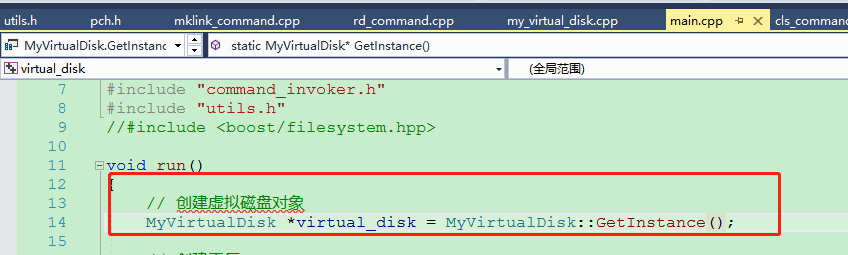
1. 给我讲下整体思路
2. 整体流程再设计下，方案搞详细，每个命令最好搞一个流程图
3. 判空



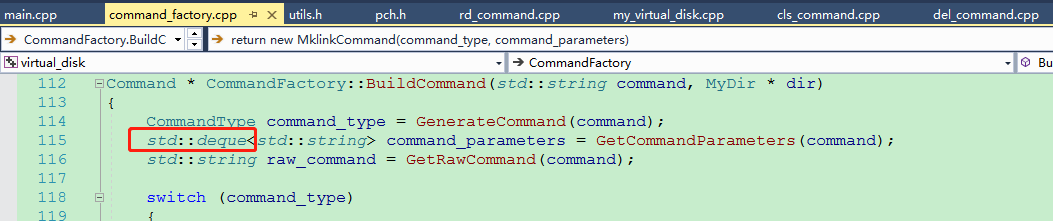
1. 这个为啥不用指针了，还有这个构造函数你是咋理解的



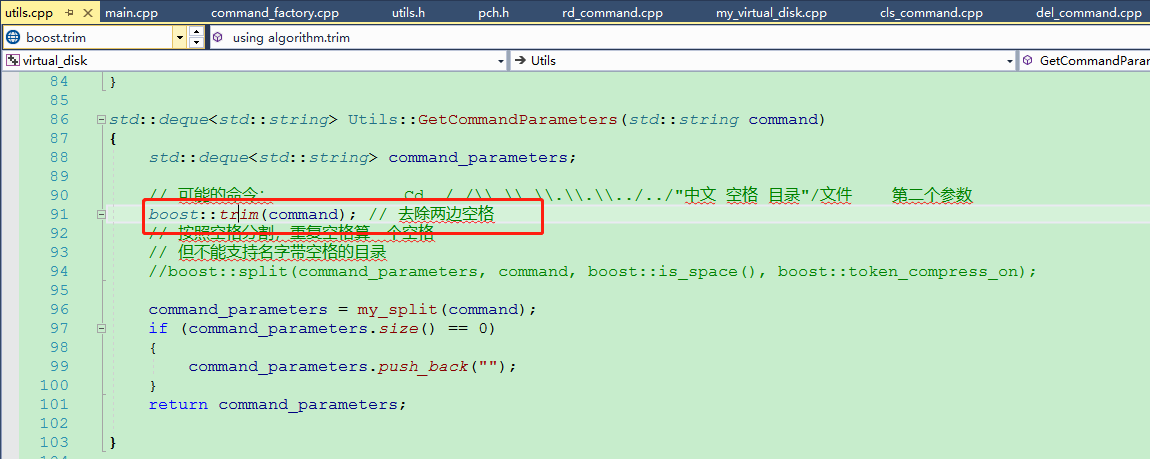
1. 这个string赋值你是咋理解的



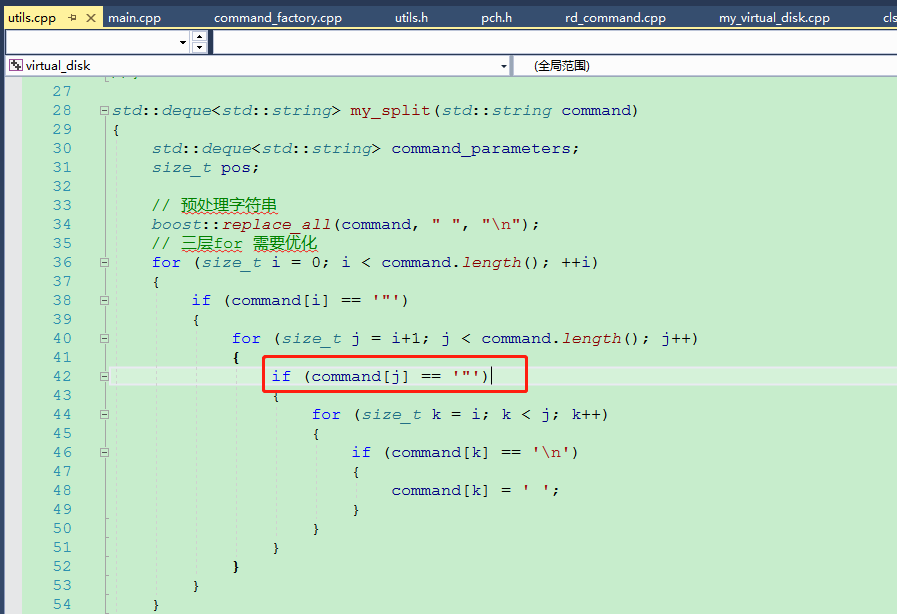
1. 这个是个啥，讲讲它是怎么存储的，增删改查的复杂度是多少



1. 为什么考虑用boost，咱们项目开发一般很少引入第三方库，需要评审
2. 这个函数是什么行为



1. 这个找不到怎么办



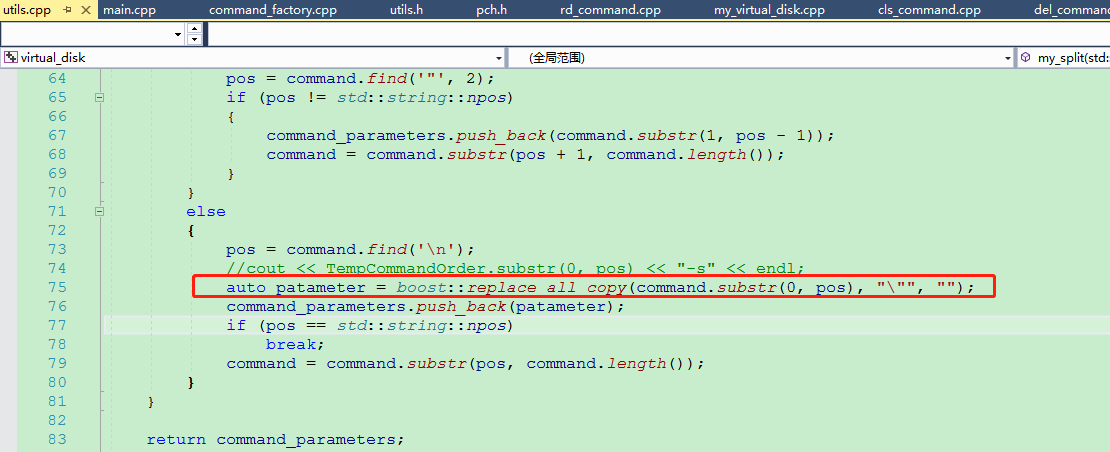
