C++ Constructors Job Interview Questions And Answers

Interview Questions Answers

https://interviewquestionsanswers.org/

About Interview Questions Answers

Interview Questions Answers . ORG is an interview preparation guide of thousands of Job Interview Questions And Answers, Job Interviews are always stressful even for job seekers who have gone on countless interviews. The best way to reduce the stress is to be prepared for your job interview. Take the time to review the standard interview questions you will most likely be asked. These interview questions and answers on C++ Constructors will help you strengthen your technical skills, prepare for the interviews and quickly revise the concepts.

If you find any **question or answer** is incorrect or incomplete then you can **submit your question or answer** directly with out any registration or login at our website. You just need to visit C++ Constructors Interview Questions And Answers to add your answer click on the *Submit Your Answer* links on the website; with each question to post your answer, if you want to ask any question then you will have a link *Submit Your Question*; that's will add your question in C++ Constructors category. To ensure quality, each submission is checked by our team, before it becomes live. This C++ Constructors Interview preparation PDF was generated at **Wednesday 29th November, 2023**

You can follow us on FaceBook for latest Jobs, Updates and other interviews material.
www.facebook.com/InterviewQuestionsAnswers.Org

Follow us on Twitter for latest Jobs and interview preparation guides.
https://twitter.com/InterviewQA

If you need any further assistance or have queries regarding this document or its material or any of other inquiry, please do not hesitate to contact us.

Best Of Luck.

Interview Questions Answers.ORG Team
https://InterviewQuestionsAnswers.ORG/
Support@InterviewQuestionsAnswers.ORG

C++ Constructors Interview Questions And Answers Guide.

Question - 1:

Which of the following statement is correct?
A. Destructor destroys only integer data members of the object.
B. Destructor destroys only float data members of the object.
C. Destructor destroys only pointer data members of the object.
D. Destructor destroys the complete object.

Ans:

Option D
Destructor destroys the complete object.

View All Answers

Question - 2:

How many default constructors per class are possible?
A. Only one
B. Two
C. Three
D. Unlimited

Ans:

Option A
Only one

View All Answers

Question - 3:

Which of the following statement is correct about destructors?
A. A destructor has void return type.
B. A destructor has integer return type.
C. A destructor has no return type.
D. A destructors return type is always same as that of main().

Ans:

Option C
A destructor has no return type.

View All Answers

Question - 4:

Which of the following cannot be declared as virtual?
A. Constructor
B. Destructor
C. Data Members
D. Both A and C

Ans:

Option D
Both A and C

View All Answers

Question - 5:

Which of the following are NOT provided by the compiler by default?
A. Zero-argument Constructor
B. Destructor
C. Copy Constructor
D. Copy Destructor

Ans:

Option D
Copy Destructor

View All Answers

Question - 6:

If the copy constructor receives its arguments by value, the copy constructor would
A. call one-argument constructor of the class
B. work without any problem
C. call itself recursively
D. call zero-argument constructor

Ans:

Option C
call itself recursively

View All Answers

Question - 7:

Which of the following statement is correct?
A. A destructor has the same name as the class in which it is present.
B. A destructor has a different name than the class in which it is present.
C. A destructor always returns an integer.
D. A destructor can be overloaded.

Ans:

Option A
A destructor has the same name as the class in which it is present.

View All Answers

Question - 8:

Constructors __________ to allow different approaches of object construction.
A. cannot overloaded
B. can be overloaded
C. can be called
D. can be nested

Ans:

Option B
can be overloaded

View All Answers

Question - 9:

A __________ is a constructor that either has no parameters, or if it has parameters, all the parameters have default values.
A. default constructor
B. copy constructor
C. Both A and B
D. None of these

Ans:

Option A
default constructor

View All Answers

Question - 10:

If the programmer does not explicitly provide a destructor, then which of the following creates an empty destructor?
A. Preprocessor
B. Compiler
C. Linker
D. main() function

Ans:

Option B
Compiler

View All Answers

Question - 11:

It is a __________ error to pass arguments to a destructor.
A. logical
B. virtual
C. syntax
D. linker

Ans:

Option C

syntax
View All Answers

Question - 12:
__________ used to make a copy of one class object from another class object of the same class type.
A. constructor
B. copy constructor
C. destructor
D. default constructor

Ans:
Option B
copy constructor
View All Answers

Question - 13:
A union that has no constructor can be initialized with another union of __________ type.
A. different
B. same
C. virtual
D. class

Ans:
Option B
same
View All Answers

Question - 14:
A function with the same name as the class, but preceded with a tilde character (~) is called __________ of that class.
A. constructor
B. destructor
C. function
D. object

