C++ Exception Handling Interview Questions And Answers Guide.

Global Guideline.

http://www.globalguideline.com/
Question # 1
Can you please Illustrate Rethrowing exceptions?

Answer:-
Rethrowing an expression from within an exception handler can be done by calling throw, by itself, with no exception. This causes current exception to be passed on to an outer try/catch sequence. An exception can only be rethrown from within a catch block. When an exception is rethrown, it is propagated outward to the next catch block.
Consider following code:
```cpp
#include <iostream>
using namespace std;
void MyHandler()
{
    try
    {
        throw "hello";
    } catch (const char*)
    {
        cout <<"Caught exception inside MyHandler";
        throw; //rethrow char* out of function
    }
} int main()
{
    cout << "Main start";
    try
    {
        MyHandler();
    } catch(const char*)
    {
        cout <<"Caught exception inside Main";
    }
    cout << "Main end";
    return 0;
}
```
O/p:
Main start
Caught exception inside MyHandler
Caught exception inside Main
Main end
Thus, exception rethrown by the catch block inside MyHandler() is caught inside main();

Question # 2
Explain Exception handling for a class with an example?

Answer:-
Exceptions are certain disastrous error conditions that occur during the execution of a program. They could be errors that cause the programs to fail or certain conditions that lead to errors. If these run time errors are not handled by the program, OS handles them and program terminates abruptly, which is not good. To avoid this, C++ provides Exception Handling mechanism.

The three keywords for Exception Handling are:
Try, Catch and Throw.
The program tries to do something. If it encounters some problem, it throws an exception to another program block which catches the exception.
Ex:
```cpp
void main()
```
```cpp
int no1, no2;
try
{  
cout << "Enter two nos:";
cin >> no1 >> no2;
if (no2 == 0)
  throw "Divide by zero";
else
  throw no1/no2;
}
catch (char *s)
{
  cout << s;
}
catch (int ans)
{
  cout << ans;
}
}
```

We know that divide by zero is an exception. If user enters second no as zero, the program throws an exception, which is caught and an error message is printed else the answer is printed.

**Question # 3**
Describe exceptions in C++?

**Answer:-**
Exceptions: Exceptions are certain disastrous error conditions that occur during the execution of a program. They could be errors that cause the programs to fail or certain conditions that lead to errors. If these run time errors are not handled by the program, OS handles them and program terminates abruptly, which is not good.

So C++ provides Exception Handling mechanism.

**Question # 4**
What is Synchronous Exceptions?

**Answer:-**
Errors that occur due to incorrect input data or incorrect handling of array indices ("out of range index"), memory overflow are known as Synchronous Exceptions.

**Question # 5**
What is Asynchronous Exceptions?

**Answer:-**
The errors that are caused by events that are beyond control of the program are Asynchronous Exceptions. E.g. Keyboard interrupts.

**Question # 6**
Explain benefits of Exception Handling?

**Answer:-**
The benefits of Exception Handling are:
1. Program is not terminated abruptly
2. User will understand what errors are occurring in the program.

The three keywords for Exception Handling are: Try, Catch and Throw.

**Question # 7**
Explain terminate() function?

**Answer:-**
terminate() is a library function which by default aborts the program. It is called whenever the exception handling mechanism cannot find a handler for a thrown exception.

**Question # 8**
Explain unexpected() function?

**Answer:-**
unexpected() is called when a function with an exception specification throws an exception of a type that is not listed in the exception specification for the function. A function declaration without a specification like throw(char*) may throw any type of exception, and one with throw() is not allowed to throw exceptions at all. By default unexpected() calls terminate().
Question # 9
How to implement exception handling in C++?

Answer:-
Exception handling in C++ is implemented by using the try{} and catch{} statements. When a try block throws an exception, the program leaves the try block and enters the catch statement of the catch block. If they type of the object thrown matches the arg type in the catch block, catch block is executed for handling the code. If they are not caught, abort() function is executed by default. When no exception is detected or thrown then the control goes to the statement below the catch block.

Read More Answers.

Question # 10
An exception is thrown using _____________ keyword in CPP.
a. throws
b. throw
c. threw
d. thrown?

Answer:-
b. throw

Read More Answers.

Question # 11
The code of statements which may cause abnormal termination of the program should be written under_________ block.
a. try
b. catch
c. Finally
d. None of these

Answer:-
a. try

Read More Answers.

Question # 12
Catch handler can have multiple parameters.
a. True
b. False

Answer:-
b. False

Read More Answers.

Question # 13
Exception handlers are declared with ____________ keyword.
a. Try
b. catch
c. throw
d. finally

Answer:-
b. catch

Read More Answers.

Question # 14
A try block can be nested under another try block -
a. Yes
b. No

Answer:-
a. Yes

Read More Answers.