1. 这个为啥用2



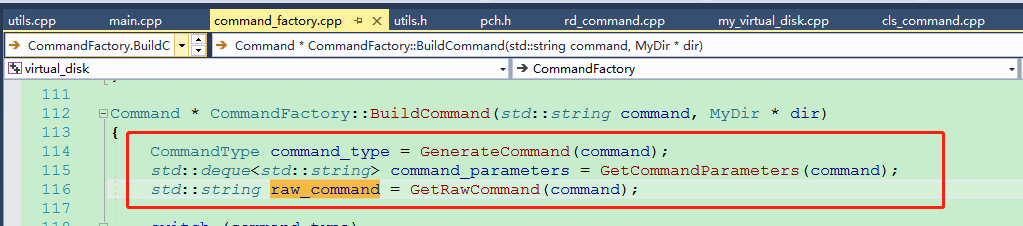
1. 这个要是没找到怎么办



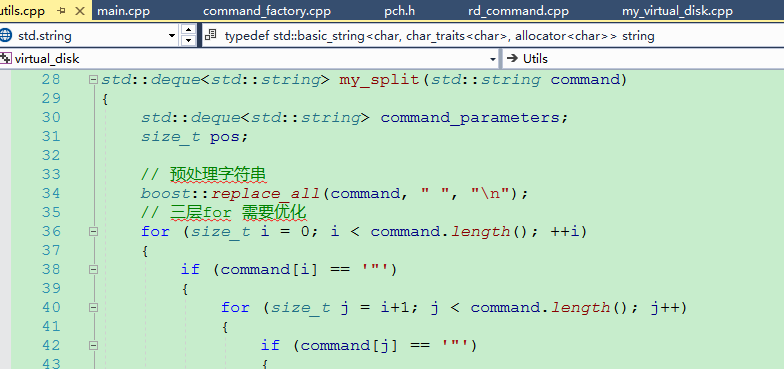
1. 这种情况是不是应该直接报错



1. 上面说的这个字符串分割优化下
2. 异常输入情况看起来都没有处理，得处理下
3. 这三个每个里面都有字符串处理的函数，执行了重复代码，优化下



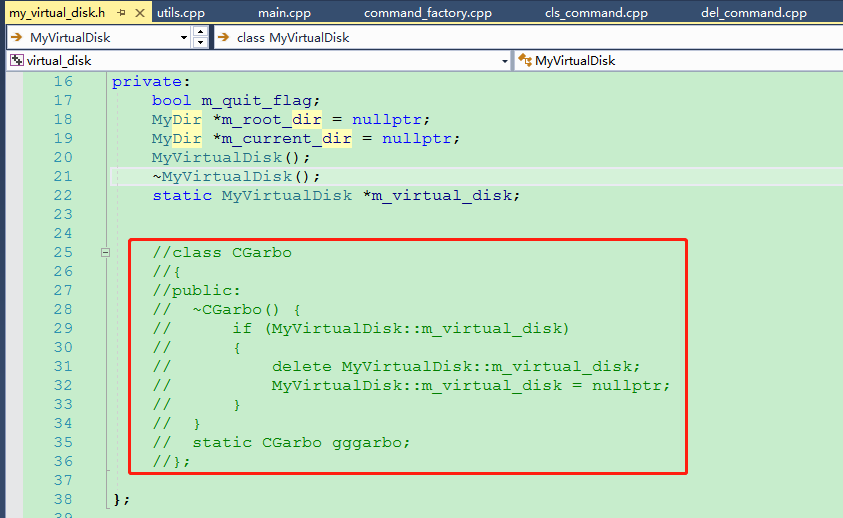
1. Const引用，所有函数传参都过一遍



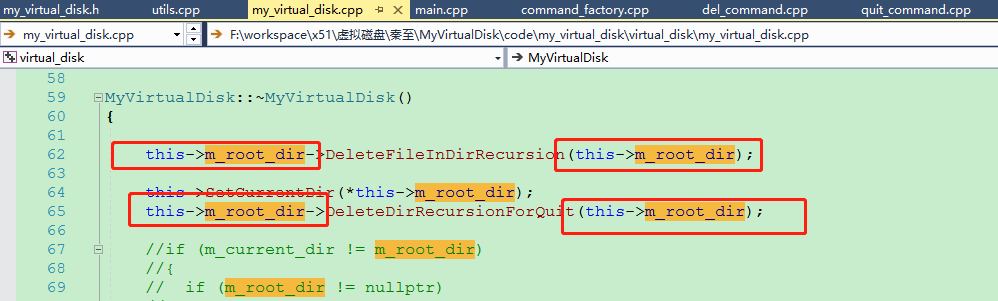
1. 这个参数没用到



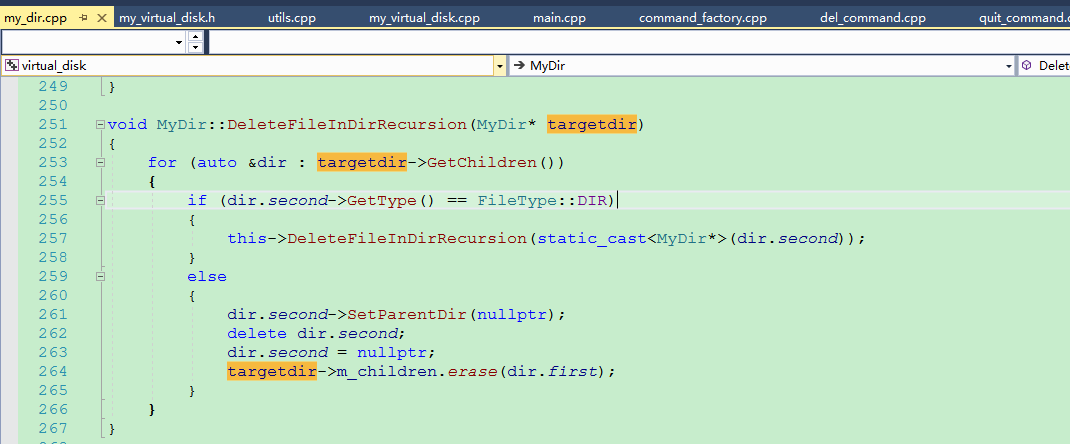
1. 这个是想干啥