Ans:
Option B
destructor
View All Answers

Question - 15:
Which of the following never requires any arguments?
A. Member function
B. Friend function
C. Default constructor
D. const function

Ans:
Option C
Default constructor
View All Answers

Question - 16:
Which of the following gets called when an object goes out of scope?
A. constructor
B. destructor
C. main
D. virtual function

Ans:
Option B
destructor
View All Answers

Question - 17:
Destructor calls are made in which order of the corresponding constructor calls?
A. Reverse order
B. Forward order
C. Depends on how the object is constructed
D. Depends on how many objects are constructed

Ans:
Option A
Reverse order
View All Answers

Question - 18:

A destructor takes __________ arguments.
A. one
B. two
C. three
D. no

Ans:

Option D
no

View All Answers

Question - 19:

Which of the following implicitly creates a default constructor when the programmer does not explicitly define at least one constructor for a class?
A. Preprocessor
B. Linker
C. Loader
D. Compiler

Ans:

Option D
Compiler

View All Answers

Question - 20:

Which of the following statement is correct?
A. A constructor has the same name as the class in which it is present.
B. A constructor has a different name than the class in which it is present.
C. A constructor always returns an integer.
D. A constructor cannot be overloaded.

Ans:

Option A
A constructor has the same name as the class in which it is present.

View All Answers

Question - 21:

Copy constructor must receive its arguments by __________ .
A. either pass-by-value or pass-by-reference
B. only pass-by-value
C. only pass-by-reference
D. only pass by address

Ans:

Option C
only pass-by-reference

View All Answers

Question - 22:

When are the Global objects destroyed?
A. When the control comes out of the block in which they are being used.
B. When the program terminates.
C. When the control comes out of the function in which they are being used.
D. As soon as local objects die.

Ans:

Option B
When the program terminates.

View All Answers

Question - 23:

Which of the following statement is incorrect?
A. Constructor is a member function of the class.
B. The compiler always provides a zero argument constructor.
C. It is necessary that a constructor in a class should always be public.
D. Both B and C.

Ans:

Option D
Both B and C.

View All Answers

Question - 24:

Which of the following statement is correct?

A. Constructor has the same name as that of the class.
B. Destructor has the same name as that of the class with a tilde symbol at the beginning.
C. Both A and B.
D. Destructor has the same name as the first member function of the class.

Ans:

Option C
Both A and B.

View All Answers

Question - 25:

Which constructor function is designed to copy objects of the same class type?
A. Create constructor
B. Object constructor
C. Dynamic constructor
D. Copy constructor

Ans:

Option D
Copy constructor

View All Answers

Question - 26:

How many times a constructor is called in the life-time of an object?
A. Only once
B. Twice
C. Thrice
D. Depends on the way of creation of object

Ans:

Option A
Only once

View All Answers

Question - 27:

Which of the following statements are correct?
A. Constructor is always called explicitly.
B. Constructor is called either implicitly or explicitly, whereas destructor is always called implicitly.
C. Destructor is always called explicitly.
D. Constructor and destructor functions are not called at all as they are always inline.

Ans:

Option B
Constructor is called either implicitly or explicitly, whereas destructor is always called implicitly.

View All Answers

Question - 28:

Which of the following statement is correct?
A. A constructor of a derived class can access any public and protected member of the base class.
B. Constructor cannot be inherited but the derived class can call them.
C. A constructor of a derived class cannot access any public and protected member of the base class.
D. Both A and B.

Ans:

Option D
Both A and B.

View All Answers

Question - 29:

Destructors __________ for automatic objects if the program terminates with a call to function exit or function abort.
A. are called
B. are inherited
C. are not called
D. are created

Ans:

Option C
are not called

View All Answers

Question - 30:

A class's __________ is called when an object is destroyed.
A. constructor
B. destructor
C. assignment function

D. copy constructor

Ans:

Option B
destructor

View All Answers

Question - 31:

For automatic objects, constructors and destructors are called each time the objects
A. enter and leave scope
B. inherit parent class
C. are constructed
D. are destroyed

Ans:

Option A
enter and leave scope

View All Answers

Question - 32:

Destructor has the same name as the constructor and it is preceded by ______ .
A. !
B. ?
C. ~
D. $

Ans:

Option C
(~)

View All Answers

Question - 33:

Can a class have virtual destructor?
A. Yes
B. No

Ans:

Option A
Yes

View All Answers

Question - 34:

What happens when a class with parameterized constructors and having no default constructor is used in a program and we create an object that needs a zero-argument constructor?
A. Compile-time error.
B. Preprocessing error.
C. Runtime error.
D. Runtime exception.

Ans:

Option A
Compile-time error.

