

Equity Order Matching



In stock trading world, order book is the place where all active orders (both Buy and Sell) are maintained in certain priority order for the purposes are matching buy and sell orders.

Design a component Matcher which performs matching between Buy and Sell sides of the order book. Matcher receives an input stream consisting of Buy and Sell orders and with each Match command matcher component attempts to match any outstanding orders at that point of time. Matcher must ensure that buy order with highest prices are matched with sell order with lowest price. In case of multiple orders have the same price on given side (Buy/Sell), and then matcher should pick order on first come first serve basis. A Buy order for a symbol X at price P can be matched with any Sell order on the same symbol X at prices less than or equal to P. Thus, a new Buy or Sell order may be matched with one or more of outstanding Sell or Buy orders.

Orders (both Buy/Sell Sides) can be of different types, for the purposes of this challenge please consider following order types

OrderType Details:

Market - A Market order is an order to buy or sell a symbol that must be matched at best price on the other side. Market Buy - amount x at current lowest selling price Market Sell - amount x at current highest buying price

Limit - A Limit order is an order to buy or sell a symbol that must be matched at a specified price. Limit Buy - amount x at price $\leq p$, eventually Limit Sell - amount x at price $\geq p$, eventually

IOC - An immediate or cancel order (IOC) is an order to buy or sell a symbol that must be matched in the next matching cycle, and any portion of the order that cannot be matched will be automatically cancelled. Immediate Or Cancel(IOC) Limit Buy - amount x at price $\leq p$, now Immediate Or Cancel(IOC) Limit Sell - amount x at price $\geq p$, now

Input Format

Matcher reads following commands from stdin and responds by writing to stdout. It should terminate when it reaches end of file. It should read one command at a time and process it completely before proceeding to the next command. Any sort of look-ahead is not allowed and any submission utilizing look-ahead will not be considered for evaluation.

An input command may contain following fields:

Action: N(New), A(Amend), X(Cancel), M(Match) or Q(Query)

Order ID: an integer in range $[0, 2^{63} - 1]$

Timestamp: a long data type, typically milliseconds since epoch

Symbol: Varying length string containing only alphabets

OrderType: M (Market), L (Limit) or I(IOC)

Side: B (Buy) or S (Sell)

Price: Float, 0 if orderType is M (Market)

Quantity: an integer in range $[0, 2^{63} - 1]$

Typical input command will have following structure:

<Action>, <Order ID>, <timestamp>, <symbol>, <orderType>, <side>, <price>, <quantity> However, some of the input commands may have their own format, please see below further details.

Action Details:

N(New) - This command means that a new order is being requested to be entered into the matching book. Matcher should reject the incoming order in case of duplicate order Id or invalid input fields. Matcher should output "<incoming orderid> - Accept" or "<incoming orderid> - Reject Error Code".

Input Command format:

<Action>, <Order ID>, <timestamp>, <symbol>, <orderType>, <side>, <price>, <quantity>

E.g. N, 1, 0000001, XYZ, L, B, 104.53, 100

A(Amend) - This command means an existing order is being requested to be updated per details in this command. Only Quantity or Price can be updated, update in any other field should result into error code. In case of quantity amend down, amended order should not lose existing priority in the matching book. In all other amend requests, priority of amended order in matching book may vary accordingly and needs to pick latest timestamp. Matcher should output "<incoming orderid> - AmendAccept" or "<incoming orderid> - AmendReject - Error Code".

Input Command format:

<Action>, <Order ID>, <timestamp>, <symbol>, <orderType>, <side>, <price>, <quantity>

E.g. A, 1, 0000001, XYZ, L, B, 103.53, 150

X(Cancel) - This command means an existing order is being requested to be cancelled. Matcher should output "<incoming orderid> - CancelAccept" or "<incoming orderid> - CancelReject - Error Code".

