instance can be created only in another class
d) Instance can be created anywhere in the program
View Answer
15. Which among the following is correct, based on the given code below?

```
class student
{
     int marks;
     public : student()
     {
          cout<<"New student details can be added now";
     }
};
student s1;
```

a) Cout can't be used inside the constructor
b) Constructor must contain only initializations
c) This program works fine
d) This program produces errors

1. Copy constructor is a constructor which _____
a) Creates an object by copying values from any other object of same class
b) Creates an object by copying values from first object created for that class
c) Creates an object by copying values from another object of another class
d) Creates an object by initializing it with another previously created object of same class
View Answer
2. The copy constructor can be used to _____
a) Initialize one object from another object of same type
b) Initialize one object from another object of different type
c) Initialize more than one object from another object of same type at a time

d) Initialize all the objects of a class to another object of another class
View Answer
3. If two classes have exactly same data members and member function and only they differ by class name. Can copy constructor be used to initialize one class object with another class object?
a) Yes, possible
b) Yes, because the members are same
c) No, not possible
d) No, but possible if constructor is also same

4. The copy constructors can be used to _____
a) Copy an object so that it can be passed to a class
b) Copy an object so that it can be passed to a function
c) Copy an object so that it can be passed to another primitive type variable
d) Copy an object for type casting

5. Which returning an object, we can use _____
a) Default constructor
b) Zero argument constructor
c) Parameterized constructor
d) Copy constructor6. If programmer doesn't define any copy constructor then _____
a) Compiler provides an implicit copy constructor
b) Compiler gives an error
c) The objects can't be assigned with another objects
d) The program gives run time error if copying is used
View Answer
7. If a class implements some dynamic memory allocations and pointers then _____
a) Copy constructor must be defined
b) Copy constructor must not be defined
c) Copy constructor can't be defined
d) Copy constructor will not be used

8. What is the syntax of copy constructor?
a) classname (classname &obj){ /*constructor definition*/ }
b) classname (cont classname obj){ /*constructor definition*/ }
c) classname (cont classname &obj){ /*constructor definition*/ }
d) classname (cont &obj){ /*constructor definition*/ }
View Answer
9. Object being passed to a copy constructor _____
a) Must be passed by reference
b) Must be passed by value
c) Must be passed with integer type
d) Must not be mentioned in parameter list
View Answer

10. Out of memory error is given when the object _____ to the copy constructor.
a) Is passed with & symbol
b) Is passed by reference
c) Is passed as <classname &obj>
d) Is not passed by reference
View Answer
11. Copy constructor will be called whenever the compiler _____
a) Generates implicit code
b) Generates member function calls
c) Generates temporary object
d) Generates object operations

12. The deep copy is possible only with the help of _____
a) Implicit copy constructor
b) User defined copy constructor
c) Parameterized constructor
d) Default constructor
View Answer
13. Can a copy constructor be made private?
a) Yes, always
b) Yes, if no other constructor is defined
c) No, never
d) No, private members can’t be accessed
View Answer
14. The arguments to a copy constructor _____
a) Must be const
b) Must not be cosnt
c) Must be integer type
d) Must be static
View Answer
15. Copy constructors are overloaded constructors.
a) True
b) False

1. Which among the following best describes constructor overloading?
a) Defining one constructor in each class of a program
b) Defining more than one constructor in single class
c) Defining more than one constructor in single class with different signature
d) Defining destructor with each constructor
View Answer
2. Can constructors be overloaded in derived class?
a) Yes, always
b) Yes, if derived class has no constructor
c) No, programmer can’t do it
d) No, never
View Answer

3. Does constructor overloading include different return types for constructors to be overloaded?
a) Yes, if return types are different, signature becomes different
b) Yes, because return types can differentiate two functions
c) No, return type can't differentiate two functions
d) No, constructors doesn't have any return type

4. Which among the following is possible way to overload constructor?
a) Define default constructor, 1 parameter constructor and 2 parameter constructor
b) Define default constructor, zero argument constructor and 1 parameter constructor
c) Define default constructor, and 2 other parameterized constructors with same signature
d) Define 2 default constructors

5. Which constructor will be called from the object created in the code below?

```
class A
{
        int i;
        A()
        {
                i=0; cout&lt;&lt;i;
        }
        A(int x=0)
        {
                i=x;  cout&lt;&lt;I;
        }
};
A obj1;
```

a) Default constructor
b) Parameterized constructor
c) Compile time error
d) Run time error
View Answer
6. Which among the following is false for a constructor?
a) Constructors doesn't have a return value
b) Constructors are always user defined
c) Constructors are overloaded with different signature
d) Constructors may or may not have any arguments being accepted
View Answer
7. When is the constructor called for an object?
a) As soon as overloading is required
b) As soon as class is derived
c) As soon as class is created
d) As soon as object is created
View Answer
8. Which among the following function can be used to call default constructor implicitly in java?
a) this()
b) that()
c) super()

d) sub()
View Answer
9. Why do we use constructor overloading?
a) To use different types of constructors
b) Because it's a feature provided
c) To initialize the object in different ways
d) To differentiate one constructor from another
View Answer
10. If programmer have defined parameterized constructor only, then _____
a) Default constructor will not be created by the compiler implicitly
b) Default constructor will be created by the compiler implicitly
c) Default constructor will not be created but called at runtime
d) Compile time error
View Answer
11. Which among the following is not valid in java?
a) Constructor overloading
b) Recursive constructor call
c) Default value constructors
d) String argument constructor
View Answer
12. Which constructor will be called from the object obj2 in the following program?

```
class A
{
        int i;
        A()
        {
                i=0;
        }
        A(int x)
        {
                i=x+1;
        }
        A(int y, int x)
        {
                i=x+y;
        }
};
A obj1(10);
A obj2(10,20);
A obj3;
```

a) A(int x)
b) A(int y)
c) A(int y, int x)
d) A(int y; int x)
View Answer
13. What are we only create an object but don't call any constructor for it in java?
a) Implicit constructor will be called
b) Object is initialized to some null values

c) Object is not created
d) Object is created but points to null
View Answer
14. Which among the following is false?
a) Constructor can’t be overloaded in Kotlin
b) Constructors can’t be called recursively in java
c) Constructors can be overloaded in C++
d) Constructors overloading depends on different signatures
View Answer
15. Which is correct syntax?
a) classname objectname= new() integer;
b) classname objectname= new classname;
c) classname objectname= new classname();
d) classname objectname= new() classname();

1. Which among the following best describes the constructors?
a) A function which is called whenever an object is referenced
b) A function which is called whenever an object is created to initialize the members
c) A function which is called whenever an object is assigned to copy the values
d) A function which is called whenever an object is to be given values for members
View Answer
2. Which among the following best describes destructor?
a) A function which is called just before the objects are destroyed
b) A function which is called after each reference to the object
c) A function which is called after termination of the program
d) A function which is called before calling any member function
View Answer
3. Which among the following represents correct constructor?
a) ()classname
b) ~classname()
c) –classname()
d) classname()

4. Which among the following is correct syntax for the destructors?
a) classname()
b) ()classname
c) ~classname()
d) -classname()