Question # 15
Which of the following statements are true about Catch handler?
1. It must be placed immediately after try block T
2. It can have multiple parameters
3. There must be only one catch handler for every try block
4. There can be multiple catch handler for a try block T
5. Generic catch handler can be placed anywhere after try block.
a. Only 1, 4, 5
b. Only 1, 2, 3
c. Only 1, 4
d. Only 1, 2

Answer:-
a. Only 1, 4, 5

Read More Answers.
c. Only 1, 4

**Question # 16**
In nested try blocks, there is no need to specify catch handler for inner try block. Outer catch handler is sufficient for the program.

a. True
b. False

**Answer:-**
b. False

**Question # 17**
In nested try block, if inner catch handler gets executed, then__________

a. Program execution stops immediately
b. Outer catch handler will also get executed
c. Compiler will jump to the outer catch handler and then executes remaining executable statements of main()
d. Compiler will execute remaining executable statements of outer try block and then the main()

**Answer:-**
d. Compiler will execute remaining executable statements of outer try block and then the main()

**Question # 18**
Generic catch handler must be placed at the end of all the catch handlers.

a. True
b. False

**Answer:-**
a. True

**Question # 19**
If inner catch handler is not able to handle the exception then__________.

a. Compiler will look for outer try handler
b. Program terminates abnormally
c. Compiler will check for appropriate catch handler of outer try block
d. None of these

**Answer:-**
c. Compiler will check for appropriate catch handler of outer try block

**Question # 20**
In nested try blocks, if both inner and outer catch handlers are not able to handle the exception, then__________.

a. Compiler executes only executable statements of main()
b. Compiler issues compile time errors about it
c. Program will be executed without any interrupt
d. Program will be terminated abnormally

**Answer:-**
d. Program will be terminated abnormally

**Question # 21**
Functions called from within a try block may also throw exception.

a. True
b. False

**Answer:-**
a. True

**Question # 22**
Irrespective of exception occurrence, catch handler will always get executed.

a. True
b. False

**Answer:-**
b. False
Question # 23
An exception can be of only built-In type.
a. True
b. False

**Answer:-**
b. False
[Read More Answers.]

Question # 24
We can prevent a function from throwing any exceptions.
a. True
b. False

**Answer:-**
a. True
[Read More Answers.]

Question # 25
How can we restrict a function to throw certain exceptions?
a. Defining multiple try and catch block inside a function
b. Defining generic function within try block
c. Defining function with throw clause
d. It is not possible in CPP to restrict a function

**Answer:-**
c. Defining function with throw clause
[Read More Answers.]

Question # 26
Return type of uncaught_exception() is_______________.
a. int
b. bool
c. char *
d. double

**Answer:-**
b. bool
[Read More Answers.]

Question # 27
Attempting to throw an exception that is not supported by a function call results in calling ____________ library function.
a. indeterminate()
b. unutilized()
c. unexpected()d. unpredicted()

**Answer:-**
c. unexpected()
[Read More Answers.]

Question # 28
Throwing an unhandled exception causes standard library function ______________ to be invoked.
a. stop()
b. aborted()
c. terminate()
d. abandon()

**Answer:-**
c. terminate()
[Read More Answers.]

Question # 29
Generic catch handler is represented by _____________.
a. catch(...) b. catch(...) c. catch(...) d. catch( void x)

**Answer:-**
c. catch(...)  
[Read More Answers.]
C++ Programming Most Popular Interview Topics.

1: Basic C++ Syntax Frequently Asked Interview Questions and Answers Guide.
2: C++ Virtual Functions Frequently Asked Interview Questions and Answers Guide.
3: C++ Constructors Frequently Asked Interview Questions and Answers Guide.
4: C++ Operator Overloading Frequently Asked Interview Questions and Answers Guide.
5: C++ Access Control Frequently Asked Interview Questions and Answers Guide.
6: C++ Inheritance Frequently Asked Interview Questions and Answers Guide.
7: C++ Friend Frequently Asked Interview Questions and Answers Guide.
8: C++ Containers Frequently Asked Interview Questions and Answers Guide.
9: C++ COM ActiveX Frequently Asked Interview Questions and Answers Guide.
10: C++ Pointers & Functions Frequently Asked Interview Questions and Answers Guide.
About Global Guideline.

Global Guideline is a platform to develop your own skills with thousands of job interview questions and web tutorials for fresher's and experienced candidates. These interview questions and web tutorials will help you strengthen your technical skills, prepare for the interviews and quickly revise the concepts. Global Guideline invite you to unlock your potentials with thousands of Interview Questions with Answers or begin a tutorial right away, such as HTML, XML, XSLT, Cascading Style Sheet (CSS), Search Engine Optimization (SEO), JavaScript, Structure Query Language (SQL), Database Articles, Web Hosting Guide and much more. Learn the most common technologies Interview Questions and Answers. We will help you to explore the resources of the World Wide Web and develop your own skills from the basics to the advanced. Here you will learn anything quite easily and you will really enjoy while learning. Global Guideline will help you to become a professional and Expert, well prepared for the future.

* This PDF was generated from http://www.GlobalGuideline.com at January 29th, 2017

* If any answer or question is incorrect or inappropriate or you have correct answer or you found any problem in this document then don't hesitate feel free and e-mail us we will fix it.

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

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

Best Of Luck.

Global Guideline Team
http://www.globalguideline.com
Support@globalguideline.com