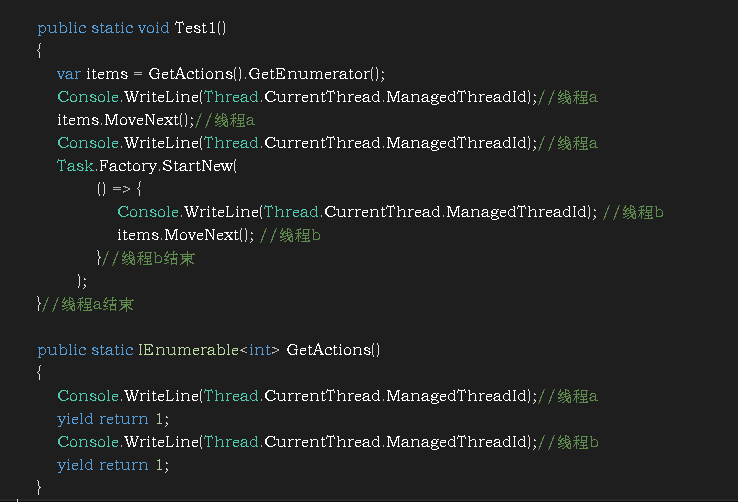
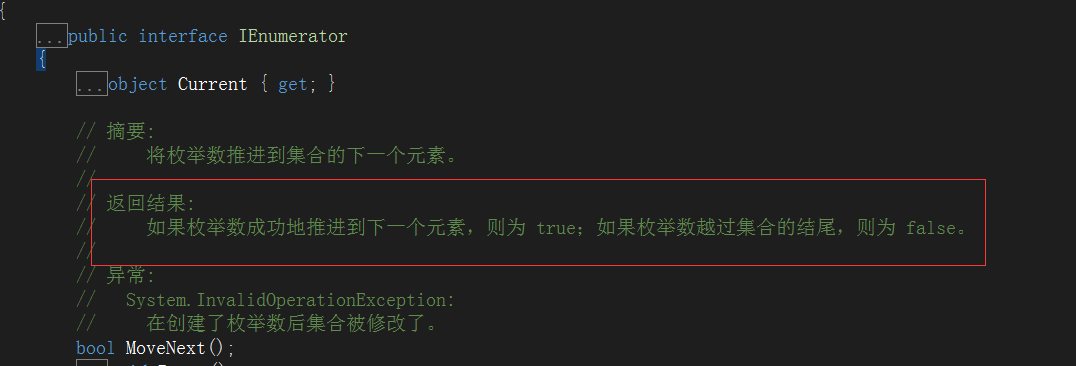
1. 关于IEnumerator的线程



