

BASICS OF EXCEPTION HANDLING

C++ provides means for exception (or error) handling. You use the 'try', 'catch', and 'throw' keywords for this purpose. When programming, you tell the compiler to 'try' a section of code. If an error occurs in the section (or in any nested methods/functions in the section), you 'throw' an exception. The exception is then handled like a hot potato and is passed up the ladder until it is caught. If an exception is not caught, the program terminates. Here's the basic syntax:

```
// exceptions.cp

#include <iostream.h>

void doFunction( float x );

main()

    float val = 5.0;

    try

        doFunction( val );

    catch( const char *s )

        cout << "string exception caught" << endl;
        cout << s << endl;

    catch(...)

        // catch all other exceptions here

    return 0;

void doFunction( float x )

    float limit = 3.14159;

    if( x > limit )
        throw "value exceeds limit";

// end exceptions.cp
```

In most C++ applications, you 'throw' an exception class object rather than a string or error code. An example of this is shown below:

```
// potatoe.cp

#include <iostream.h>

void doFunction( float x );

class HotPotatoe

public:
    HotPotatoe(float m, float v) max=m; val=v;
```

```

    float max;
    float val;
;

main()

    float val = 5.0;

    try

        cout << "Example One:" << endl;
        doFunction( val );

    catch( HotPotatoe &thePotatoe )

        cout << "exception caught" << endl;
        cout << "  max. limit: " << thePotatoe.max << endl;
        cout << "  value: " << thePotatoe.val << endl << endl;

    try

        cout << "Example Two:" << endl;
        try

            doFunction( val );

        catch( HotPotatoe )

            cout << "exception caught" << endl;
            cout << "exception not handled" << endl;
            cout << "rethrow exception" << endl;
            throw;

    catch(...)

        cout << "rethrown exception caught" << endl;

    return 0;

void doFunction( float x )

    float limit = 3.14159;

    if( x > limit )

        cout << "throw exception" << endl;
        throw HotPotatoe(limit,x);

// end potatoe.cp

```