# Department of Electrical Engineering

**CS212**

**Object Oriented Programming**



# Lab 12: Exceptions

**Class**: BEE - 12C

**Date**: December 27th, 2021

**Time**: Monday (1400 – 1700)

**Name**: Muhammad Umer

**CMS ID**: 345834

**Tools**

* Microsoft Visual Studio 2013

**Exceptions Handling**

One of the advantages of C++ over C is Exception Handling. Exceptions are run-time anomalies or abnormal conditions that a program encounters during its execution. There are two types of exceptions: a)Synchronous, b)Asynchronous(Ex:which are beyond the program’s control, Disc failure etc). C++ provides following specialized keywords for this purpose.

* **try**: represents a block of code that can throw an exception.
* **catch**: represents a block of code that is executed when a particular exception is thrown.
* **throw**: Used to throw an exception. Also used to list the exceptions that a function throws, but doesn’t handle itself.

**Lab Tasks**

* **Task 1**

**Following is a simple example to show exception handling in C++. The output of program explains flow of execution of try/catch blocks. Show the output of the following program.**

**#include <iostream>**

**using namespace std;**

**int main()**

**{**

**int x = -1;**

**// Some code**

**cout << "Before try \n";**

**try {**

**cout << "Inside try \n";**

**if (x < 0)**

**{**

**throw x;**

**cout << "After throw (Never executed) \n";**

**}**

**}**

**catch (int x ) {**

**cout << "Exception Caught \n";**

**}**

**cout << "After catch (Will be executed) \n";**

**return 0;**

**}**

**Terminal Output**

PS D:\NUST\Semester 3\Object Oriented Programming\Labs> lab12.exe

Before try

Inside try

Exception Caught

After catch (Will be executed)

* **Task 2**

**There is a special catch block called ‘catch all’ catch(…) that can be used to catch all types of exceptions. For example, in the following program, an int is thrown as an exception, but there is no catch block for int, so catch(…) block will be executed.(show the output).**

**#include <iostream>**

**using namespace std;**

**int main()**

**{**

**try  {**

**throw 10;**

**}**

**catch (char \*excp)  {**

**cout << "Caught " << excp;**

**}**

**catch (...)  {**

**cout << "Default Exception\n";**

**}**

**return 0;**

**}**

**Terminal Output**

PS D:\NUST\Semester 3\Object Oriented Programming\Labs> lab12.exe

Default Exception

* **Task 3:**

**Find error and mention in comment.**

**#include <iostream>**

**using namespace std;**

**int main()**

**{**

**try  {**

**throw 'a';**

**}**

**catch (int x)  {**

**cout << "Caught " << a;**

**}**

**catch (x,a)  {**

**cout << "Default Exception\n";**

**}**

**return 1;**

**}**

## Code

#include <iostream>

using namespace std;

int main()

{

    try  {

       throw 'a';

    }

    catch (char x)  {                // NOT AN ERROR, but since a char is being

                                    // thrown, use a char to catch instead

        cout << "Caught " << x;   // Undefined identifier 'a' being used

                                    // use 'x' instead

    }

    catch (...)  {              // x is not a data type, use ellipsis instead

        cout << "Default Exception\n";

    }

    return 0;                   //  A return 1 means that there is some error while

                                //  executing the program and it is not

                                //  performing what it was intended to do.

}

**Terminal Output**

PS D:\NUST\Semester 3\Object Oriented Programming\Labs> lab12.exe

Caught a

* **Task 4:**

**Is it possible to keep other statements in between ‘try’, ‘catch’, and ‘finally’ blocks?**

No, we cannot write any statements in between**try, catch and finally blocks**and these blocks **form one unit.** If we try to put any statements between these blocks, it will throw a **compile-time error.**