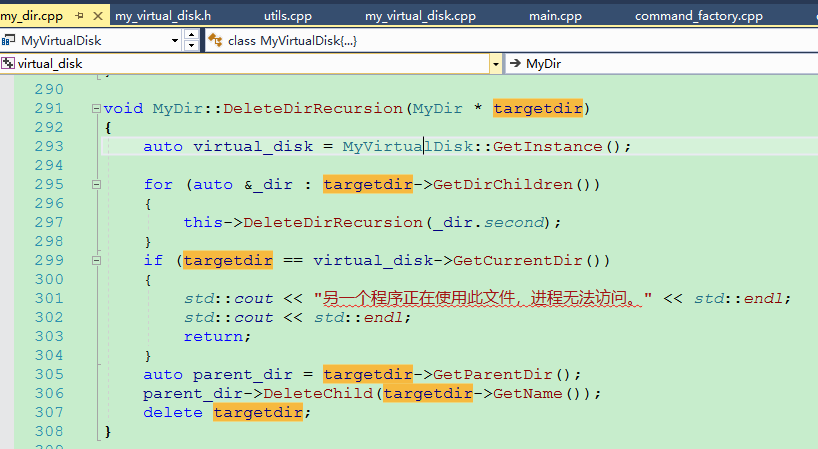
1. 这是做啥的，看起来很奇怪啊



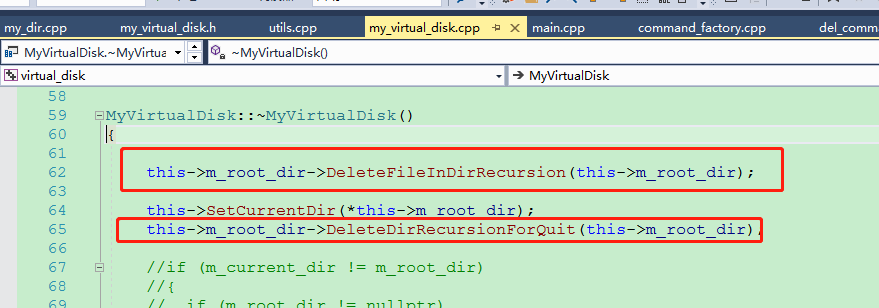
1. 这个函数看起来不太适合做成员函数啊



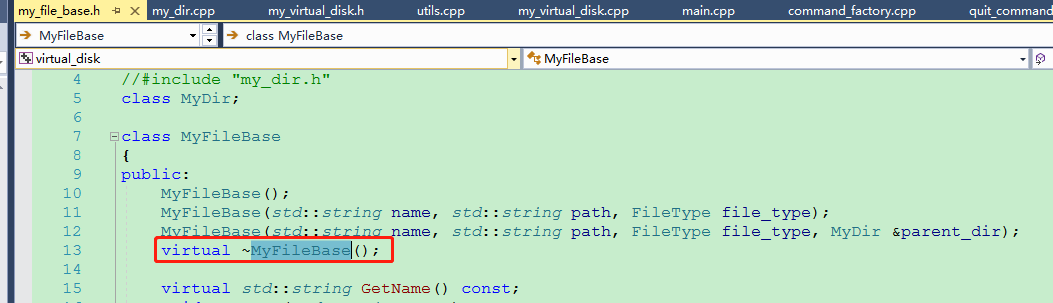
1. 这逻辑是不是有问题，如果有非目录，是不是就删错了



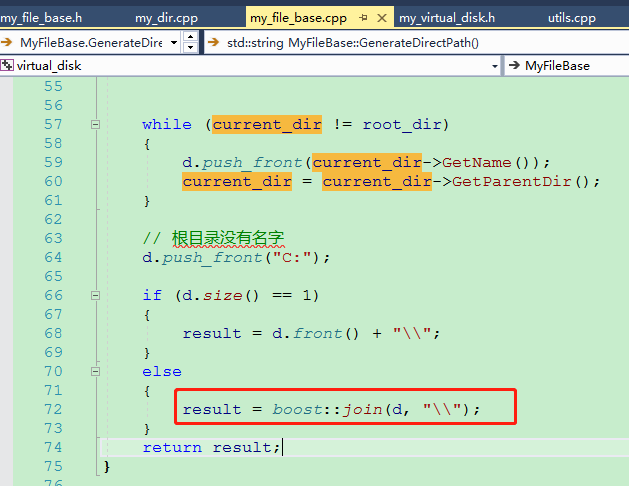
1. 为啥要文件目录分开删



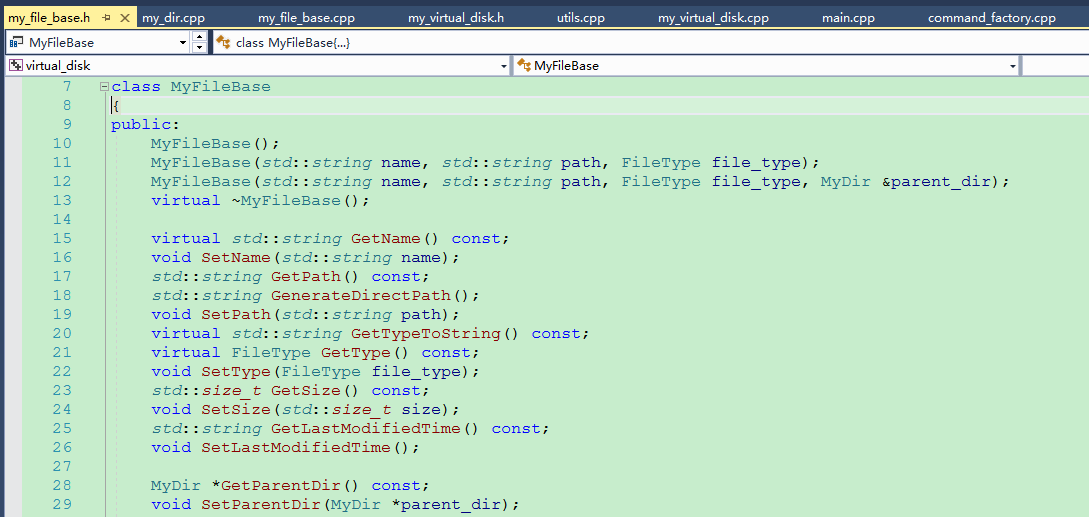
1. 没有面向对象的设计
2. 这个是为啥这么搞



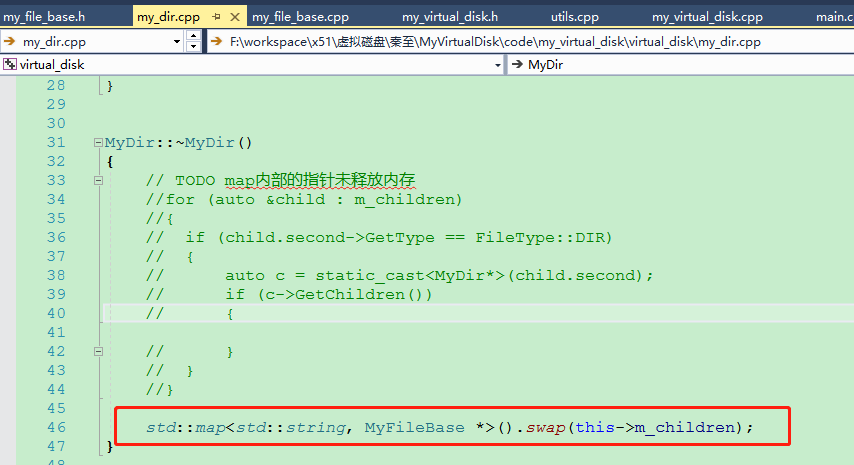
1. 这个是做了啥，会不会少一个”\\”