5. Which among the following is true?
a) First the constructor of parent classes are called in sequence of inheritance
b) First the constructor of child classes are called in the sequence of inheritance
c) First constructor called is of the object being created
d) Constructors are called randomly

6. What is the sequence of destructors call?
a) Same order as that of the constructors call

b) Random order
c) According to the priority
d) Revere of the order of constructor call
View Answer
7. The destructors _____
a) Can have maximum one argument
b) Can't have any argument
c) Can have more than one argument
d) Can't have more than 3 arguments

8. Destructor calls _____ (C++)
a) Are only implicit
b) Are only explicit
c) Can be implicit or explicit
d) Are made at end of program only
View Answer
9. Number of destructors called are _____
a) Always equal to number of constructors called
b) Always less than the number of constructors called
c) Always greater than the number of constructors called
d) Always less than or equal to number of constructors
View Answer
10. For explicit call _____
a) The destructor must be private
b) The destructor must be public
c) The destructor must be protected
d) The destructor must be defined outside the class
View Answer
11. If a class have 4 constructors then it must have 4 destructors also.
a) True
b) False
View Answer
12. Which among the following is true for destructors?
a) Destructors can be overloaded
b) Destructors can be define more than one time
c) Destructors can't be overloaded
d) Destructors are overloaded in derived classes
View Answer
13. The constructor _____
a) Have a return type
b) May have a return type
c) Of derived classes have return type
d) Doesn't have a return type
View Answer
14. The destructors _____
a) Have a return type
b) May have a return type

c) Of derived classes have return type
d) Doesn’t have a return type
View Answer
15. The destructor can be called before the constructor if required.
a) True
b) False

1. Which among the following describes a destructor?
a) A special function that is called to free the resources, acquired by the object
b) A special function that is called to delete the class
c) A special function that is called anytime to delete an object
d) A special function that is called to delete all the objects of a class
View Answer
2. When a destructor is called?
a) After the end of object life
b) Anytime in between object’s lifespan
c) At end of whole program
d) Just before the end of object life
View Answer
3. Which among the following is correct for abstract class destructors?
a) It doesn’t have destructors
b) It has destructors
c) It may or may not have destructors
d) It contains an implicit destructor

4. If in multiple inheritance, class C inherits class B, and Class B inherits class A. In which sequence are their destructors called if an object of class C was declared?
a) ~C() then ~B() then ~A()
b) ~B() then ~C() then ~A()
c) ~A() then ~B() then ~C()
d) ~C() then ~A() then ~B()

5. Choose the correct sequence of destructors being called for the following code.

```
class A{   };
class B{   };
class C: public A, public B{   };
```

a) ~A(), ~B(), ~C()
b) ~B(), ~C(), ~A()
c) ~A(), ~C(), ~B()
d) ~C(), ~B(), ~A()

6. When is the destructor of a global object called?
a) Just before end of program
b) Just after end of program
c) With the end of program
d) Anytime when object is not needed
View Answer

7. How the constructors and destructors can be differentiated?
a) Destructor have a return type but constructor doesn't
b) Destructors can't be defined by the programmer, but constructors can be defined
c) Destructors are preceded with a tilde (~) symbol, and constructor doesn't
d) Destructors are same as constructors in syntax
View Answer
8. Destructors doesn't accept parameters.
a) True
b) False
View Answer
9. Destructors can be _____
a) Abstract type
b) Virtual
c) Void
d) Any type depending on situation
View Answer
10. Global destructors execute in _____ order after main function is terminated.
a) Sequential
b) Random
c) Reverse
d) Depending on priority
View Answer
11. When is it advised to have user defined destructor?
a) When class contains some pointer to memory allocated in class
b) When a class contains static variables
c) When a class contains static functions
d) When a class is inheriting another class only
View Answer
12. Which among the following is correct for the destructors concept?
a) Destructors can be overloaded
b) Destructors can have only one parameter at maximum
c) Destructors are always called after object goes out of scope
d) There can be only one destructor in a class
View Answer
13. Which class destructor will be called first, when following code go out of scope?

```
class A{  };
class B{  };
class C: public B{  };
A a;
B b;
C c;
```

a) ~A()
b) ~B()
c) ~C()
d) ~B() and ~C()
View Answer

14. When an object is passed to a function, its copy is made in the function and then _____
a) The destructor of the copy is called when function is returned
b) The destructor is never called in this case
c) The destructor is called but it is always implicit
d) The destructor must be user defined
View Answer
15. What happens when an object is passed by reference?
a) Destructor is not called
b) Destructor is called at end of function
c) Destructor is called when function is out of scope
d) Destructor is called when called explicitly

OOP MCQ on Member Functions & its Types

1. Virtual function is _____ class function which expected to be redefined in _____ class, so that when reference is made to derived class object using pointer then we can call virtual function to execute _____ class definition version.
a) Base, derived, derived
b) Derived, Derived, Derived
c) Base, derived, base
d) Base, base, derived
View Answer
2. What does a virtual function ensure for an object, among the following?
a) Correct method is called, regardless of the class defining it
b) Correct method is called, regardless of the object being called
c) Correct method is called, regardless of the type of reference used for function call
d) Correct method is called, regardless of the type of function being called by objects
View Answer
3. Virtual functions are mainly used to achieve _____
a) Compile time polymorphism
b) Interpreter polymorphism
c) Runtime polymorphism
d) Functions code polymorphism

4. Which keyword is used to declare virtual functions?
a) virtual
b) virt
c) anonymous
d) virtually
View Answer
5. Where the virtual function should be defined?
a) Twice in base class
b) Derived class
c) Base class and derived class
d) Base class

6. The resolving of virtual functions is done at _____
a) Compile time
b) Interpret time
c) Runtime
d) Writing source code
View Answer
7. In which access specifier should a virtual function be defined?
a) Private
b) Public
c) Protected
d) Default
View Answer
8. Virtual functions can never be made _____
a) Static function
b) Parameterized function
c) Default argument function
d) Zero parameter function
View Answer
9. Virtual functions can't be made friend function of other classes.
a) True
b) False
View Answer
10. Which is a must condition for virtual function to achieve runtime polymorphism?
a) Virtual function must be accessed with direct name
b) Virtual functions must be accessed using base class object
c) Virtual function must be accessed using pointer or reference
d) Virtual function must be accessed using derived class object only
View Answer
11. Which among the following is true for virtual functions?
a) Prototype must be different in base and derived class
b) Prototype must be same in base class and derived class
c) Prototype must be given only in base class
d) Prototype must have different signature in base and derived class
View Answer
12. The virtual functions must be declared and defined in _____ class and overridden in _____ class.
a) Base, base
b) Derived, derived
c) Derived, base
d) Base, derived
View Answer
13. It is _____ to redefine the virtual function in derived class.
a) Necessary
b) Not necessary
c) Not acceptable
d) Good practice
View Answer

14. Which among the following is true?
a) A class may have virtual destructor but not virtual constructor
b) A class may have virtual constructor but not virtual destructor
c) A class may have virtual constructor and virtual constructor
d) A class may have either virtual destructor or virtual constructor
View Answer
15. If virtual function of base class is redefined in derived class then _____
a) It must be declared virtual in derived class also
b) It may or may not be declared virtual in derived class
c) It can must not be declared virtual in derived class
d) It must be declared normally in derived class