Input Command format:

<Action>, <Order ID>, <timestamp>

E.g. - X, 1, 0000001

M(Match) - This command means that the existing orders in the matching book so far can be matched now. <Symbol> is an optional parameter if specified would mean matching should be done only for the specified symbol else it can be done for all the securities in alphabetical order. Matcher should output all resulting matches. In case of no matches, there shouldn't be any output. *Input Command format:*

<Action>, <timestamp> [, <Symbol>] E.g. M, 000001 E.g. M, 000002, XYZ

Q(Query) - This command is a request to print the state of matching book. <Symbol> and <timestamp> are optional parameters.

If only symbol is specified then matcher should print state of the matching book for the given symbol.

If only timestamp is specified then matcher should print state of the matching book(s) at the given timestamp for all securities in alphabetical order.

If both symbol and timestamp are specified then matcher should print state of the matching book for the given symbol and timestamp.

If no optional parameters are specified, then matcher should print state of the matching book(s) as of now for all securities in alphabetical order.

Matcher should output at max top 5 buy and sell entries as the state of the matching book for the given/available security(s) in the alphabetical order.

Please note that as orders could be amended, special care should be taken return order details as of given timestamp.

Input Command format:

<Action> [, <timestamp>] [,<Symbol>]

E.g. Q

E.g. Q,000001

E.g. Q,XYZ

E.g. Q,XYZ,0000001

E.g. Q,0000001,XYZ

Error Codes:

303 - Invalid order details

404 - Order does not exist

101 - Invalid amendment details

450 - Action not supported

Constraints

Price and quantity can not be negative.

Any input command having lower timestamp than previous command should be rejected, except query command.

Please ignore empty/white spaces in the input fields.

Orders that are fully matched/cancelled do not further participate in matching.

Output Format

N(New) – Matcher should output "<incoming orderid> - Accept" or "<incoming orderid> - Reject - <Error Code>".

1) Input - N, 2, 0000002, XYZ, L, B, 104.53, 100

Output - "2 - Accept"

2) Input - N, 3, 0000002, XYZ, L, B, 104.53, 100.3

Output - "3 - Reject - 303 - Invalid order details"

In the above example, Order 3 is rejected as quantity (100.3) is not an integer.

A(Amend) - Matcher should output "<incoming orderid> - AmendAccept" or "<incoming orderid> - AmendReject - <Error Code>".

1) Input - A, 2, 0000001, XYZ, L, B, 103.53, 150

Output - "2 - AmendAccept"

2) Input - A,2, 0000001, XYZ, L, B, 103.53, 150.4

Output - "2 - AmendReject - 101 - Invalid amendment details"

In the above example, Amend Order 2 is rejected as quantity (150.4) being amended is invalid

X(Cancel) - Matcher should output "<incoming orderid> - CancelAccept" or "<incoming orderid> - CancelReject - <Error Code>".

1) Input - X, 1, 0000001

Output - "1 - CancelAccept"

2) Input:

X,1, 0000001

X,1, 0000001

Output:

"1 - CancelAccept"

"1 - CancelReject - 404 - Order does not exist"

In scenarios where order is fully matched/already cancelled, new cancel requests should be rejected as in the second example.

M(Match) – Matcher should output all resulting matches in the below format. In case of no matches, there shouldn't be any output. Please follow output format strictly especially with spaces and commas.

Output format : <symbol>|<orderid>,<orderType>,<Buy Qty matched>,<Buy Price matched>|<Sell Price matched>,<Sell Qty matched>,<OrderType>,<orderId>

Sample Output:

Before match command:

N, 1, 0000001, ALN, L, B, 60.9, 100

N, 11, 0000002, XYZ, L, B, 60.9, 200

N, 110, 0000003, XYZ, L, B, 60.9, 100

N, 112, 0000003, XYZ, L, B, 60.9, 120

N, 10, 0000006, ALN, L, S, 60.9, 100

1) Input - M, 00010

Output:

ALN|1,L,100,60.9|60.9,100,L,10

XYZ|11,L,100,60.9|60.9,100,L,110

XYZ|11,L,100,60.9|60.9,100,L,112

2) Input - M,00010,ALN.NS

Output:

ALN|1,L,100,60.9|60.9,100,L,10

Q(Query) – Matcher should output state of the matching book per details in the input format. If there is no output for a given query no output should be printed. Please follow output format strictly especially with spaces and commas.