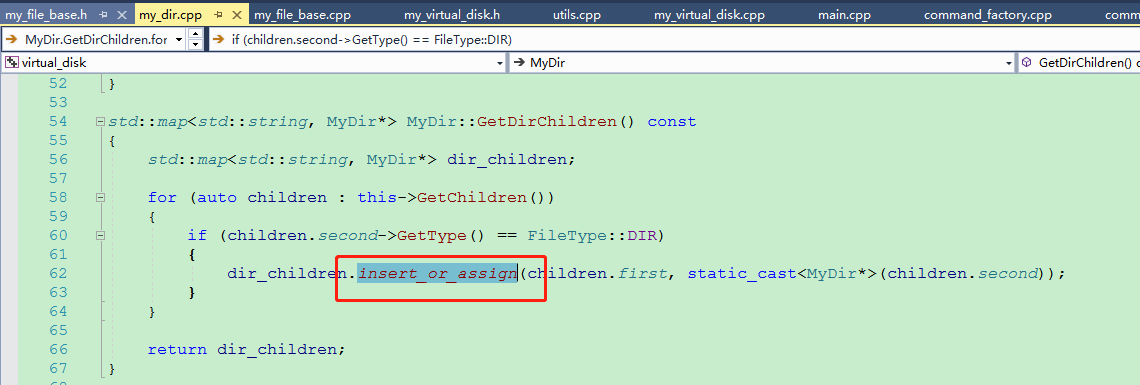
1. 几乎没用多态啊



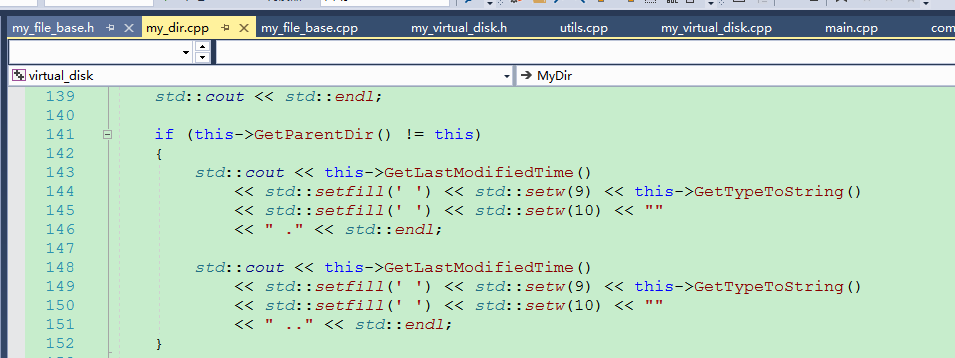
1. 这是啥意思，上面还有个todo



1. 这个和insert有啥区别



1. 这是干啥的

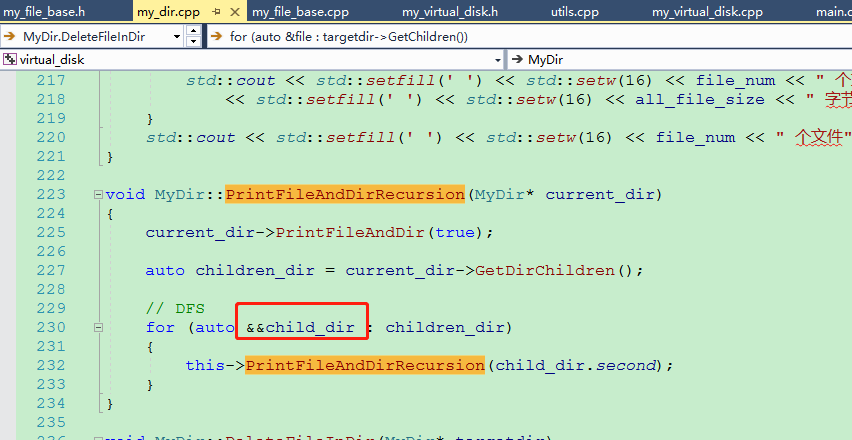


1. 面向对象的实现

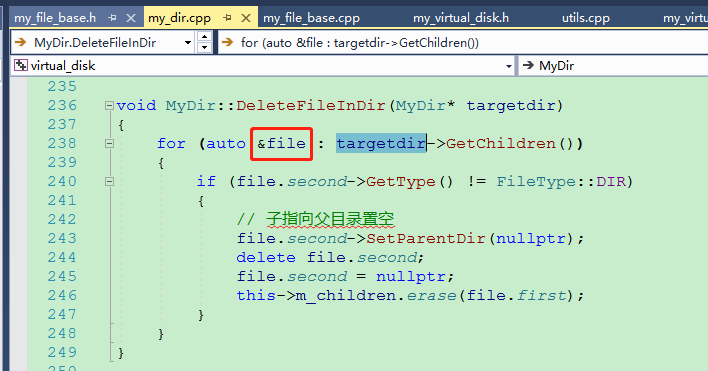




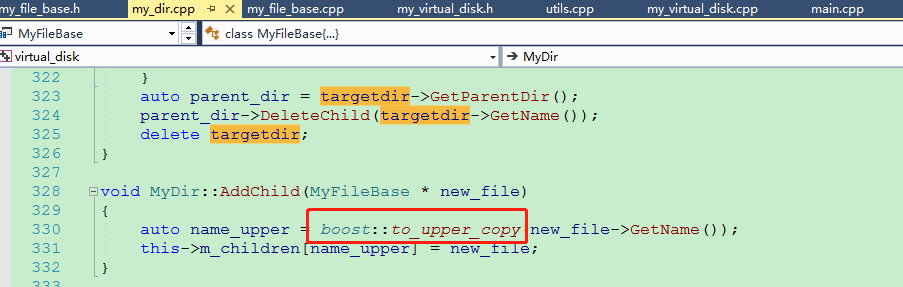
1. 这个是为啥