View All Answers

Question - 35:

A constructor that accepts __________ parameters is called the default constructor.
A. one
B. two
C. no
D. three

Ans:

Option C
no

View All Answers

Question - 36:

Which of the following statement is correct whenever an object goes out of scope?
A. The default constructor of the object is called.
B. The parameterized destructor is called.
C. The default destructor of the object is called.
D. None of the above.

Ans:

Option C

The default destructor of the object is called.

View All Answers

Question - 37:

Which of the following statement is correct about constructors?
A. A constructor has a return type.
B. A constructor cannot contain a function call.
C. A constructor has no return type.
D. A constructor has a void return type.

Ans:

Option C
A constructor has no return type.

View All Answers

Question - 38:

To ensure that every object in the array receives a destructor call, always delete memory allocated as an array with operator __________ .
A. destructor
B. delete
C. delete[]
D. kill[]
E. free[]

Ans:

Option C
delete[]

View All Answers

Question - 39:

Which of the following gets called when an object is being created?
A. constructor
B. virtual function
C. destructor
D. main

Ans:

Option A
constructor

View All Answers

Question - 40:

Can you please explain the order in which constructors are called when an object of a derived class is created?

Ans:

The constructors of any virtual base classes are called first in the order of inheritance.
Non-virtual base class constructors are called next.
The derived class constructor is called last.

View All Answers

Question - 41:

Tell me what are the restrictions apply to constructors and destructors?

Ans:

The following restrictions apply to constructors and destructors
Constructors and destructors don't return values.
The addresses of constructors and destructors can't be taken so we can't use references and pointers on them.
Constructors cannot be declared with the keyword virtual.
Constructors and destructors cannot be declared static, const, or volatile.

View All Answers

Question - 42:

What is deep copy?

Ans:

A deep copy creates a copy of the dynamically allocated objects too. You would need to use a copy constructor and overload an assignment operator for this.

View All Answers

Question - 43:

What is shallow?

Ans:

A shallow copy just copies the values of the data as they are. Even if there is a pointer that points to dynamically allocated memory, the pointer in the copy will point to the same dynamically allocated object.

View All Answers

Question - 44:

Can you please explain the difference between copy constructor and an assignment operator?

Ans:

A copy constructor is used to declare and initialize an object from another object.
E.g: integer I2(I1);
An assignment operator doesnot invoke the copy constructor. It simply assigns the values of an object to another, member by member.

View All Answers

Question - 45:

What is virtual destructor and how to use it?

Ans:

If the destructor in the base class is not made virtual, then an object that might have been declared of type base class and instance of child class would simply call the base class destructor without calling the derived class destructor.
Hence, by making the destructor in the base class virtual, we ensure that the derived class destructor gets called before the base class destructor.

```
class a
{
        public:
        a(){printf("
Base Constructor
");}
        ~a(){printf("
Base Destructor
");}
};
class b : public a
{
       public:
       b(){printf("
Derived Constructor
");}
       ~b(){printf("
Derived Destructor
");}
};
int main()
{
       a* obj=new b;
       delete obj;
       return 0;
}
```

Output:

```
Base Constructor
Derived Constructor
Base Destructor
```

By Changing

```
~a(){printf("
Base Destructor
");}
```

to

```
virtual ~a(){printf("
Base Destructor
");}
```

Output:

```
Base Constructor
Derived Constructor
Derived Destructor
Base Destructor
```

View All Answers

Question - 46:

Do you know what are destructors?

Ans:

Destructors are complements of constructors. When an object is destroyed, its destructor is automatically called. Destructors are mainly useful for doing the clean up job. E.g. an object may have allocated some memory during its lifetime; destructors are the place where this memory is deallocated. Or an object may need to close some files by releasing its handles which it had previously obtained.
Destructor function has same name as that of a constructor; but the name is preceded by a tilde (â€˜~') sign.
We will expand the above example of a stack class to include a destructor:

```
#include <iostream>
using namespace std;
class stack
{
        int top, bottom;
        int data[20];
        stack()         //constructor function
        {
              top = bottom;
              cout <<â€•Inside Stack Constructor: Stack initialized
â€•;
        }
        int stackfull()
        {
             return ((top == max â€" 1)?1:0);
        }
        int stackempty()
        {
             return (top == bottom)?1:0);
        }
        void push(int no)
        {
             if(stackfull())
                  cout<<â€•Stack is fullâ€•;
             else
                   data[++top] = no;
        }
        int pop()
        {
             if(stackempty())
                 cout<<â€•Nothing to pop.. Stack is Empty!
â€•;
             else
                 return(data[top- -];
        }
        ~stack() //destructor function
         {
             cout <<â€•Inside Stack Destructor: Stack Destroyed
â€•;
         }
};
int main()
{
      int i, no;
      stack st; //object is created; hence constructor is invoked- stack initialization done
      cout <<â€•Entered Main
â€•;
      for(i = 0; i < 10; i++)
              st.push(i);
     no = s.pop();
     cout <<â€•The popped element is:â€•<<no << â€œ
â€•;
     return 0;
}// at the end of object's lifetime, destructor is invoked
```

