# **Exchange Simulator**

# Business Logic

根据symbol的不同，划分为多个orderbook。

|  |  |
| --- | --- |
| **symbol** | **OrderBook** |
| 600000 | orderbook0 |
| 600001 | orderbook1 |
| … | … |
|  |  |

对于每个orderbook分为sell和buy两种情况，对于每档行情（价格相同），均按照FIFO的方式进行订单执行，如果该档行情没有订单，则将该pricelevel从orderbook中移除。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sell PriceLevels** | | | | |
| PriceLevel | Quantity | OrderList | | |
| 1046 | 1097 | order001 | order002 |  |
| 1045 | 653 | order003 |  |  |
| 1044 | 605 | order005 |  |  |
| 1043 | 184 | order006 |  |  |
| 1042 | 428 | order009 |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Buy PriceLevels** | | | | |
| PriceLevel | Quantity | OrderList | | |
| 1041 | 3865 | order004 | order007 | order0013 |
| 1040 | 13201 | order008 |  |  |
| 1039 | 1738 | order010 |  |  |
| 1038 | 3264 | order011 |  |  |
| 1037 | 1001 | order012 |  |  |

# 订单采用标准FIX表达式

例如

8=FIX.4.2;35=D;11=order103;55=600000;54=1;38=3500;44=10.44;40=2;59=3

具体的FixTag如下所示

*// FIX tag*static constexpr uint32\_t FixVersion = 8;  
static constexpr uint32\_t OrderId = 11;  
static constexpr uint32\_t MsgType = 35;  
static constexpr uint32\_t Symbol = 55;  
static constexpr uint32\_t Price = 44;  
static constexpr uint32\_t Quantity = 38;  
static constexpr uint32\_t Side = 54;  
static constexpr uint32\_t OrderTypeTag = 40;  
static constexpr uint32\_t TimeInForceTag = 59;

# 支持多种订单模式

以7月24号的浦发银行为例，以该orderbook，描述支持的订单类型。



## Limit GTC Order

1. 如果没有对手方，或者有价差，则加入已有的pricelevel或者新建一个pricelevel

通过该方式构建如下的orderbook：

|  |
| --- |
| OrderBook for symbol: 600000  =============SELL==============  Sell 1 Price: 10.46 Quantity: 1097  Sell 2 Price: 10.45 Quantity: 653  Sell 3 Price: 10.44 Quantity: 605  Sell 4 Price: 10.43 Quantity: 184  Sell 5 Price: 10.42 Quantity: 428  =============BUY==============  Buy 1 Price: 10.41 Quantity: 3865  Buy 2 Price: 10.4 Quantity: 13201  Buy 3 Price: 10.39 Quantity: 1738  Buy 4 Price: 10.38 Quantity: 3264  Buy 5 Price: 10.37 Quantity: 1001 |

1. 如果有满足价格的对手方，则根据价格进行订单消化，剩余的继续留在orderbook上。

|  |
| --- |
| 8=FIX.4.2;35=D;11=order101;55=600000;54=1;38=3500;44=10.44;40=2;59=0  该限价单为以价格10.44买入3500单。根据orderbook，10.42，10.43和10.44的quantity将被match，剩余的quantity继续挂在orderbook上，价格为10.44  2283 = 3500 - (605+184+428)  =============SELL==============  Sell 1 Price: 10.46 Quantity: 1097  Sell 2 Price: 10.45 Quantity: 653  =============BUY==============  Buy 1 Price: 10.44 Quantity: 2283  Buy 2 Price: 10.41 Quantity: 3865  Buy 3 Price: 10.4 Quantity: 13201  Buy 4 Price: 10.39 Quantity: 1738  Buy 5 Price: 10.38 Quantity: 3264 |
| 8=FIX.4.2;35=D;11=order102;55=600000;54=2;38=3500;44=10.40;40=2;59=0  该限价单为以价格10.40卖出3500单。根据orderbook，10.41的quantity为3865大于3500，因此以该价格执行，并根据先后顺序match整个order004，以及部分order007  365 = 3865-3500  OrderBook for symbol: 600000  =============SELL==============  Sell 1 Price: 10.46 Quantity: 1097  Sell 2 Price: 10.45 Quantity: 653  Sell 3 Price: 10.44 Quantity: 605  Sell 4 Price: 10.43 Quantity: 184  Sell 5 Price: 10.42 Quantity: 428  =============BUY==============  Buy 1 Price: 10.41 Quantity: 365  Buy 2 Price: 10.4 Quantity: 13201  Buy 3 Price: 10.39 Quantity: 1738  Buy 4 Price: 10.38 Quantity: 3264  Buy 5 Price: 10.37 Quantity: 1001 |