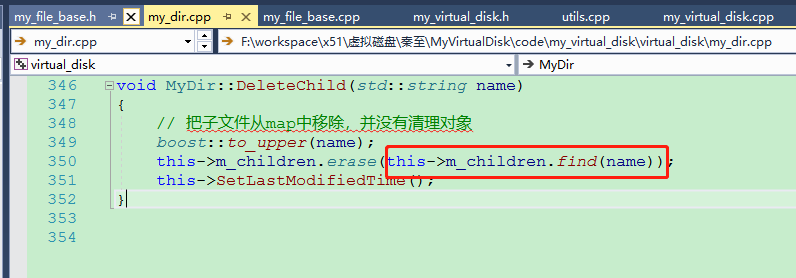
1. 这个是为啥



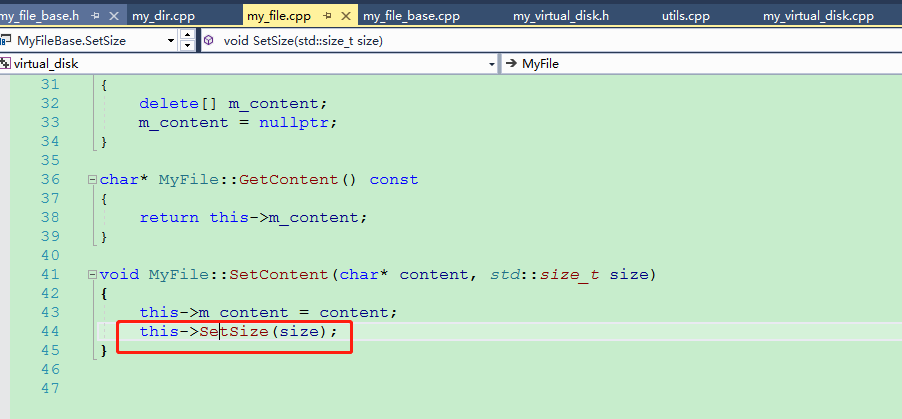
1. 这是为啥，这种是不是应该在输入的时候统一处理



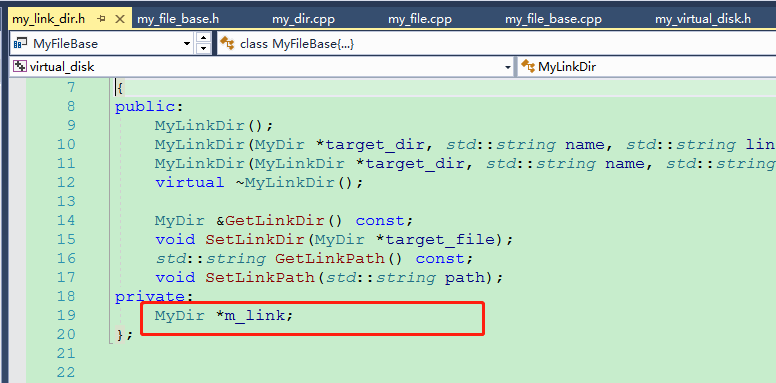
1. 这么写和直接删name有啥区别



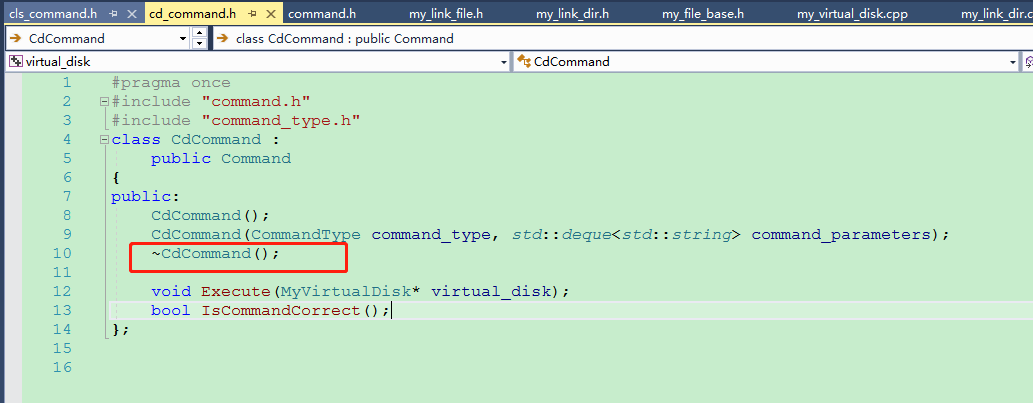
1. 这个为啥放到父类里



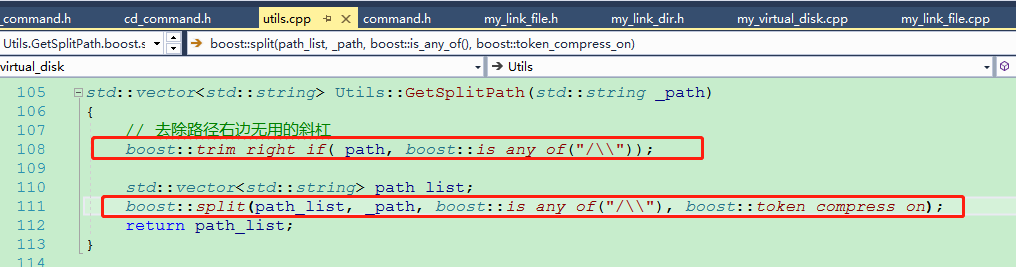
1. 初始化