1. Which among the following best defines the abstract methods?
a) Functions declared and defined in base class
b) Functions only declared in base class
c) Function which may or may not be defined in base class
d) Function which must be declared in derived class
View Answer
2. Which among the following is true?
a) The abstract functions must be only declared in derived classes
b) The abstract functions must not be defined in derived classes
c) The abstract functions must be defined in base and derived class
d) The abstract functions must be defined either in base or derived class
View Answer
3. How are abstract functions different from the abstract functions?
a) Abstract must not be defined in base class whereas virtual function can be defined
b) Either of those must be defined in base class
c) Different according to definition
d) Abstract functions are faster

4. Which among the following is correct?
a) Abstract functions should not be defined in all the derived classes
b) Abstract functions should be defined only in one derived class
c) Abstract functions must be defined in base class
d) Abstract functions must be defined in all the derived classes
View Answer
5. It is _____ to define the abstract functions.
a) Mandatory for all the classes in program
b) Necessary for all the base classes
c) Necessary for all the derived classes
d) Not mandatory for all the derived classes

6. The abstract function definitions in derived classes is enforced at _____
a) Runtime
b) Compile time
c) Writing code time

d) Interpreting time
View Answer
7. What is this feature of enforcing definitions of abstract function at compile time called?
a) Static polymorphism
b) Polymorphism
c) Dynamic polymorphism
d) Static or dynamic according to need
View Answer
8. What is the syntax for using abstract method?
a) <access-modifier>abstract<return-type>method_name (parameter)
b) abs<return-type>method name (parameter)
c) <access-modifier>abstract return-type method name (parameter)
d) <access-modifier>abstract <returning> method name (parameter)
View Answer
9. If a function declared as abstract in base class doesn't have to be defined in derived class then _____
a) Derived class must define the function anyhow
b) Derived class should be made abstract class
c) Derived class should not derive from that base class
d) Derived class should not use that function
View Answer
10. Static methods can't be made abstract in java.
a) True
b) False
View Answer
11. Which among the following is true?
a) Abstract methods can be static
b) Abstract methods can be defined in derived class
c) Abstract methods must not be static
d) Abstract methods can be made static in derived class
View Answer
12. Which among the following is correct for abstract methods?
a) It must have different prototype in the derived class
b) It must have same prototype in both base and derived class
c) It must have different signature in derived class
d) It must have same return type only
View Answer
13. If a class have all the abstract methods the class will be known as _____
a) Abstract class
b) Anonymous class
c) Base class
d) Derived class
View Answer
14. The abstract methods can never be _____ in a base class.
a) Private
b) Protected
c) Public

d) Default
View Answer
15. The abstract method definition can be made _____ in derived class.
a) Private
b) Protected
c) Public
d) Private, public, or protected

1. How many types of member functions are possible in general?
a) 2
b) 3
c) 4
d) 5
View Answer
2. Simple member functions are _____
a) Ones defined simply without any type
b) Ones defined with keyword simple
c) Ones that are implicitly provided
d) Ones which are defined in all the classes
View Answer
3. What are static member functions?
a) Functions which use only static data member but can’t be accessed directly
b) Functions which uses static and other data members
c) Functions which can be accessed outside the class with the data members
d) Functions using only static data and can be accessed directly in main() function

4. How can static member function can be accessed directly in main() function?
a) Dot operator
b) Colon
c) Scope resolution operator
d) Arrow operator
View Answer
5. Correct syntax to access the static member functions from the main() function is _____
a) classObject::functionName();
b) className::functionName();
c) className:classObject:functionName();
d) className.classObject:functionName();

6. What are const member functions?
a) Functions in which none of the data members can be changed in a program
b) Functions in which only static members can be changed
c) Functions which treat all the data members as constant and doesn’t allow changes
d) Functions which can change only the static members
View Answer
7. Which among the following best describes the inline member functions?
a) Functions defined inside the class only

b) Functions with keyword inline only
c) Functions defined outside the class
d) Functions defined inside the class or with the keyword inline
View Answer
8. What are friend member functions (C++)?
a) Member function which can access all the members of a class
b) Member function which can modify any data of a class
c) Member function which doesn't have access to private members
d) Non-member functions which have access to all the members (including private) of a class
View Answer
9. What is the syntax of a const member function?
a) void fun() const {}
b) void fun() constant {}
c) void const fun() {}
d) const void fun(){}
View Answer
10. Which keyword is used to make a nonmember function as friend function of a class?
a) friendly
b) new
c) friend
d) connect
View Answer
11. Member functions _____
a) Must be defined inside class body
b) Can be defined inside class body or outside
c) Must be defined outside the class body
d) Can be defined in another class
View Answer
12. All type of member functions can't be used inside a single class.
a) True
b) False
View Answer
13. Which among the following is true?
a) Member functions can never be private
b) Member functions can never be protected
c) Member functions can never be public
d) Member functions can be defined in any access specifier
View Answer
14. Which keyword is used to define the static member functions?
a) static
b) stop
c) open
d) state
View Answer
15. Which keyword is used to define the inline member function?
a) no keyword required
b) inline

c) inlined
d) line

1. Which among the following are valid ways of overloading the operators?
a) Only using friend function
b) Only using member function
c) Either member functions or friend functions can be used
d) Operators can’t be overloaded
View Answer
2. Which among the following is mandatory condition for operators overloading?
a) Overloaded operator must be member function of the left operand
b) Overloaded operator must be member function of the right operand
c) Overloaded operator must be member function of either left or right operand
d) Overloaded operator must not be dependent on the operands
View Answer
3. When the operator to be overloaded becomes the left operand member then _____
a) The right operand acts as implicit object represented by *this
b) The left operand acts as implicit object represented by *this
c) Either right or left operand acts as implicit object represented by *this
d) *this pointer is not applicable in that member function

4. If the left operand is pointed by *this pointer, what happens to other operands?
a) Other operands are passed as function return type
b) Other operands are passed to compiler implicitly
c) Other operands must be passed using another member function
d) Other operands are passed as function arguments
View Answer
5. If a friend overloaded operator have to be changed to member overloaded operator, which operator should be used with the class name?
a) Scope resolution operator
b) Colon
c) Arrow operator
d) Dot operator
View Answer

6. What is the syntax to overload an operator?
a) className::operator<operatorSymbol>(parameters)
b) className:operator<operatorSymbol>(parameters)
c) className.operator<operatorSymbol>(paramteres)
d) className->operator<operatorSymbol>(parameters)
View Answer
7. Why the left parameter is removed from parameter list?
a) Because it is of no use
b) Because it is never used in definitions
c) Because it becomes parameter pointed by *this

d) Because it can’t be referred by *this pointer
View Answer
8. Which object’s members can be called directly while overloading operator function is used (In function definition)?
a) Left operand members
b) Right operand members
c) All operand members
d) None of the members
View Answer
9. If left operand member is specified directly in the function definition, which is the correct implicit conversion of that syntax?
a) *this className
b) *this parameterObject
c) *this returnedObject
d) *this object
View Answer
10. When the friend operator overloading is converted into member operator overloading _____
a) Two parameters of friend function remains same parameters in member operator overloading
b) Two parameters of friend function becomes only one parameter of member function
c) Two parameters of friend function are removed while using member function
d) Two parameters of friend function are made 4 in member operator overloading
View Answer
11. Where in the parameter list is the implicit *this is added?
a) Right most parameter
b) Anywhere in parameter list
c) Left most parameter
d) Not added to parameter list
View Answer
12. Which operator among the following can be overloading using only member function?
a) Assignment operator
b) Addition operator
c) Subtraction operator
d) Multiplication and division operator
View Answer
13. Which operator among the following can be overloaded using both friend function and member function?
a) Assignment operator
b) Subscript
c) Member selection (arrow operator)
d) Modulus operator
View Answer
14. All the operators can be overloaded using the member function operator overloading.
a) True
b) False
View Answer

