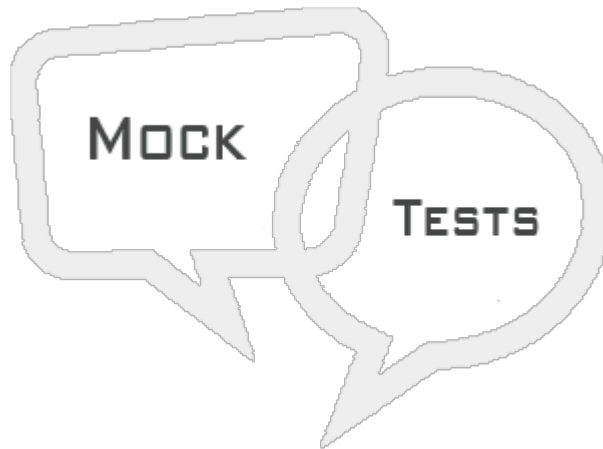


C++ MOCK TEST

<http://www.tutorialspoint.com>

Copyright © tutorialspoint.com

This section presents you various set of Mock Tests related to **C++ Framework**. You can download these sample mock tests at your local machine and solve offline at your convenience. Every mock test is supplied with a mock test key to let you verify the final score and grade yourself.



C++ MOCK TEST II

Q 1 - What is the output of the following program?

```
#include<iostream>

using namespace std;
main() {

    int i = 1, j = 2, k = 3, r;

    r = (i, j, k);

    cout<<r<<endl;

}
```

- A - 1
- B - 2
- C - 3
- D - Compile Error

Q 2 - In the following program f is overloaded.

```
void f(int x) {

}

void f(signed x) {

}

main() {

}
```

- A - True

B - False

Q 3 - In the following program f is overloaded.

```
void f(int x) {  
}  
  
int f(signed x) {  
    return 1;  
}  
  
main() {  
}
```

A - True

B - False

Q 4 - A protected member of the class is accessible in

A - Only same class

B - Same class and derived class

C - Outside the class

D - None of the above.

Q 5 - Runtime polymorphism is done using.

A - Function overloading

B - Virtual classes

C - Virtual functions

D - Friend function

Q 6 - Choose the Object oriented programming language from below.

A - C++

B - Small talk

C - Simula

D - All the above.

Q 7 - Class function which is called automatically as soon as the object is created is called as __

A - Constructor

B - Destructor

- C - Friend function
- D - Inline function.

Q 8 - Escape sequence character '\0' occupies __ amount of memory.

- A - 0
- B - 1
- C - 2
- D - 4

Q 9 - How can we make an class act as an interface in C++?

- A - By only providing all the functions as virtual functions in the class.
- B - Defining the class following with the keyword virtual
- C - Defining the class following with the keyword interface
- D - Defining the class following with the keyword abstract

Q 10 - The pointer which stores always the current active object address is __

- A - auto_ptr
- B - this
- C - p
- D - none of the above.

Q 11 - We can use this pointer in static member function of the class.

- A - True
- B - False

Q 12 - Designer of C++ programming language.

- A - Charles Babbage
- B - Dennis Ritchie
- C - Brain Kernighan
- D - Bjarne Stroustrup

Q 13 - How many number of arguments can a destructor of a class receives?

- A - 0
- B - 1
- C - 2

D - None of the above.

Q 14 - What is the output of the following program?

```
#include<iostream>

using namespace std;
class Base {
public:
    virtual void f() {
        cout<<"Base\n";
    }
};
class Derived:public Base {
public:
    void f() {
        cout<<"Derived\n";
    }
};

main() {
    Base *p = new Derived();
    p->f();
}
```

A - Base

B - Derived

C - Compile error

D - None of the above.

Q 15 - What is the output of the following program?

```
#include<iostream>

using namespace std;
class Base {
public:
    void f() {
        cout<<"Base\n";
    }
};
class Derived:public Base {
public:
    f() {
        cout<<"Derived\n";
    }
};

main() {
    Base *p = new Derived();

    p->f();
}
```

A - Base

B - Derived

C - Compile error

D - None of the above.

Q 16 - What is the output of the following program?

```
#include<iostream>

using namespace std;
class Base {
public:
    void f() {
        cout<<"Base\n";
    }
};
class Derived:public Base {
public:
    void f() {
        cout<<"Derived\n";
    }
};
main() {
    Derived obj;
    obj.Base::f();
}
```

A - Base

B - Derived

C - Compile error

D - None of the above.

Q 17 - What is the output of the following program?

```
#include<iostream>

using namespace std;
class Base {
public:
    void f() {
        cout<<"Base\n";
    }
};
class Derived:public Base {
public:
    void f() {
        cout<<"Derived\n";
    };
};
main() {
    Derived obj;
    obj.Base::f();
}
```

A - Base

B - Derived

C - Compile error

D - None of the above.

Q 18 - What is the output of the following program?

```
#include<iostream>

using namespace std;
main() {
    int *p = new int;
    delete p;
    delete p;
    cout<<"Done";
}
```

- A - Done
- B - Compile error
- C - Runtime error
- D - None of the above

Q 19 - What is the output of the following program?

```
#include<iostream>

using namespace std;
main() {
    int const a = 5;

    a++;
    cout<<a;
}
```

- A - 5
- B - 6
- C - Runtime error
- D - Compile error

Q 20 - Which operator is used to resolve the scope of the global variable?

- A - -->
- B - .
- C - *
- D - ::

Q 21 - Which feature of the OOPS gives the concept of reusability?

- A - Abstraction
- B - Encapsulation
- C - Inheritance
- D - None of the above.

Q 22 - Objects created using new operator are stored in __ memory.

- A - Cache
- B - Heap
- C - Stack
- D - None of the above.

Q 23 - What is the full form of RTTI.

- A - Runtime type identification
- B - Runtime template identification
- C - Robust Template Type Inheritance
- D - None of the above.

Q 24 - The programs machine instructions are store in __ memory segment.

- A - Data
- B - Stack
- C - Heap
- D - Code

Q 25 - The copy constructor is executed on

- A - Assigned one object to another object at its creation
- B - When objects are sent to function using call by value mechanism
- C - When the function return an object
- D - All the above.

ANSWER SHEET

Question Number	Answer Key
-----------------	------------

- | | |
|---|---|
| 1 | C |
| 2 | B |
| 3 | B |
| 4 | B |
| 5 | C |
| 6 | D |
| 7 | A |
| 8 | B |

9	A
10	B
11	B
12	D
13	A
14	B
15	A
16	A
17	C
18	C
19	D
20	D
21	C
22	B
23	A
24	D
25	D

Loading [MathJax]/jax/output/HTML-CSS/jax.js