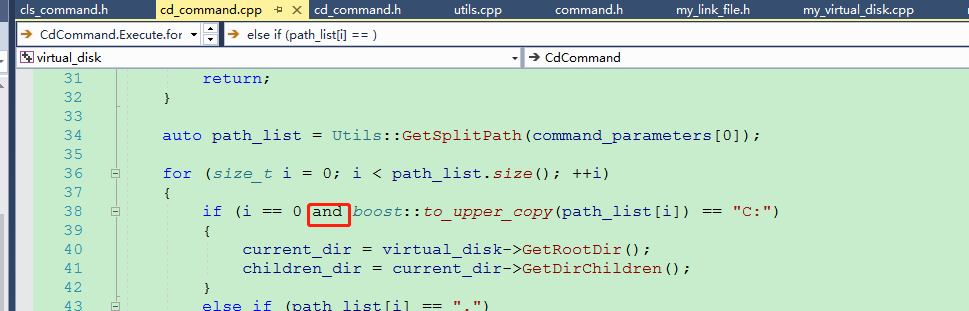
1. 这种也建议加上virtual



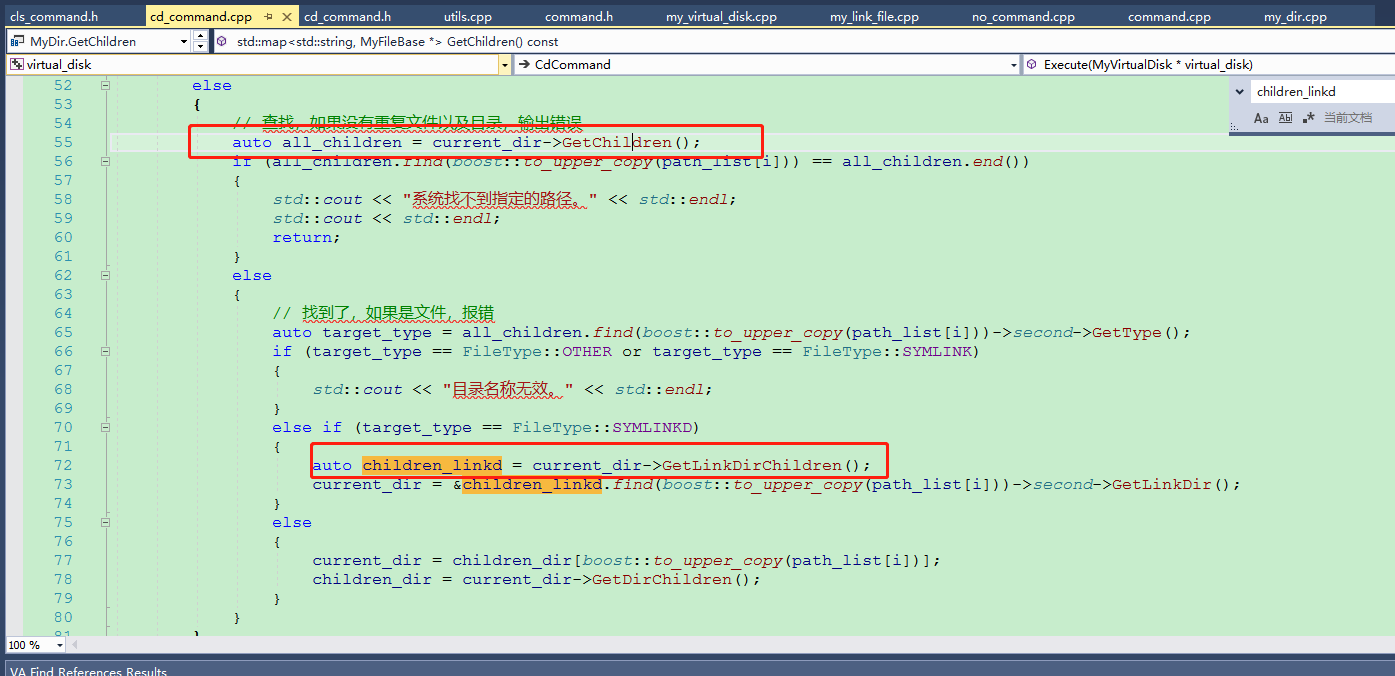
1. 这两个函数是什么原理



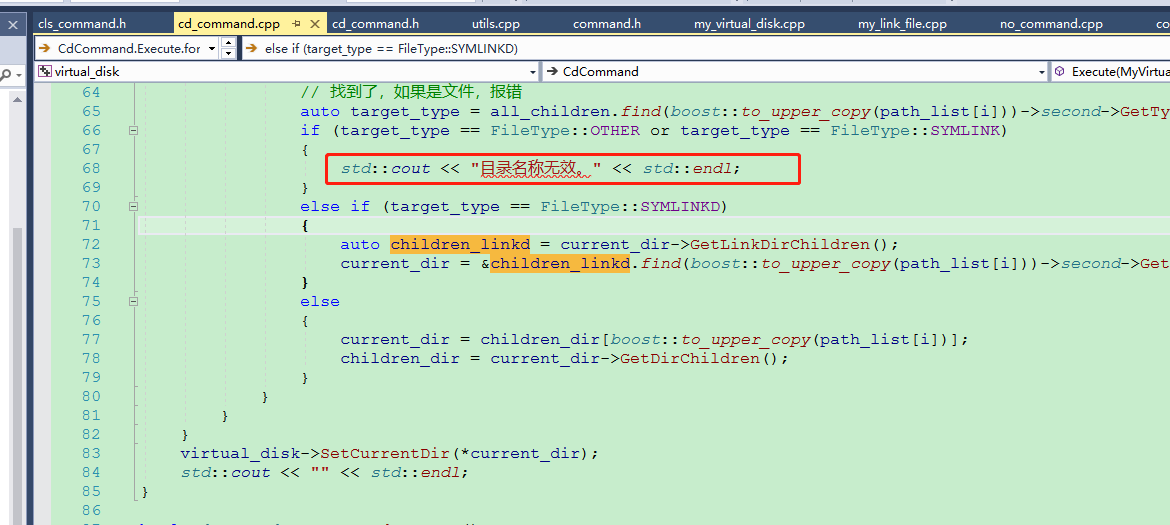
1. 这个啥意思



1. 调用冗余



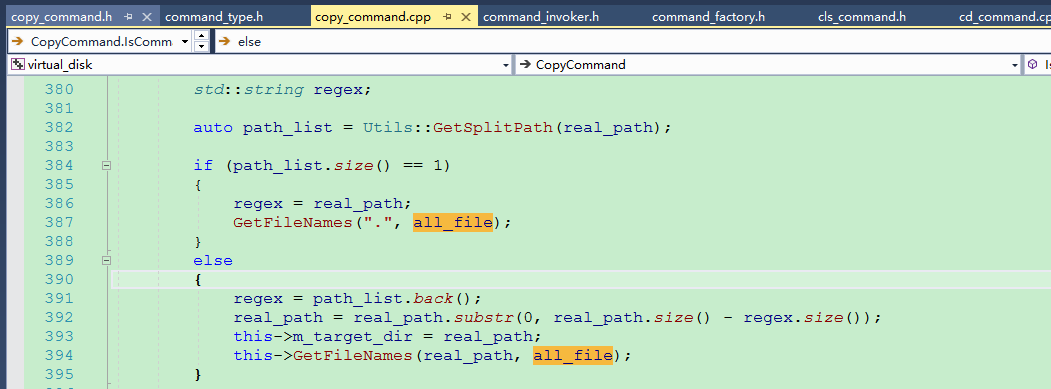
1. 这种不得直接返回吗



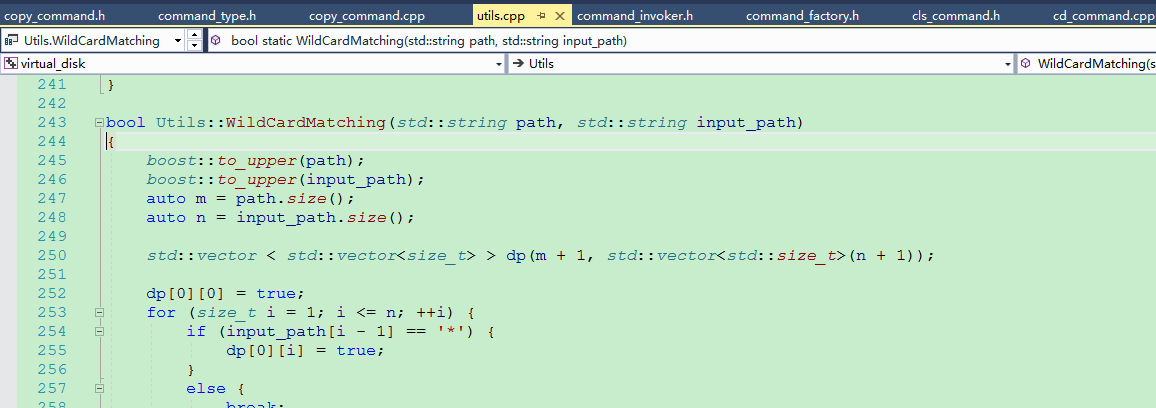
1. 这里设计就不是太合理，这个函数名应该是判断是否正确，不应该有修改里面数据的情况