15. Which operator among the following must be overloaded using the friend function?
a) << operator only
b) >> operator only
c) Both << and >> operators
d) It's not mandatory to use friend function in any case

1. What does memory allocation for objects mean?
a) Actual creation and memory allocation for object members
b) Creation of member functions
c) Creation of data members for a class
d) Actual creation and data declaration for object members
View Answer
2. Where is the memory allocated for the objects?
a) HDD
b) Cache
c) RAM
d) ROM
View Answer
3. When is the memory allocated for an object?
a) At declaration of object
b) At compile time
c) When object constructor is called
d) When object is initialized to another object

4. Using new is type safe as _____
a) It require to be specified with type of data
b) It doesn't require to be specified with type of data
c) It requires the name of data
d) It allocated memory for the data
View Answer
5. Which of the following function can be used for dynamic memory allocation of objects?
a) malloc()
b) calloc()
c) create()
d) both malloc() and calloc()

6. How much memory will be allocated for an object of class given below?

```
class Test{
int mark1;
int mark2;
float avg;
char name[10];
};
```

a) 22 Bytes
b) 24 Bytes
c) 20 Bytes
d) 18 Bytes

7. Which keyword among the following can be used to declare an array of objects in java?
a) new
b) create
c) allocate
d) arr
View Answer
8. When is the memory allocated for an object gets free?
a) At termination of program
b) When object goes out of scope
c) When main function ends
d) When system restarts
View Answer
9. Which among the following keyword can be used to free the allocated memory for an object?
a) delete
b) free
c) either delete or free
d) only delete
View Answer
10. Which function is called whenever an object goes out of scope?
a) Destructor function
b) Constructor function
c) Delete function
d) Free function

11. Which operator can be used to check the size of an object?
a) sizeof(objectName)
b) size(objectName)
c) sizeofobject(objectName)
d) sizedobject(objectName)
View Answer
12. The memory allocated for an object _____
a) Can be only dynamic
b) Can be only static
c) Can be static or dynamic
d) Can’t be done using dynamic functions
View Answer
13. If an object is declared in a user defined function _____
a) Its memory is allocated in stack
b) Its memory is allocated in heap
c) Its memory is allocated in HDD
d) Its memory is allocated in cache
View Answer
14. In java _____ takes care of managing memory for objects dynamically.
a) Free collector
b) Dust collector
c) Memory manager

d) Garbage collector
View Answer
15. Which operator can be used to free the memory allocated for an object in C++?
a) Free()
b) delete
c) Unallocate
d) Collect

1. Which among the following best describes member function overriding?
a) Member functions having same name in base and derived classes
b) Member functions having same name in base class only
c) Member functions having same name in derived class only
d) Member functions having same name and different signature inside main function
View Answer
2. Which among the following is true?
a) Inheritance must not be using when overriding is used
b) Overriding can be implemented without using inheritance
c) Inheritance must be done, to use overriding are overridden
d) Inheritance is mandatory only if more than one functions
View Answer
3. Which is the correct condition for function overriding?
a) The declaration must not be same in base and derived class
b) The declaration must be exactly the same in base and derived class
c) The declaration should have at least 1 same argument in declaration of base and derived class
d) The declaration should have at least 1 different argument in declaration of base and derived class

4. Exactly same declaration in base and derived class includes_____
a) Only same name
b) Only same return type and name
c) Only same return type and argument list
d) All the same return type, name and parameter list

5. Which among function will be overridden from the function defined in derived class below:

```
class A
{
        int i;
        void show()
        {
                cout&lt;&lt;i;
        }
        void print()
        {
                cout &lt;&lt;i;
        }
};
class B
```

```
{
        int j;
        void show()
        {
                cout&lt;&lt;j;
        }
};
```

a) show()
b) print()
c) show() and print()
d) Compile time error

6. How to access the overridden method of base class from the derived class?
a) Using arrow operator
b) Using dot operator
c) Using scope resolution operator
d) Can't be accessed once overridden
View Answer

7. The functions to be overridden _____
a) Must be private in base class
b) Must not be private base class
c) Must be private in both derived and base class
d) Must not be private in both derived and base class
View Answer

8. Which language doesn't support the method overriding implicitly?
a) C++
b) C#
c) Java
d) SmallTalk
View Answer

9. In C# _____
a) Non – virtual or static methods can't be overridden
b) Non – virtual and static methods only can be overridden
c) Overriding is not allowed
d) Overriding must be implemented using C++ code only
View Answer

10. In Delphi _____
a) Method overriding is done implicitly
b) Method overriding is not supported
c) Method overriding is done with directive override
d) Method overriding is done with the directive virtually
View Answer

11. What should be used to call the base class method from the derived class if function overriding is used in Java?
a) Keyword super
b) Scope resolution
c) Dot operator

d) Function name in parenthesis
View Answer
12. In Kotlin, the function to be overridden must be _____
a) Private
b) Open
c) Closed
d) Abstract
View Answer
13. Abstract functions of a base class _____
a) Are overridden by the definition in same class
b) Are overridden by the definition in parent class
c) Are not overridden generally
d) Are overridden by the definition in derived class
View Answer
14. If virtual functions are defined in the base class then _____
a) It is not necessary for derived classes to override those functions
b) It is necessary for derived classes to override those functions
c) Those functions can never be derived
d) Those functions must be overridden by all the derived classes
View Answer
15. Which feature of OOP is exhibited by the function overriding?
a) Inheritance
b) Abstraction
c) Polymorphism
d) Encapsulation

1. What are the constant member functions?
a) Functions which doesn’t change value of calling object
b) Functions which doesn’t change value of any object inside definition
c) Functions which doesn’t allow modification of any object of class
d) Functions which doesn’t allow modification of argument objects
View Answer
2. Which keyword must be used to declare a member function as a constant member function?
a) Constant
b) Const
c) FunctionConst
d) Unchanged
View Answer
3. Which objects can call the const functions?
a) Only const objects
b) Only non-const objects
c) Both const and non-const objects
d) Neither const not non-const objects

4. Non-const functions _____
a) Can be called only from non-const object

b) Can be called only from const object
c) Can be called both by const and non-const object
d) Can’t be called with object

5. Which is the correct condition on const member functions?
a) Const member functions can’t call non-const member functions
b) Const member functions can’t call any other function
c) Const member functions can call only the functions which are neither const nor non-const
d) Const member functions can call only data members of call not member functions