调用了movenext后，当前yield return之后执行的代码所在的线程就是调用movenext那个方法的线程，线程直到方法结束才会被释放

二．

关于IEnumerator的MoveNext

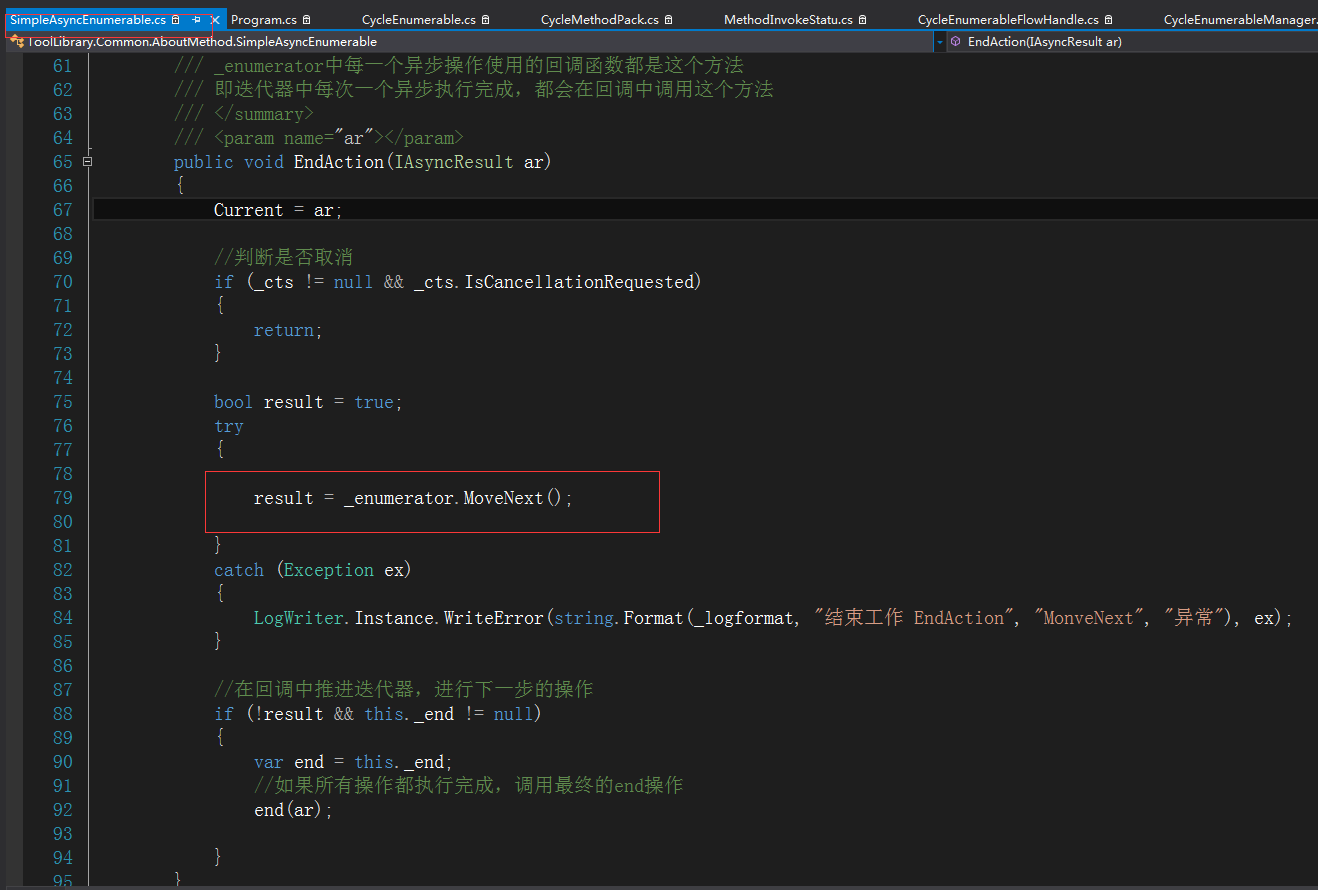


这个方法的返回值通常用来作为遍历是否结束的判断，但是需要注意，如果MoveNext是在多线程下被调用，那么很有可能在没有遍历结束的时候，得到返回值是false，这是因为MoveNext要求线程安全

每一个线程，会在调用MoveNext后将迭代推送到下一个yiled return后释放，但是在一个线程没有把遍历推送到下一个yield ruturn之前，其他线程在调用MoveNext时，得到的必然是false。

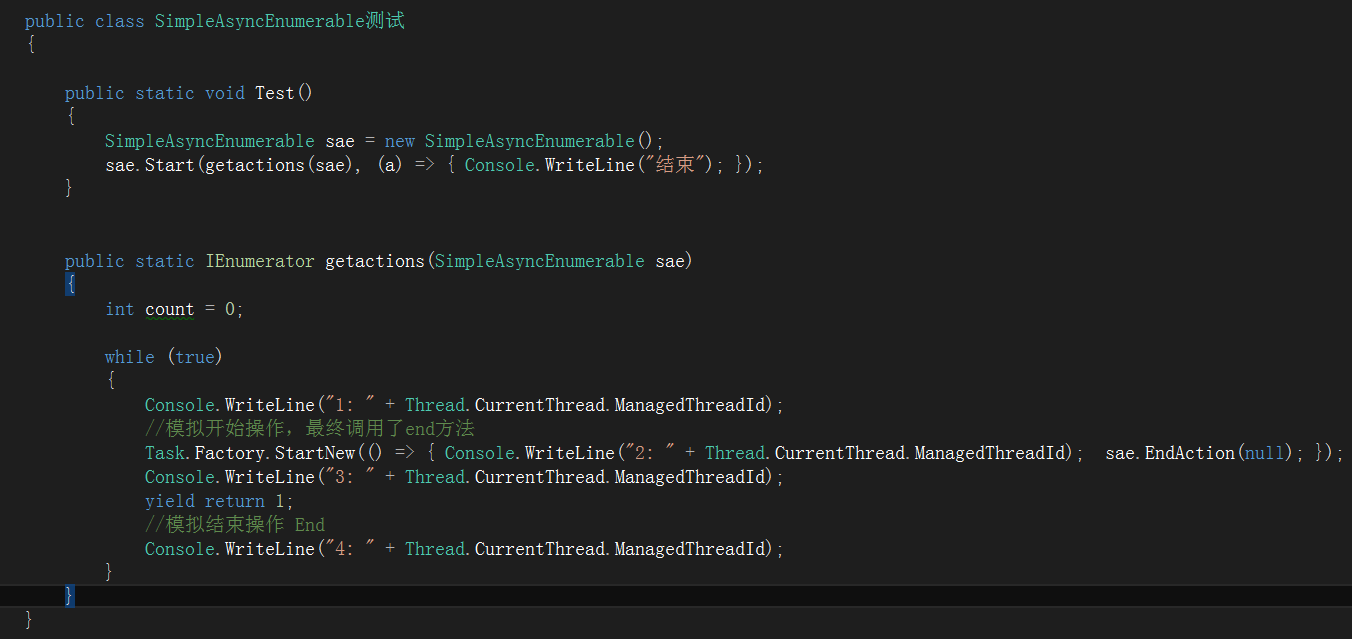
因此这个方法一定要保证线程安全

相关使用：

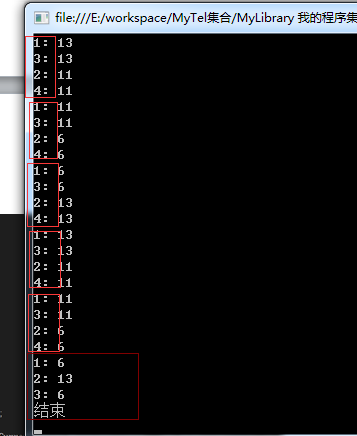


在SimpleAsyncEnumerable中，EndAction方法作为外部每次异步结束时的回调函数，在这个方法中会调用MoveNext方法，并根据返回值判断循环是否结束。

但是实际使用中：



理论上是无限执行的，但是实际上：



正常情况，每次调用MonvNext时候，都是单线程去操作的，但是在结束前的最后一次循环中，2在3之前出现，也就是说，在当前线程推送到yield return 1 之前，EndAction已经被执行，内部的MoveNext方法被调用，此时迭代是不能被成功处理的，所以得到的必然是false，因此循环结束。