The o/p of the program would be:

```
Entered Main
Inside Stack Constructor: Stack initialized
The popped element is: 9
Inside Stack Destructor: Stack Destroyed
```

As seen from the o/p the constructors and destructors are automatically called.

View All Answers

Question - 47:

Tell me how should a constructor handle a failure?

Ans:

Throw an exception
Constructors don't have a return type, so it's not possible to use return codes. The best way to signal constructor failure is therefore to throw an exception.

View All Answers

Question - 48:

Can you please explain Explain constructors and destructors?

Ans:

Constructors are the member functions of the class that executes automatically whenever an object is created. Constructors have the same name as the class. Constructors initialize the class. Constructors can't have return type. Destructors are called when the objects are destroyed.
Destructors are usually used to deallocate memory and do other cleanup for a class object and its class members when the object is destroyed. A destructor is called for a class object when that object passes out of scope or is explicitly deleted. A destructor takes no arguments and has no return type.

View All Answers

Question - 49:

What is constructor in C++?

Ans:

Constructors allow initialization of objects at the time of their creation. Constructor function is a special function that is a member of the class and has same name as that of the class. An object's constructor is automatically called whenever the object is created (statically or dynamically). Constructors are always public. They are

used to make initializations at the time of object creation.
Consider following example of a stack class:

```
#include <iostream>
using namespace std;
class stack
{
           int top, bottom;
           int data[20];
           stack()        //constructor function
          {
              top = bottom;
              cout <<â€•Inside Stack Constructor: Stack
                      initializedâ€•;
          }
          int stackfull()
          {
                return ((top == max â€" 1)?1:0);
          }
          int stackempty()
          {
               return (top == bottom)?1:0);
          }
          void push(int no)
          {
                 if(stackfull())
                        cout<<â€•Stack is fullâ€•;
                 else
                        data[++top] = no;
          }
          int pop()
          {
                 if(stackempty())
                     cout<<â€•Nothing to pop.. Stack is Empty!â€•;
                 else
                      return(data[top--];
          }
};
int main()
{
         int i, no;
         stack st; //object is created; hence constructor is
              invoked-  stack initialization done
         cout <<â€• Entered Main
â€•;
         for(i = 0; i < 10; i++)
              st.push(i);
          no = s.pop();
          cout <<â€•The popped element is:â€•<
              return 0;
}
```

The o/p of the program would be:
Entered Main
Inside Stack Constructor: Stack initialized
The popped element is: 9
As seen above, the stack object is initialized automatically at the time of creation by the constructor. So we don't need to write and invoke initialization functions explicitly.

View All Answers

Question - 50:

What is virtual constructor?

Ans:

A constructor of a class can not be virtual and if causes a syntax error.

View All Answers

Question - 51:

What is Virtual destructors?

Ans:

The explicit destroying of object with the use of delete operator to a base class pointer to the object is performed by the destructor of the base-class is invoked on that object.
The above process can be simplified by declaring a virtual base class destructor.
All the derived class destructors are made virtual in spite of having the same name as the base class destructor. In case the object in the hierarchy is destroyed explicitly by using delete operator to the base class pointer to a derived object, the appropriate destructor will be invoked.

View All Answers

C++ Most Popular & Related Interview Guides

1 : **C++ Pointers & Functions Interview Questions and Answers.**

2 : C++ Operator Overloading Interview Questions and Answers.

3 : C++ Exception Handling Interview Questions and Answers.

4 : C++ Template Interview Questions and Answers.

5 : C++ Friend Interview Questions and Answers.

6 : C++ Virtual Functions Interview Questions and Answers.

7 : C++ Type Checking Interview Questions and Answers.

8 : C++ Inheritance Interview Questions and Answers.

9 : C++ Access Control Interview Questions and Answers.

10 : C++ Inline Function Interview Questions and Answers.

Follow us on FaceBook
www.facebook.com/InterviewQuestionsAnswers.Org

Follow us on Twitter
https://twitter.com/InterviewQA

For any inquiry please do not hesitate to contact us.

Interview Questions Answers.ORG Team
https://InterviewQuestionsAnswers.ORG/
support@InterviewQuestionsAnswers.ORG