6. If a const object calls a non-const member function then _____
a) Run time error may get produced
b) Compile time error may get produced
c) Either compile time or run time error is produced
d) The program can’t be compiled
View Answer
7. Can a constructor function be constant?
a) Yes, always
b) Yes, only if permissions are given
c) No, because objects are not involved
d) No, never
View Answer
8. A function can have both the const and non-const version in the same program.
a) True
b) False
View Answer
9. How is it possible to have both const and non-const version of a function?
a) Function overriding
b) Function prototyping
c) Function overloading
d) Function declaring
View Answer
10. When both the const and non-const version of functions are required?
a) Return value have to be different in const
b) Return value have to be same in const
c) Return values have to be ignored
d) Return values have to be suppressed
View Answer
11. If a function is to be made const, which is the correct syntax?
a) const functionName(parameters);
b) const returnType functionName(parameters);
c) const functionName(returnType)(Parameters);
d) const (functionName(parameters));
View Answer
12. Functions which differ in const-ness are considered _____
a) To have same signature
b) To have different signature

c) To produce compile time error
d) To produce runtime error
View Answer
13. If const version of a function when overloading is used, the function _____
a) Returns reference to object
b) Returns volatile reference
c) Returns mutable reference
d) Returns const reference
View Answer
14. Which among the following is recommended for const functions?
a) Const function use should be reduced in a program
b) Const function use should be more in a program
c) Const function use should not matter in a program
d) Const function use should be able to modify the values
View Answer
15. Use of const member function in a program _____
a) Is mandatory, always
b) Is optional, always
c) Is mandatory, if objects are used
d) Is optional, if const objects are used

1. Which is private member functions access scope?
a) Member functions which can only be used within the class
b) Member functions which can used outside the class
c) Member functions which are accessible in derived class
d) Member functions which can't be accessed inside the class
View Answer
2. Which among the following is true?
a) The private members can't be accessed by public members of the class
b) The private members can be accessed by public members of the class
c) The private members can be accessed only by the private members of the class
d) The private members can't be accessed by the protected members of the class
View Answer
3. Which member can never be accessed by inherited classes?
a) Private member function
b) Public member function
c) Protected member function
d) All can be accessed

4. Which syntax among the following shows that a member is private in a class?
a) private: functionName(parameters)
b) private(functionName(parameters))
c) private functionName(parameters)
d) private::functionName(parameters)

5. If private member functions are to be declared in C++ then _____
a) private: <all private members>
b) private <member name>
c) private(private member list)
d) private :- <private members>

6. In java, which rule must be followed?
a) Keyword private preceding list of private member's
b) Keyword private with a colon before list of private member's
c) Keyword private with arrow before each private member
d) Keyword private preceding each private member
View Answer
7. How many private member functions are allowed in a class?
a) Only 1
b) Only 7
c) Only 255
d) As many as required
View Answer
8. How to access a private member function of a class?
a) Using object of class
b) Using object pointer
c) Using address of member function
d) Using class address
View Answer
9. Private member functions _____
a) Can't be called from enclosing class
b) Can be accessed from enclosing class
c) Can be accessed only if nested class is private
d) Can be accessed only if nested class is public
View Answer
10. Which function among the following can't be accessed outside the class in java in same package?
a) public void show()
b) void show()
c) protected show()
d) static void show()
View Answer
11. If private members are to be called outside the class, which is a good alternative?
a) Call a public member function which calls private function
b) Call a private member function which calls private function
c) Call a protected member function which calls private function
d) Not possible
View Answer
12. A private function of a derived class can be accessed by the parent class.
a) True
b) False
View Answer

13. Which error will be produced if private members are accessed?
a) Can’t access private message
b) Code unreachable
c) Core dumped
d) Bad code
View Answer
14. Can main() function be made private?
a) Yes, always
b) Yes, if program doesn’t contain any classes
c) No, because main function is user defined
d) No, never
View Answer
15. If a function in java is declared private then it _____
a) Can’t access the standard output
b) Can access the standard output
c) Can’t access any output stream
d) Can access only the output streams

1. What are public member functions?
a) Functions accessible outside the class but not in derived class
b) Functions accessible outside the class directly
c) Functions accessible everywhere using object of class
d) Functions that can’t be accessed outside the class
View Answer
2. Which among the following is true for public member functions?
a) Public member functions doesn’t have a return type
b) Public member functions doesn’t have any security
c) Public member functions are declared outside the class
d) Public member functions can be called using object of class
View Answer
3. Which type of member functions get inherited in the same specifier in which the inheritance is done? (If private inheritance is used, those become private and if public used, those become public)
a) Private member functions
b) Protected member functions
c) Public member functions
d) All member functions

4. Which syntax among the following is correct for public member functions?
a) public::void functionName(parameters)
b) public void functionName(parameters)
c) public(void functionName(parameters))
d) public:-void functionName(Parameters)
View Answer
5. Which syntax is applicable to declare public member functions in C++?
a) public: <function declaration>
b) public(<function declaration>)

c) public void <function declaration>
d) public::<function declaration>

6. In java, which rule among the following is applicable?
a) Keyword public can't be preceded with all the public members
b) Keyword public must be preceded with all the public members
c) Keyword public must be post mentioned the function declaration
d) Keyword public is not mandatory
View Answer
7. How many public members are allowed in a class?
a) Only 1
b) At most 7
c) Exactly 3
d) As many as required

8. Which is not a proper way to access public members of a class?
a) Using object pointer with arrow operator
b) Using object of class in main function
c) Using object of class with arrow operator
d) Using object anywhere in the program
View Answer
9. Which call is correct for public members of a nested class?
a) Can be called from object of enclosing class
b) Can be called within enclosing class only with direct names
c) Direct names should be used for the nested classes
d) Only with help of nested class object pointer
View Answer
10. Which public function call among the following is correct outside the class, if return type is void (C++)?
a) object.void functionName(parameters);
b) object.functionName(parameters);
c) object.functionName void (parameters)
d) object.void functionName();
View Answer
11. If public members are to be restricted from getting inherited from the subclass of the class containing that function, which alternative is best?
a) Make the function private
b) Use private inheritance
c) Use public inheritance
d) Use protected inheritance
View Answer
12. A derived class object can access the public members of the base class.
a) True
b) False
View Answer
13. If a class have a public member function and is called directly in the main function then
_____

a) Undeclared function error will be produced
b) Out of memory error is given
c) Program gives warning only
d) Program shut down the computer
View Answer
14. The function main() must always be public.
a) True
b) False
View Answer
15. All the public member functions _____
a) Can’t access the private members of a class
b) Can’t access the protected members of a class
c) Can access only public members of a class
d) Can access all the member of its class

OOPs MCQ on Default Arguments vs Overloading, Upcasting and Downcasting

1. What are default arguments?
a) Arguments which are not mandatory to be passed
b) Arguments with default value that aren’t mandatory to be passed
c) Arguments which are not passed to functions
d) Arguments which always take same data value
View Answer
2. Which is the correct condition for the default arguments?
a) Those must be declared as last arguments in argument list
b) Those must be declared first in the argument list
c) Those can be defined anywhere in the argument list
d) Those are declared inside the function definition
View Answer
3. If a member function have to be made both zero argument and parameterized constructor, which among the following can be the best option?
a) Two normal and one default argument
b) At least one default argument
c) Exactly one default argument
d) Make all the arguments default