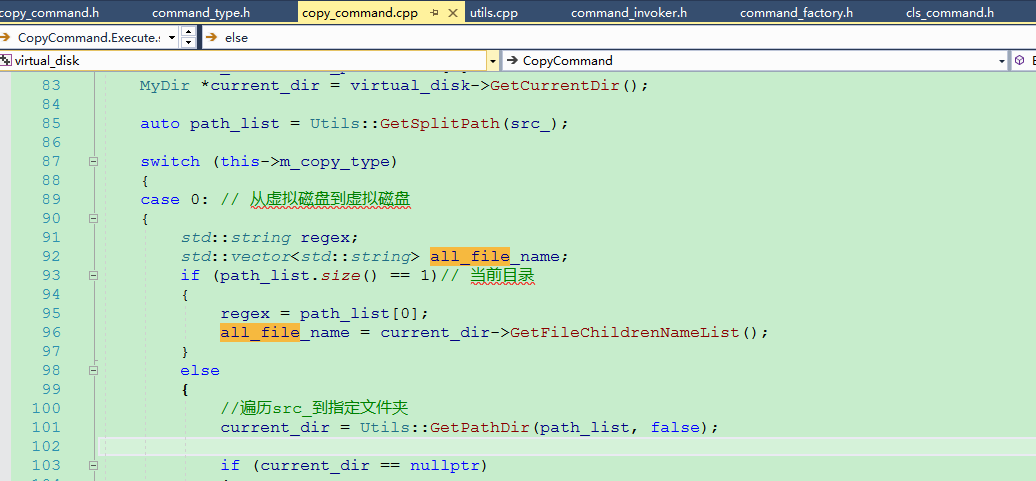
1. 这是啥意思



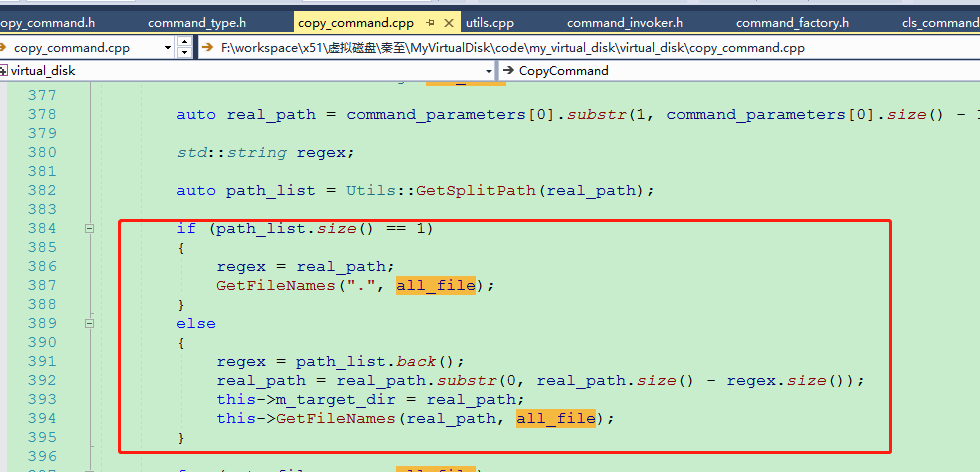
1. 这个是纯自己写的？有啥替代方案吗



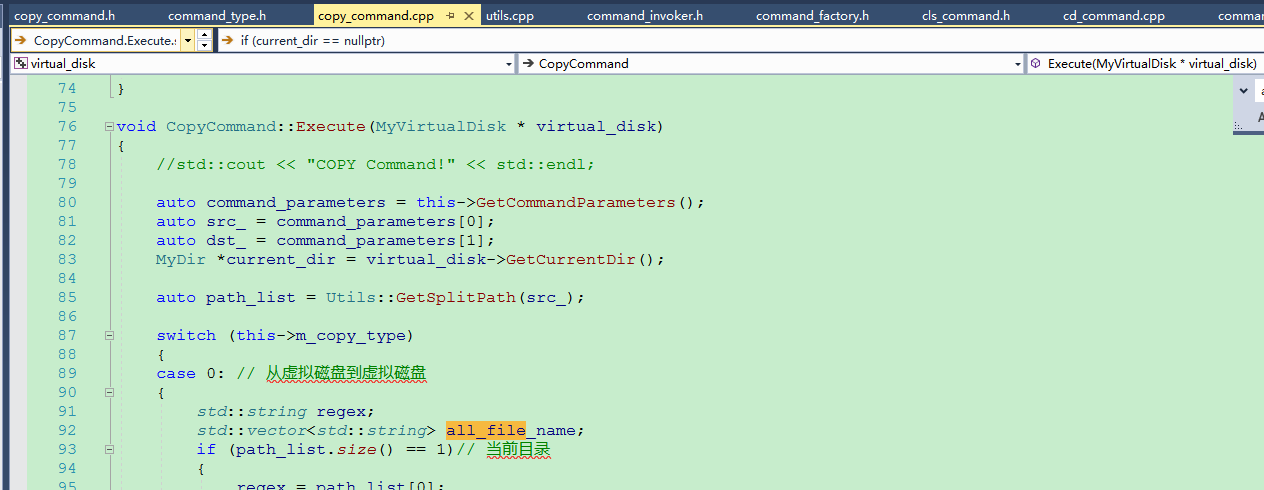
1. 像这种拷贝的流程，就可以把方案写详细，包括输入验证、分支处理，文件读取等等



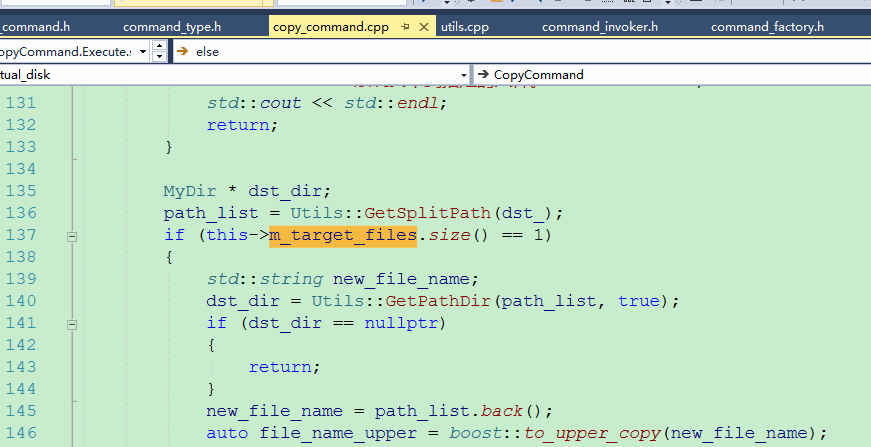
1. 这是啥意思



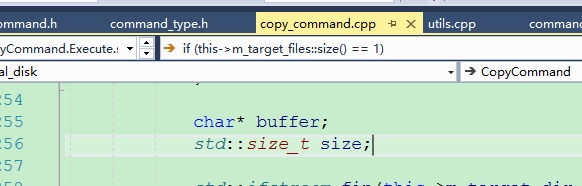
