黑客派

C 语言实现 try catch 处理

作者：bfchen

原文链接：https://hacpai.com/article/1510981407646

来源网站：黑客派

我们都知道，在 Java、C# 等高级程序语言中，都自带有异常处理机制，其基本结构如下：
 ry{
 程序语句;
 }catch(Exception ex){
 异常处理;
 }
 这样做不但可以防止程异常终止，而且在出现错误时可以及时作一些释放资源处理，对程序能继续健壮的运行下去尤其重要

但是 C 语言中没有这样异常处理机制，我们可以通过宏定义去实现类似这样的操作，这时候就用到 C 语言强大而又被我们‘嫌弃’的 goto 语句了。
 这里我们需要定义几个宏：
 （1）常处理框架

 在 BEGIN_PROC 和 END_PROC 里处理我的程序，有异常的情况时跳到 CATCH_ERROR 和 END_ERROR 里统一处理
 （2）几种异常条宏定义：
（3）异常统一处理宏定义：
下面我们来分析这三种情况：
 （1）不满足某种条件产生异常跳转：

 这里是判断数组是否索引是否越界，如果越界则判断跳转失败，程序如何结构：

 可以看到，这里打印了出错的函数名、行号和错误码我们可以很容易的定位出错误
 (2)出错的时候抛出错误，统一处理：

 这里我们打开文件或分配内存失败都统一到错误处理当中，而不是直接 return 返回这是一种很好的处理方法，不然很容易导致内存泄露或文件未关闭，
 从上面我们可以看出，当用 return 语句时，文件未打开成功，则返回，而当文件打开成功，而内存申请失败的时候，我们很易直接返回，而导致
 文件未正常关闭。使用了框架后，我们把使用的资源都作一个统一的管理这样就不会轻易造成资源泄露的问题了，同时也有问题定位打印信息
 (3)执行函数是否成功而跳：

 当函数执行的返回值不成功时，我们也进行到异常转，进行统一处理
 以上就是 C 语言的一些常见的异常处理，当然还有其他的异常处理条件，但基于以上框架可以很容易的做到。