4. Which among the following function can be called without arguments?
a) void add(int x, int y=0)
b) void add(int=0)
c) void add(int x=0, int y=0)
d) void add(char c)
View Answer
5. If a function have all the default arguments but still some values are passed to the function then _____
a) The function will use the values passed to it

b) The function will use the default values as those are local
c) The function can use any value whichever is higher
d) The function will choose the minimum values

6. Which among the following is correct?
a) void test(int x=0, int y, int z=0)
b) void test(int x=0, int=0)
c) void test(int x, int y=0)
d) void test(int x='c, int y)
View Answer
7. What function will be called with the independent syntax "test(5,6,7);"?
a) void test(int x, int y)
b) void test(int x=0, int y, int z)
c) int test(int x=0, y=0, z=0)
d) void test(int x, int y, int z=0)
View Answer
8. Which among the following is a wrong call to the function void test(int x, int y=0, int z=0)?
a) test(5,6,7);
b) test(5);
c) test();
d) test(5,6);
View Answer
9. Default arguments are _____
a) Only allowed in the parameter list of the function declaration
b) Only allowed in the return type of the function declaration
c) Only allowed with the class name definition
d) Only allowed with the integer type values
View Answer
10. Which among the following is false for default arguments?
a) Those are not allowed with a declaration of pointer to functions
b) Those are not allowed with the reference to functions
c) Those are not allowed with the typedef declarations
d) Those are allowed with pointer and reference to function declaration
View Answer
11. The non-template functions can be added with default arguments to already declared functions _____
a) If and only if the function is declared again in the same scope
b) If and only if the function is declared only once in the same scope
c) If and only if the function is declared in different scope
d) If and only if the function is declared twice in the program
View Answer
12. The using declaration _____
a) Doesn't carry over the default values
b) Carries over the known default arguments
c) Carries over only the normal arguments
d) Carries over only few default arguments
View Answer

13. The names given to the default arguments are only looked up and _____ and are bound during declaration.
a) Checked for availability
b) Checked for random access
c) Checked for accessibility
d) Checked for feasibility
View Answer
14. The default argument get bound during declaration _____
a) And are never executed
b) And are executed simultaneously
c) But are executed only if priority is given
d) But are executed during function call
View Answer
15. The virtual function overrides _____
a) Do not acquire base class declaration of default arguments
b) Do acquire base class declaration of default arguments
c) Do not link with the default arguments of base class
d) Do link with the default argument but only of derived classes

1. Which among the following is true for constructors overloading?
a) Constructors can't be overloaded
b) Constructors can be overloaded using different signatures
c) Constructors can be overloaded with same signatures
d) Constructors can be overloaded with different return types
View Answer
2. If a constructors should be capable of creating objects without argument and with arguments, which is a good alternative for this purpose?
a) Use zero argument constructor
b) Use constructor with one parameter
c) Use constructor with all default arguments
d) Use default constructor
View Answer
3. The Constructors with all the default arguments are similar as default constructors.
a) True
b) False

4. Which among the following is true?
a) The constructors overloading can be done by using different names
b) The constructors overloading can be done by using different return types
c) The constructors can be overloaded by using only one argument
d) The constructors must have the same name as that of class

5. Which among the following can be used in place of default constructor?
a) constructorName(int x, int y=0)
b) constructorName(int x=0, int y=0)
c) constructorName(int x=0, int y)
d) constructorName(int x, int y)

6. Can a class have more than one function with all the default arguments?
a) Yes, always
b) Yes, if argument list is different
c) No, because constructors overloading doesn't depend on argument list
d) No, never
View Answer
7. Which is the correct syntax for using default arguments with the constructor?
a) default constructorName(default int x=0)
b) constructorName(default int x=0)
c) constructorName(int x=0)
d) constructorName()
View Answer
8. How many parameters must be passed if only the following prototype is given to a constructor?
Prototype: className(int x, int y, int z=0);
a) 1
b) 2
c) 3
d) Compile time error
View Answer
9. If the constructors are overloaded by using the default arguments, which problem may arise?
a) The constructors might have all the same arguments except the default arguments
b) The constructors might have same return type
c) The constructors might have same number of arguments
d) The constructors can't be overloaded with respect to default arguments
View Answer
10. Which among the following is true?
a) More than one constructors with all default arguments is allowed
b) More than one constructors with all default arguments can be defined outside the class
c) More than one constructors can be used with same argument list
d) More than one constructors with all default arguments can't exist in same class
View Answer
11. Which constructor among the following will be called if a call is made like className(5,'a');?
a) className(int x=5,char c='a');
b) int className(int x, char c, char d);
c) className(int x, char c, int y);
d) char className(char c,int x);
View Answer
12. Which constructor definition will produce a compile time error?
a) className(int x=0);
b) className(char c);
c) className(int x=0,char c);
d) className(char c,int x=0);
View Answer
13. If there is a constructor with all the default arguments and arguments are not passed then _____

a) The default values given will not be used
b) Then all the null values will be used
c) Then all the default values given will be used
d) Then compiler will produce an error
View Answer
14. Which is the correct statement for default constructors?
a) The constructors with all the default arguments
b) The constructors with all the null and zero values
c) The constructors which can't be defined by programmer
d) The constructors with zero arguments
View Answer
15. Which is a good alternative instead of having one zero argument constructor and one single argument constructor with default argument?
a) No constructor defined
b) One default value constructor
c) Defining the default constructor
d) Using one constructor with two arguments

1. What is upcasting?
a) Casting subtype to supertype
b) Casting super type to subtype
c) Casting subtype to super type and vice versa
d) Casting anytype to any other type
View Answer
2. Which among the following is true for upcasting in inheritance?
a) Downward to the inheritance tree
b) Upward to the inheritance tree
c) Either upward or downward
d) Doesn't apply on inheritance
View Answer
3. Which among the following is safe?
a) Upcasting
b) Downcasting
c) Both upcasting and downcasting
d) If upcasting is safe then downcasting is not, and vice versa

4. Which among the following is the best situation to use upcasting?
a) For general code dealing with only subtype
b) For general code dealing with only supertype
c) For general code dealing with both the supertype and subtype
d) For writing a rigid code with respect to subtype
View Answer
5. Which property is shown most when upcasting is used?
a) Code reusability
b) Code efficiency
c) Complex code simple syntax
d) Encapsulation

6. Upcasting and downcasting objects are the same as casting primitive types.
a) True
b) False
View Answer
7. Which casting among the following is allowed for the code given below?

```
class A
{
        public :int a;
}
class B:public A
{
        int b;
}
main()
{
        B b=new A();  //casting 1
        A a=new B();  //casting 2
}
```

a) Casting 1
b) Casting 2
c) casting 1 and casting 2
d) casting 1 nor casting 2
View Answer
8. If multiple inheritance is implemented, which upcasting will be correct?
a) Upcast to first base class listed in inheritance
b) Upcast to send base class listed in inheritance
c) Upcast to any base class
d) Upcast is not possible
View Answer
9. If class C inherits class B and class B inherits class A _____
a) Class C object can be upcasted to object of class B only
b) Class C object can be upcasted to object of class A only
c) Class C object can be upcasted to object of either class A or B
d) Class C object can’t be upcasted
View Answer
10. Upcasting is _____ without an explicit type cast.
a) Always allowed for public inheritance
b) Always allowed for protected inheritance
c) Always allowed for private inheritance
d) Not allowed
View Answer
11. Which concept is needed because of implicit type casting use?
a) Static binding
b) Dynamic binding
c) Compile time binding
d) Source code binding
View Answer