Output format : <symbol>|<orderid>,<orderType>,<Buy Qty >,<Buy Price>|<Sell Price>,<Sell Qty>,<OrderType>,<orderId>

Matcher should output at max top 5 buy and sell entries as the state of the matching book for the given/available security(s) in the alphabetical order.

Before the query command:

Prestate:

N, 1, 0000001, ALN, L, B, 60.9, 100

N, 13, 0000002, ALN, L, B, 60.9, 100

N, 10, 0000003, ALN, L, S, 60.9, 100

N, 12, 0000004, ALN, L, S, 60.9, 100

1) Input – Q,ALN

Output:

ALN|1,L,100,60.9|60.9,100,L,10

ALN|13,L,100,60.9|60.9,100,L,12

In this example matching book has only 4 orders, two on each side.

Prestate:

N, 10, 0000001, ALB, L, S, 60.9, 100

N, 14, 0000002, ALB, L, S, 62.9, 101

N, 16, 0000003, ALB, L, S, 63.9, 102

N, 18, 0000004, ALB, L, S, 64.9, 103

N, 20, 0000005, ALB, L, S, 65.9, 104

2) Input – Q,ALB

Output:

ALB||60.9,100,L,10

ALB||62.9,101,L,14

ALB||63.9,102,L,16

ALB||64.9,103,L,18

ALB||65.9,104,L,20

In the above example there are only sell orders. If there are no corresponding buy/sell orders that section should be left blank.

Prestate:

4 Sell orders of ALB, 5 buy orders and 4 sell orders of ALN

3) Input – Q,001020

Output:

ALB||60.9,100,L,17

ALB||62.9,101,L,18

ALB||63.9,102,L,19

ALN|1,L,100,60.9|60.9,100,L,10

ALN|3,L,100,60.9|60.9,100,L,12

ALN|5,L,100,60.9|60.9,100,L,14

ALN|7,L,100,60.9|60.9,100,L,15

ALN|16,L,100,60.9|

In the above example, system has only ALB and ALN securities. Matcher is printing state of the matching book(s) in alphabetical order, first ALB followed by ALN. Matcher should output at max top 5 buy and sell entries as the state of the matching book and hence only upto 5 entries(buy and sell) were printed for ALN

Prestate for 4) and 5)

N, 17, 0000005, ALB, L, S, 60.9, 100

N, 18, 0000007, ALB, L, S, 62.9, 101

N, 19, 0000020, ALB, L, S, 63.9, 102

4) Input – Q,ALB,000020

Output:

ALB||60.9,100,L,17

ALB||62.9,101,L,18

ALB||63.9,102,L,19

5) Input – Q,000020,ALB

Output:

ALB||60.9,100,L,17

ALB||62.9,101,L,18

ALB||63.9,102,L,19

Please note Query timestamp is inclusive as shown in the example 4) and 5)

Incase of Market orders, price should be printed as 0.00

Sample Input 0

```
12
N,1,0000001,AB,L,B,104.53,100
N,2,0000002,AB,L,S,105.53,100
N,3,0000003,AB,L,B,104.53,90
M,0000004
N,4,0000005,AB,L,S,104.43,80
A,2,0000006,AB,L,S,104.42,100
Q
M,0000008
N,5,0000009,AB,L,S,105.53,120
X,3,0000010
N,6,0000011,XYZ,L,B,1214.82,2568
Q
```

Sample Output 0

```
1 - Accept
2 - Accept
3 - Accept
4 - Accept
2 - AmendAccept
AB|1,L,100,104.53|104.42,100,L,2
AB|3,L,90,104.53|104.43,80,L,4
AB|1,L,100,104.42|104.42,100,L,2
AB|3,L,80,104.43|104.43,80,L,4
5 - Accept
3 - CancelAccept
6 - Accept
AB||105.53,120,L,5
XYZ|6,L,2568,1214.82|
```