Exception Handling in C++

Exceptions

- An exception is a problem that arises during the execution of a program.
- Exceptions are run-time anomalies or abnormal conditions that a program encounters during its execution.
- It allows us to transfer control from one part of a program to another.
- There are two types of exceptions:

a) **Synchronous**-occur during program execution due to some fault in input data e.g. out of range exception

b) **Asynchronous** - which are beyond the program's control e.g. Disc failure.

- Exception handling is supported by three keywords **try, catch and throw**.

try block

- A try block identifies a block of code which throws an exception.
- It is followed by one or more catch blocks.
- Syntax:

```
try
{
//code which throws an exception
}
```

Catch Block

- It represents a block of code that is executed when a particular exception is thrown.
- We can handle any type of exception from catch block like int ,char etc.
- Syntax:

```
catch(int x)
{
   cout<<“Exception Caught”<<x<<endl;
}
```

Throw

- It is used to throw an exception.
- It transfers the control to the exception handler(catch block). If no exception is thrown the code continues normally and all handlers are ignored.
- An exception is thrown using throw keyword from within try block.
- Syntax:

```
throw e; // e is object of exception
```

Multiple Exception

```
try
 {  //  code  }
 catch( ExceptionName e1 )
{    // catch block }
 catch( ExceptionName e2 )
 { // catch block }
catch( ExceptionName eN )
 { // catch block }
```

Example of try , catch & throw

```
#include<iostream.h>
using namespace std;
void main()
{
 int a=1;
 try
 {
    if(a==0)
       throw a;     //throwing integer exception
    else if(a==1)
       throw 'A';     //throwing character exception
    else if(a==2)
       throw 3.5;     //throwing float exception
}
catch(int a)
 {
     cout<<"\n Integer exception caught.";
}
catch(char ch)
{
    cout<<"\n Character exception caught.";
}
catch(double d)
{
   cout<<"\n Double exception caught.";
 }
    cout<<"\n End of program.";
}
```

Output :

Character exception caught.

End of program.

Generic Catch Block

- An exception is handled using catch block.
- It handles any type of exception.
- Syntax:

```
try
{
// code
}
catch(…)
{
//Block of code that handles exception  }
```

Example of Generic Catch Block

```
#include<iostream.h>
using namespace std;
void main()
{
int n1,n2,res,e;
try
{
 if(n2==0)
  throw ‘e’;
else
{
    res=n1/n2;
    cout<<“res=“<<res<<endl;
}
}
```

```
catch(int a)
{
  cout<<“Exception caught”<<a;
}
catch(…)
{
cout<<“Exception caught”;
}
```

If type of exception object does not matches with any object then it execute generic exception as a default exception.

Thank You