12. When are the pointer types known for upcasting the objects?
a) Compile time
b) Runtime
c) Source code build time
d) Doesn’t apply to pointer types
View Answer
13. When are the object type known for upcasting the objects?
a) Compile time
b) Runtime
c) Source code build time
d) Doesn’t apply to objects directly
View Answer
14. If two classes are defined “Parent” and “Child” then which is the correct type upcast syntax in C++?
a) Parent *p=child;
b) Parent *p=*child;
c) Parent *p=&child;
d) Parent *p=Child();
View Answer
15. Which among the following is true?
a) Upcasting is possible only for single level inheritance
b) Upcasting is possible only for multilevel inheritance
c) Upcasting is possible only for multiple inheritance
d) Upcasting is possible for any type of inheritance

1. What is downcasting?
a) Casting subtype to supertype
b) Casting supertype to subtype
c) Casting subtype to supertype and vice versa
d) Casting anytype to any other type
View Answer
2. Which among the following is a mandatory condition for downcasting?
a) It must not be done explicitly
b) It must be done implicitly
c) It must be done explicitly
d) It can’t be done explicitly
View Answer
3. Downcasting is _____
a) Always safe
b) Never safe
c) Safe sometimes
d) Safe, depending on code

4. Downcasting _____
a) Can result in unexpected results
b) Can’t result in unexpected result
c) Can result only in out of memory error

d) Can’t result in any error
View Answer
5. What should be used for safe downcast?
a) Static cast
b) Dynamic cast
c) Manual cast
d) Implicit cast

6. What does dynamic_cast return after successful type casting?
a) Address of object which is converted
b) Address of object that is used for conversion
c) Address of object that is mentioned in the syntax
d) Doesn’t return any address
View Answer
7. If dynamic_cast fails, which value is returned?
a) void
b) null
c) void pointer
d) null pointer
View Answer
8. Which is the proper syntax of dynamic_cast?
a) dynamic_cast(object)
b) dynamic_cast new (object)
c) dynamic_cast(object)
d) dynamic_cast(object)
View Answer
9. Which is the exception handler for the exceptions of downcasting?
a) CastException
b) ClassCastingExeption
c) ClassCasting
d) ClassCastException
View Answer
10. How to prevent the ClassCastExceptions?
a) By using instanceof
b) By using is-a check
c) By using arrow operator with check function
d) By checking type of conversion
View Answer
11. Java supports direct downcasting.
a) True
b) False
View Answer
12. Which way the downcasting is possible with respect to inheritance?
a) Upward the inheritance order
b) Downward the inheritance order
c) Either upward or downward the inheritance order

d) Order of inheritance doesn’t matter
View Answer
13. What happens when downcasting is done but not explicitly defined in syntax?
a) Compile time error
b) Runtime error
c) Code write time error
d) Conversion error
View Answer
14. When is the downcasting used?
a) To separate inherited class from base class
b) To write a more complex code
c) To compare two objects
d) To disable one class in inheritance
View Answer
15. Why is downcasting possible in any language?
a) Because inheritance follows has-a relationship
b) Because inheritance follows is-a relationship
c) Because inheritance doesn’t follow any relationship
d) Because inheritance is not involved in casting

OOPs MCQ on Memory Allocation & Scope of Variable

1. What is the new operator?
a) Allocates memory for an object or array
b) Allocates memory for an object or array and returns a particular pointer
c) Used as return type when an object is created
d) Used to declare any new thing in a program
View Answer
2. Microsoft C++ Components extensions support new keyword to _____
a) Modify a vtable
b) Replace a vtable slot entry
c) Add new vtable slot entries
d) Rearrange vtable slot entries
View Answer
3. What happens when new fails?
a) Returns zero always
b) Throws an exception always
c) Either throws an exception or returns zero
d) Terminates the program

4. If new throws an error, which function can be called to write a custom exception handler?
a) _set_handler
b) _new_handler
c) _handler_setter
d) _set_new_handler
View Answer

5. In C++, if new operator is used, when is the constructor called?
a) Before the allocation of memory
b) After the allocation of memory
c) Constructor is called to allocate memory
d) Depends on code

6. Which among the following is correct syntax to declare a 2D array using new operator?
a) char (*pchar)[10] = new char[][10];
b) char (pchar) = new char[][10];
c) char (*char) = new char[10][];
d) char (*char)[][10]= new char;
View Answer
7. For declaring data by using new operator _____
a) Type name can't contain const
b) Type name can't contain volatile
c) Type name can't contain class declarations
d) Type name can't contain const, volatile, class declaration or enumerations
View Answer
8. The new operator _____
a) Can allocate reference types too
b) Doesn't allocate reference types
c) Can allocate reference to objects
d) Doesn't allocate any data
View Answer
9. Which among the following is true?
a) New operator can't allocate functions but pointer to functions can be allocated
b) New operator can allocate functions as well as pointer to functions
c) New operator can allocate any type of functions
d) New operator is not applicable with functions allocation
View Answer
10. Which among the following is added in grammar of new operator?
a) Finalize
b) Arg
c) Initializer
d) Allocator
View Answer
11. Initializers _____
a) Are used for specifying arrays
b) Are used to defined multidimensional arrays
c) Can't be specified for arrays
d) Can't be specified for any data
View Answer
12. The objects allocated using new operator _____
a) Are destroyed when they go out of scope
b) Are not destroyed even if they go out of scope
c) Are destroyed anytime

d) Are not destroyed throughout the program execution
View Answer
13. The new operator _____
a) Invokes function operator new
b) Doesn't invoke function operator new
c) Invokes function operator only if required
d) Can't invoke function operator new implicitly
View Answer
14. If a new operator is defined for a class and still global new operator have to be used, which operator should be used with the keyword new?
a) Colon
b) Arrow
c) Dot
d) Scope resolution
View Answer
15. How does compiler convert "::operator new" implicitly?
a) ::operator new(sizeof(type))
b) ::operator new(sizeof( ))
c) new operator :: type sizeof(type)
d) new sizeof(type) operator

1. What is a delete operator?
a) Deallocates a block of memory
b) Deallocates whole program memory
c) Deallocates only primitive data memory
d) Deallocates all the data reserved for a class
View Answer
2. If an object is allocated using new operator _____
a) It should be deleted using delete operator
b) It can't be deleted using delete operator
c) It may or may not be deleted using delete operator
d) The delete operator is not applicable
View Answer
3. Does delete return any value?
a) Yes, positive value
b) Yes, negative value
c) Yes, zero value
d) No

4. Which type of value has resulted from the delete operator?
a) void
b) void pointer
c) null pointer
d) null
View Answer
5. If delete is used to delete an object which was not allocated using new
_____

a) Then out of memory error arises
b) Then unreachable code error arises
c) Then unpredictable errors may arise
d) Then undefined variable error arises