## Limit IOC Order

如果没有对手方，或者有对手方，根据价格进行部分成交并撤回剩余的quantity。

|  |
| --- |
| 8=FIX.4.2;35=D;11=order103;55=600000;54=1;38=3500;44=10.44;40=2;59=3  该IOC为以价格10.44卖出3500单，根据orderbook，将成交10.42，10.43，10.44的订单，并撤回剩下的订单，成交了(605+184+428)。  OrderBook for symbol: 600000  =============SELL==============  Sell 1 Price: 10.46 Quantity: 1097  Sell 2 Price: 10.45 Quantity: 653  =============BUY==============  Buy 1 Price: 10.41 Quantity: 3865  Buy 2 Price: 10.4 Quantity: 13201  Buy 3 Price: 10.39 Quantity: 1738  Buy 4 Price: 10.38 Quantity: 3264  Buy 5 Price: 10.37 Quantity: 1001 |
| 在上一个orderbook中继续，  8=FIX.4.2;35=D;11=order104;55=600000;54=1;38=3500;44=10.42;40=2;59=3  该IOC为以价格10.42卖出3500单，根据当前orderbook，没有可以成交的订单，立即撤回，对市场没有影响。  OrderBook for symbol: 600000  =============SELL==============  Sell 1 Price: 10.46 Quantity: 1097  Sell 2 Price: 10.45 Quantity: 653  =============BUY==============  Buy 1 Price: 10.41 Quantity: 3865  Buy 2 Price: 10.4 Quantity: 13201  Buy 3 Price: 10.39 Quantity: 1738  Buy 4 Price: 10.38 Quantity: 3264  Buy 5 Price: 10.37 Quantity: 1001 |

## Limit FOK Order

|  |
| --- |
| 8=FIX.4.2;35=D;11=order105;55=600000;54=2;38=3500;44=10.40;40=2;59=4  该IOC为以价格10.40卖出3500单，根据orderbook，将成交10.42，10.43，10.44的订单，并撤回剩下的订单。  OrderBook for symbol: 600000  =============SELL==============  Sell 1 Price: 10.46 Quantity: 1097  Sell 2 Price: 10.45 Quantity: 653  Sell 3 Price: 10.44 Quantity: 605  Sell 4 Price: 10.43 Quantity: 184  Sell 5 Price: 10.42 Quantity: 428  =============BUY==============  Buy 1 Price: 10.41 Quantity: 365  Buy 2 Price: 10.4 Quantity: 13201  Buy 3 Price: 10.39 Quantity: 1738  Buy 4 Price: 10.38 Quantity: 3264  Buy 5 Price: 10.37 Quantity: 1001 |
| 8=FIX.4.2;35=D;11=order106;55=600000;54=2;38=20000;44=10.40;40=2;59=4  该FOK为以价格10.40卖出20000单，根据当前orderbook，不能够fullfill成交，立即kill  OrderBook for symbol: 600000  =============SELL==============  Sell 1 Price: 10.46 Quantity: 1097  Sell 2 Price: 10.45 Quantity: 653  Sell 3 Price: 10.44 Quantity: 605  Sell 4 Price: 10.43 Quantity: 184  Sell 5 Price: 10.42 Quantity: 428  =============BUY==============  Buy 1 Price: 10.41 Quantity: 3865  Buy 2 Price: 10.4 Quantity: 13201  Buy 3 Price: 10.39 Quantity: 1738  Buy 4 Price: 10.38 Quantity: 3264  Buy 5 Price: 10.37 Quantity: 1001 |

