Exchange Simulator

516030910108 刘泽宇

一.功能实现及用法：

首先运行server，其次运行trading client，输入IP地址，最后运行monitoring client，输入IP地址。

trading client可以向server发送交易信息，在控制台中输入交易信息时遵循 "new buy(sell) item1(item2) quantity price" 或者 "cancel cancel_order_id" 的格式。

每条order的id是一个数字(从1开始递增)，当trading client向server发送交易信息后，在trading client文件夹中的order.txt文件中会有该条交易信息的记录，记录的格式是“orderid new symbol quantity price"或者"orderid cancel cancel_orderid"。例如："new buy item2 10 7.6" "new sell item1 8 7.5" "cancel 2"

交易信息发送后，trading client和monitoring client都会收到server返回的信息并在控制台中输出。如果是new order，trading client会输出这条消息被server接收，如果可以进行交易，还会输出相应的交易结果（在交易结果中会有对应的orderid）。如果是cancel order，trading client会输出这条请求是否被执行。在monitoring client的控制台中输出的信息和trading client中的基本相同，只是每一条都会输出相应的orderid。

当在trading client中输入"QUIT"之后，整个程序结束。

具体的交易方法遵循价格优先，其次是时间先后。

二.关于代码的解释：

server中有4个map用于储存商品的交易信息，总共两种商品(item1,item2),再加上buy和sell。map的key是price，因而实现了价格优先，buy的map ke由大到小，sell的由小到大。map的value是存放string的vector，vector中共两个元素，第一个是orderid，第二个是quantity。

server中的函数execute_buy_trade和execute_sell_trade分别执行买和卖的order。两个函数的逻辑类似，以execute_buy_trade为例，比较要执行的buy order的价格和相应商品的sell map中第一个pair的key，若小于则直接将这条buy order添加到buy map里，若大于则可以进行交易，交易价格是sell order的价格。重复比较直到buy order的价格小于sell map中第一个pair的key或者buy order被fullfill。若map中的订单被fullfill则将该订单从map中删除。 execute_sell_trade只是把上述大小关系相反。

server中的函数execute_cancel执行cancel order。方法是在4个map中搜索是否有该订单，

若没有则拒绝这个cancel order，若有则取消该条订单。

server的main函数分析从trading client发来的order，根据是new还是cancel，item1或item2的buy或sell相应执行，在发送给trading client和monitoring client的消息时使用了Sleep(),用于暂停使得多条信息能够被分开接收。Sleep() include<window.h> 在其他系统中运行可能会有问题。

trading client的main函数分析用户输入的order，将其转换为FIX message传送给server，并且接受从server返回的FIX message，再进行相应的转换把信息输出到控制台。接受时以"Done"作为多条fill信息传送完毕的标志。

monitoring client的main函数接收从server传送的FIX message，进行相应的转换把信息输出到控制台。