1. 这种大函数，拆成小函数，我记得代码规范有代码函数要求，还有不要用魔数



1. 逻辑嵌入了太多std::cout，这种可以考虑将界面输出、错误信息等作为一个模块
2. 这是啥意思



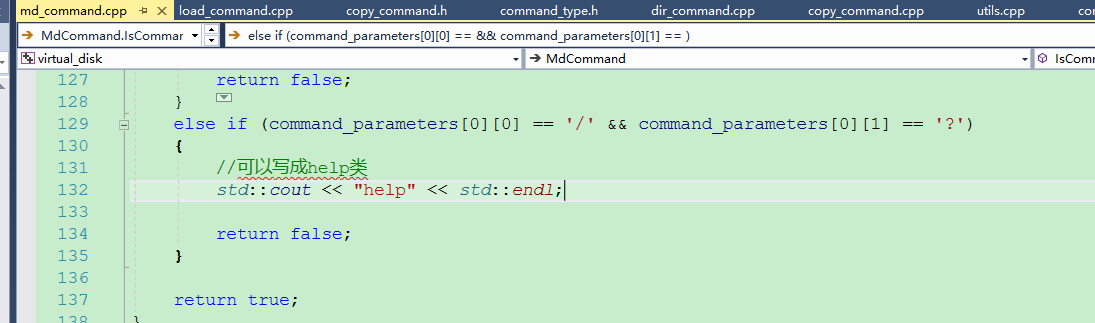
1. 初始化



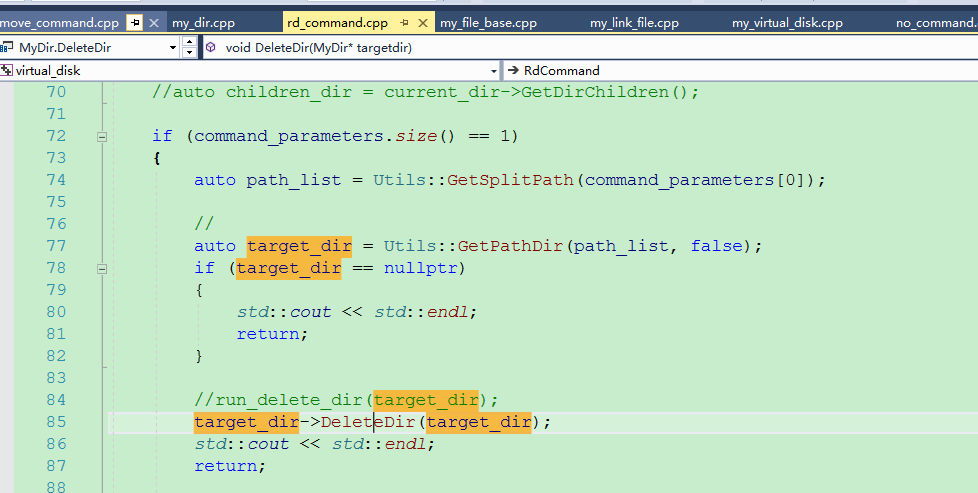
1. 这个是还没搞完？



1. 为啥搞这个



1. 这种如果子目录有文件或者目录，能删吗



1. 连接，所有的命令都考虑了吗
2. 连接，写入文件、加载文件是怎么处理的