

## LongXiaJun30天读书笔记分享计划 Day 8: C++

### MEC条款14: 明智运用exception specifications

处理异常是好事，但是不要过度使用



不应该将template与exception specification混合使用

A调用B, B无exception, 则A应该也没有exception

处理系统可能抛出的exception

编译器只做局部性检验, 很容易被违反

### MEC条款15: 了解异常处理的成本



做好权衡

为了在运行期处理异常, 程序必须做大量的簿记工作

try语句块, 代码会大约整体膨胀5-10%

### C++ STL

底层实现: 线性移动操作

move

OutputIt move( InputIt first, InputIt last, OutputIt d\_first ) ○ 移动范围 [first, last) 中的元素到始于 d\_first 的另一范围, 从首元素开始逐次到末元素。此操作后被移动范围中的元素将仍然含有适合类型的合法值, 但不必与移动前的值相同

Bidirectional move\_backward

Bidirectional move\_backward( Bidirectional first, Bidirectional last, Bidirectional d\_last ) ○ 移动来自范围 [first, last) 的元素到终于 d\_last 的另一范围。以逆序移动元素 (首先复制末元素), 但保持其相对顺序。

fill

void fill( ForwardIt first, ForwardIt last, const T& value ) ○ 赋值给定的 value 给 [first, last) 中的元素

OutputIt fill\_n( OutputIt first, Size count, const T& value ) ○ 若 count > 0, 则赋值给定的 value 给始于 的范围的首 count 个元素。否则不做任何事

### EC条款8: 别让异常逃离析构函数

dtor绝对不要抛出异常

在两个异常存在的情况下, 程序若不是结束执行就是导致不明确行为

如果一个被析构函数调用的函数可能抛出异常, 析构函数应该捕捉任何异常, 然后吞下他们 (不传播) 或结束程序

如果客户需要对某个操作函数运行期间抛出的异常做出反应, 那么class应该提供一个普通函数 (而非在析构函数中) 执行该操作

2018.12.14 Friday. By KuKuXia@github.com