6. Delete operator _____
a) Can be used on pointers with null value
b) Can be used on pointers with void value
c) Can be used on pointer with value 0
d) Can be used on pointer with any value
View Answer
7. When delete operator is used _____ (If object has a destructor)
a) Object destructor is called after deallocation
b) Object destructor is called before deallocation
c) Object destructor is not used
d) Object destructor can be called anytime during destruction
View Answer
8. If delete is applied to an object whose l-value is modifiable, then _____ after the object is deleted.
a) Its value is defined as null
b) Its value is defined as void
c) Its value is defined as 0
d) Its value is undefined
View Answer
9. How many variants of delete operator are available?
a) Only 1
b) Only 2
c) Only 3
d) Only 4
View Answer
10. Which is the correct syntax to delete a single object?
a) delete *objectName;
b) objectName delete;
c) delete objectName;
d) objectName *delete;
View Answer
11. Which is the correct syntax to delete an array of objects?
a) delete [] objectName;
b) delete * objectName;
c) objectName[] delete;
d) delete objectName[];
View Answer
12. Which cases among the following produces the undefined result?
a) delete [] on an independent object
b) delete on an object array
c) delete [] on an object and delete on object array

d) Undefined result is never produced
View Answer
13. The delete operator _____
a) Invokes function operator delete
b) Invokes function defined by user to delete
c) Invokes function defined in global scope to delete object
d) Doesn’t invoke any function
View Answer
14. For objects that are not of class type _____
a) Global delete operator is invoked
b) Local delete operator is invoked
c) Global user defined function is invoked
d) Local function to delete object is called

15. The delete operator _____
a) Can be defined for each class
b) Can’t be defined for each class
c) Can be defined globally only
d) Can’t be defined in a program explicitly

1. What are automatic variables?
a) Global variables
b) Implicit/temporary variables
c) Local variables
d) System variables
View Answer
2. The memory for automatic variables _____
a) Have to be allocated and deallocated explicitly
b) Are allocated and deallocated automatically
c) Is never actually allocated
d) Are never safe
View Answer
3. Scope of an automatic variable _____
a) Is actually the whole program
b) Is actually never fixed
c) Is always equal to the whole program execution
d) Is actually function or block in which it is defined

4. Which among the following is true for automatic variables in general?
a) Automatic variables are invisible to called function
b) Automatic variables are always visible to the called function
c) Automatic variables can’t interact with the called function
d) Automatic variables can’t be variable
View Answer
5. If an automatic variable is created and then a function is called then _____
a) The automatic variable created gets destroyed
b) The automatic variable doesn’t get destroyed

c) The automatic variable may or may not get destroyed
d) The automatic variable can’t be used in this case

6. Where are the automatic variables stored if another function is called in between the execution of the program?
a) Heap
b) Queue
c) Stack
d) Temp variable
View Answer
7. The static variables of a function _____
a) Are also automatic variables
b) Are not automatic variables
c) Are made automatic by default
d) Can be made automatic explicitly
View Answer
8. All variables declared within a block _____
a) Are not always automatic
b) Can be made non-automatic
c) Are static by default
d) Are automatic by default
View Answer
9. What values does uninitialized automatic variables contain?
a) Null value
b) Void value
c) Undefined/Garbage
d) Zero value
View Answer
10. Constructor of automatic variables is called _____
a) When execution reaches the place of declaration of automatic variables
b) When the program is compiled
c) When the execution is just started
d) Just before the execution of the program
View Answer
11. Does java contain auto or register keywords?
a) Yes, for declaring every type of variable
b) Yes, only to declare cache registers
c) No, because java doesn’t support automatic variables
d) No, java supports local variable concept
View Answer
12. The automatic variables _____
a) Must be declared after its use
b) Must be declared before using
c) Must be declared, can be anytime
d) Must not be initialized
View Answer

13. Which error is produced if the automatic variables are used without declaration?
a) Undefined symbol
b) Memory error
c) Type mismatch
d) Statement missing
View Answer
14. In Perl, using which operator are the local variables created?
a) Dot
b) Arrow
c) Scope resolution
d) my
View Answer
15. How are automatic variables different from the instance variables?
a) Automatic variables are initialized automatically but instances are not
b) Automatic variables are given zero values initially and not instances
c) Instance variables have to be initialized explicitly and automatic implicitly
d) Instance variables are initialized implicitly while automatic are not

1. What is extern variable?
a) Variables to be used that are declared in another object file
b) Variables to be used that are declared in another source file
c) Variables to be used that are declared in another executable file
d) Variables to be used that are declared in another program
View Answer
2. Which among the following is a correct statement for variables?
a) Variable can be declared many times
b) Variable can be declared only one time
c) Variable declaration can’t be done more than ones
d) Variable declaration is always done more than one time
View Answer
3. Which among the following is true for the variables?
a) Variable can be defined only once
b) Variable can be defined any number of times
c) Variable must be defined more than one time
d) Variable can be defined in different files

4. To use extern variable _____
a) The source file must not be included in the new file code
b) The source file itself must be used for a new program
c) The source file must be included in the new file
d) The source file doesn’t matter for extern variables
View Answer
5. What does a header file contain for an extern variable?
a) Only declaration of variables
b) Only definition of variables
c) Both declaration and definition of variables
d) Neither declaration nor definition

6. Which condition is true if the extern variable is used in a file?
a) All the header files declare it
b) Only few required files declare it
c) All header files declared it if required
d) Only one header file should declare it
View Answer
7. Whenever a function is declared in a program _____
a) extern can be used only in some special cases
b) extern can't be used
c) function is extern by default
d) it can't be made extern
View Answer
8. Even if a variable is not declared as extern, it is extern by default.
a) True
b) False
View Answer
9. Which of the following results in the allocation of memory for the extern variables?
a) Declaration
b) Definition
c) Including file
d) Memory is not allocated for extern variables
View Answer
10. Which is the correct syntax for extern variable declaration?
a) extern data_type variable_name;
b) extern variable_name;
c) data_type variable_name extern;
d) extern (data_type)variable_name;
View Answer
11. Which is the correct syntax for extern function declaration?
a) extern function_name(argument_list);
b) extern return_type function_name(argument_list);
c) extern (return_type)function_name(argument_list);
d) return_type extern function_name(argument_list);
View Answer
12. What will be the output of the program?

```
extern int var;
int main(void)
{
  var = 10;
  var++;
  cout<<var;
}
```

a) 10
b) 11
c) Run time error
d) Compile time error
View Answer

13. If the definition is given in the header file that we include then _____
a) The program can run successfully
b) Also the program should define the extern variable
c) The extern variable must contain two definitions
d) Extern variable can’t be used in the program
View Answer
14. If extern variable is initialized with the declaration then _____
a) Also the header file with definition is required
b) The header file with definition must be included
c) There is no need to include any other header file for definition
d) The extern variable produces compile time error
View Answer
15. Why are functions extern by default?
a) Because functions are always private
b) Because those are not visible throughout the program
c) Because those can’t be accessed in all parts of the program
d) Because those are visible throughout the program