For this project, you need to understand how trading works:

# HOW TRADING WORKS??

People can buy and sell a stock (like RELIANCE) / commodity (like OIL) / currency (like USDINR) etc on an exchange. To buy and sell, they place orders on the exchange.

For example, to Buy Reliance stock, a user can place an order with the following fields:

SYMBOL: RELIANCE, QUANTITY: 10, PRICE: 800, SIDE: BUY.

A BUY order is also called a "bid" and a SELL order is called an "ask".

An exchange acts as a matching engine, which matches these Buy/Sell Orders. For example, following are the orders, which arrive at exchange (assuming t => time):

at t=0, (user A placed an order) SYMBOL:RELIANCE, QUANTITY:10, PRICE:800, SIDE:BUY

at t=1, (user B placed an order) SYMBOL:GAIL, QUANTITY:5, PRICE:500, SIDE:SELL

at t=2, (user B placed an order) SYMBOL:RELIANCE, QUANTITY:10, PRICE:805, SIDE:SELL

At this point, user A and B both have an order for RELIANCE, but at different prices which cannot be matched. So, the exchange will show its order book as:

BUY: QTY: 10 PRICE: 800

SELL: QTY: 10 PRICE: 805

Now, consider another order:

at t=5, (user C placed an order) SYMBOL: RELIANCE, QUANTITY:1, PRICE: 800, SIDE: SELL

Since the prices are matching here, exchange will match these and inform both the parties that a trade has occurred. The order book of the exchange is also shown below:

at t=5, TRADE, SYMBOL: RELIANCE, QUANTITY: 1, PRICE: 800

BUY: QTY: 9 PRICE: 800

SELL: QTY: 10 PRICE: 805

# PROJECT DETAILS:

Exchange publishes the trading details to the trader by way of a tick-by-tick feed, which contains the following messages:

* New Order (type: N) - whenever someone sends a new order. Format is:

<type>, SYMBOL, SIDE(buy/sell), QTY, PRICE

  For example, the above order at t=0 will look as:

  N, RELIANCE, B, 10, 800

* Modification Order (type: M) - this is sent when someone modifies his order. Format is:

<type>, SYMBOL, SIDE(Buy/Sell), OLD-QUANTITY, NEW-QUANTITY, OLD-PRICE, NEW-PRICE

For example, if the quantity of order placed at time t=0 is modified from 10 to 9 (price unchanged), the message will be:

  M, RELIANCE, B, 10, 9, 800, 800

  For any order, price and quantity can be modified

* Cancel Order (type: X) - when someone cancels his order. Format is:

  <type>, SYMBOL, SIDE(Buy/Sell), QTY, PRICE

For example, if the order at t=0 is cancelled by the user, the message will be:

  X, RELIANCE, B, 10, 800

* Trade Message (type: T) - when a trade happens. Format is:

  <type>, SYMBOL, TRADED-QUANTITY, TRADE-PRICE

  For example, the trade which occured at t=5, message will be:

T, RELIANCE, 1, 800

## INPUT

Input for this project will be the above messages given on the input (stdin). Each message goes on a new line. At the end of the input, there will be a Query message. For each query message, you will print the order book for that particular symbol, up-to the rank specified in the query message.

For example:

Q, RELIANCE, 5

## OUTPUT

Output for this project will be the order book at the exchange. An output of the actual order book is attached. BUY orders are shown in GREEN and are arranged by price in the descending order. Similarly, SELL orders are shown in RED and are arranged by price in the ascending order. You need to show the top five bids and asks for each symbol.

SYMBOL: RELIANCE

BUY: (Rank 1) 10, 800 (qty, price)

BUY: (Rank 2) 8, 799

BUY: (Rank 3) 6, 798

BUY: (Rank 4) 9, 797

BUY: (Rank 5) 22, 790

SELL: (Rank 1) 2, 801

SELL: (Rank 2) 15, 805

SELL: (Rank 3) 5, 808

SELL: (Rank 4) 27, 809

SELL: (Rank 5) 100, 825

***NOTE:***

* You can assume that the prices are whole numbers, always positive,
* Quantity is always positive
* Solutions should scale to any number of symbols. (Usually, on one exchange 2000-50000 instruments may be listed)
* Please send an email to harwinder.sidhu@utradesolutions.com to clear any doubts.

Things to take care of:

* Design skills (preferable OO design) (highest priority)
* Implementation of design
* Robustness of the solution (handling error conditions, parsing logic etc)
* Correctness of the Solution
* Efficiency of the solution

# EXAMPLE

## INPUT:

N,RELIANCE,B,10,800

N,RELIANCE,S,10,805

M,RELIANCE,B,10,9,800,800

T,RELIANCE,1,800

X,RELIANCE,S,5,805

T,RELIANCE,4,805

Q,RELIANCE,1

## OUTPUT:

BUY: (Rank 1) 8, 800

SELL: (Rank 1) 1, 805