## Market Order

|  |
| --- |
| 8=FIX.4.2;35=D;11=order107;55=600000;54=1;38=2000;40=1;59=0  该市价单将sell的订单按照成交价进行执行，直到quantity总和满足。  OrderBook for symbol: 600000  =============SELL==============  Sell 1 Price: 10.46 Quantity: 967  =============BUY==============  Buy 1 Price: 10.41 Quantity: 3865  Buy 2 Price: 10.4 Quantity: 13201  Buy 3 Price: 10.39 Quantity: 1738  Buy 4 Price: 10.38 Quantity: 3264  Buy 5 Price: 10.37 Quantity: 1001 |
| 8=FIX.4.2;35=D;11=order108;55=600000;54=1;38=4000;40=1;59=0  当市价单的quantity = 4000比所以sell的quantity总和还多，因此剩余转为最优buy price order。  OrderBook for symbol: 600000  =============SELL==============  =============BUY==============  Buy 1 Price: 10.41 Quantity: 4898  Buy 2 Price: 10.4 Quantity: 13201  Buy 3 Price: 10.39 Quantity: 1738  Buy 4 Price: 10.38 Quantity: 3264  Buy 5 Price: 10.37 Quantity: 1001 |

# 支持多个orderbook

|  |
| --- |
| OrderBook for symbol: 600000  =============SELL==============  Sell 1 Price: 10.46 Quantity: 1097  Sell 2 Price: 10.45 Quantity: 653  Sell 3 Price: 10.44 Quantity: 605  Sell 4 Price: 10.43 Quantity: 184  Sell 5 Price: 10.42 Quantity: 428  =============BUY==============  Buy 1 Price: 10.41 Quantity: 3865  Buy 2 Price: 10.4 Quantity: 13201  Buy 3 Price: 10.39 Quantity: 1738  Buy 4 Price: 10.38 Quantity: 3264  Buy 5 Price: 10.37 Quantity: 1001  OrderBook for symbol: 600001  =============SELL==============  Sell 1 Price: 10.5 Quantity: 97  =============BUY==============  Buy 1 Price: 10.4 Quantity: 97 |

# Code Structure:

├── main.cpp

├── src

│   ├── model

│   │   ├── Order.cpp

│   │   ├── Order.h

│   │   ├── OrderBook.cpp

│   │   ├── OrderBook.h

│   │   ├── PriceLevel.cpp

│   │   └── PriceLevel.h

│   ├── service

│   │   ├── ExecutionService.cpp

│   │   ├── ExecutionService.h

│   │   ├── OrderBookService.cpp

│   │   ├── OrderBookService.h

│   │   ├── QuotationService.cpp

│   │   └── QuotationService.h

│   └── utils

│   ├── Constants.cpp

│   ├── Constants.h

│   ├── Utils.cpp

│   └── Utils.h

└── test

├── CMakeLists.txt

└── TestCases.cpp

model 文件夹中主要为Order，OrderBook以及PriceLevel的定义。

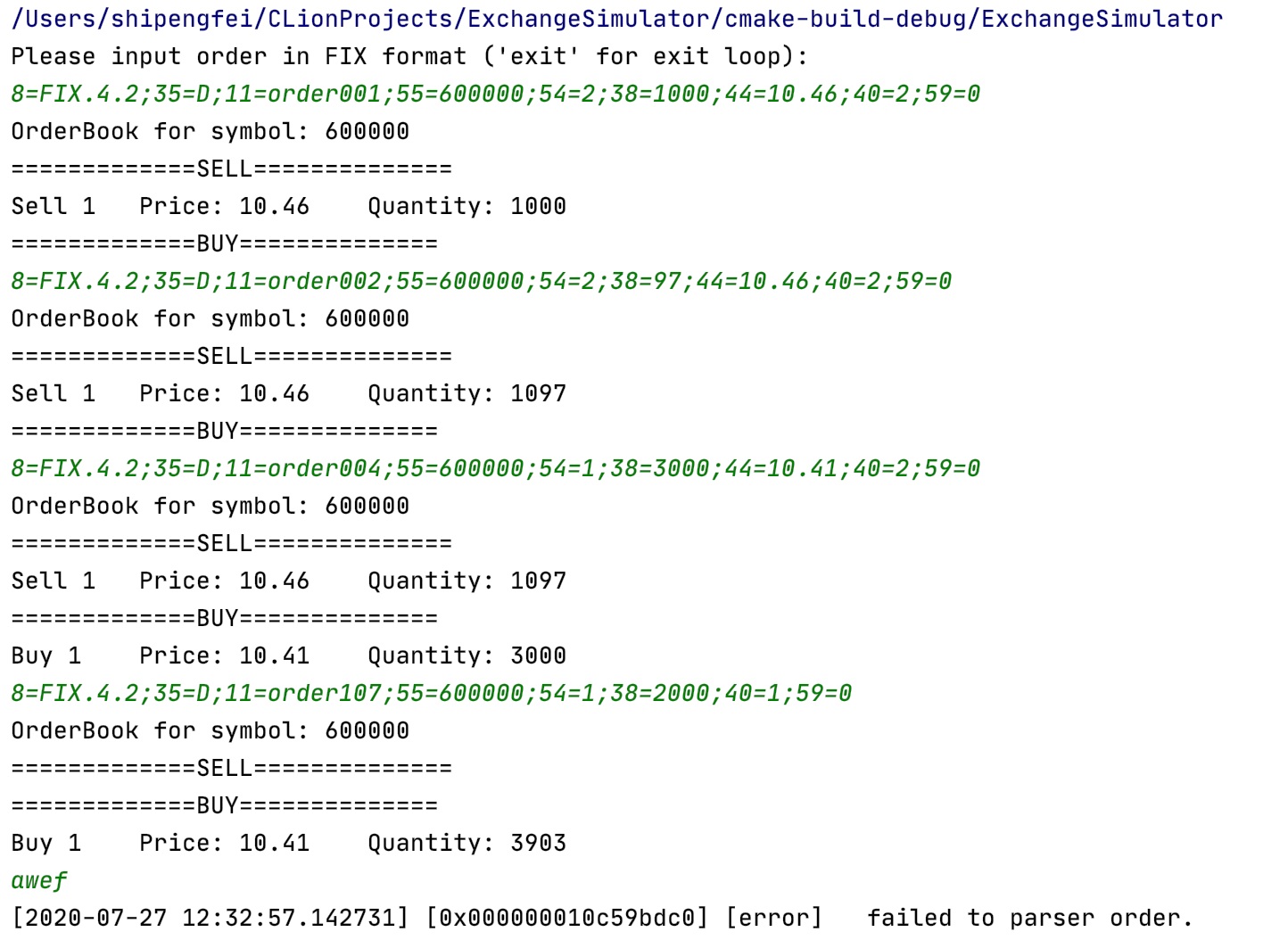
service文件夹中按照业务逻辑的不同划分为3个模块，分别为OrderBookService，QuotationService以及ExecutionService，模块均采用单例模式。

* ExecutionService负责对于订单的解析以及路由。
* OrderBookService负责对于order的执行。
* QuotationService 负责对于行情数据的获取。

utils文件夹中主要为各种辅助方程，以及预先定义的各个FixTag。

Cmake编译并

运行ExchangeSimulator即可，交互式输入，并对简单的异常情况进行处理。



运行Test将会把上述多个情况下的test cases运行。