

Basic order book management

How market works

Imagine that you are on a market which only trades one type of item (e.g. tomatoes of equal quality; generally known as "shares"). There will be a certain amount of tomatoes(shares) being sold at certain price. Also there will be people willing to buy at certain price. Imagine that everyone who could buy/sell at acceptable price(limit price) immediately does that and leaves the market. This way (most of the time), nobody can perform a trade right now and everyone has to wait until something changes (e.g. someone reconsiders "acceptable" price, or new person appears). Those "limit orders" (people willing to buy/sell in certain quantities) are the limit order book. In some cases people are willing to buy/sell at any price (that's called a "market order"), such person is always going to perform the trade and leaves the market.

In other words, each price level (for simplicity let's think of it as an integer value) can be either bid (there are people willing to buy at this price), ask (people are willing to sell at this price) or spread (nobody is willing to buy or sell at this price).

Generally, order book looks like the following example (B - bid, S - spread, A - ask), size defines how many shares can be bought/sold at this price:

PRICE	SIZE	TYPE	COMMENT
99	0	A	size is zero, but it is still ask price, because it is higher than best ask
98	50	A	best ask (lowest non-zero ask price)
97	0	S	
96	0	S	
95	40	B	best bid (highest non-zero bid price)
94	30	B	
93	0	B	
92	77	B	

Best bid is always lower than best ask. (for this task it's not important why it is so, but that is actually because otherwise those limit orders would execute).

Task

Initially you have an empty order book. Your task is to apply the updates and respond to queries as described under the "Input file" section.

Input file

There is a text file with:

updates to the limit order book in the following format:

u,<price>,<size>,bid - set bid size at <price> to <size> (<size> shares in total are now being offered at <price>)

u,<price>,<size>,ask - set ask size at <price> to <size>

queries in the following format:

q,best_bid - print best bid price and size

q,best_ask - print best ask price and size

q,size,<price> - print size at specified price (bid, ask or spread).

and market orders in the following format:

o,buy,<size> - removes <size> shares out of asks, most cheap ones.

o,sell,<size> - removes <size> shares out of bids, most expensive ones.

In case of a buy order this is similar to going to a market (assuming that you want to buy <size> similar items there, and that all instances have identical quality, so price is the only factor) - you buy <size> units at the cheapest price available.

Queries, market orders, and limit order book updates are in arbitrary sequence. Each line in the file is either one of the three and ends with newline character.

Example of input file:

```
u,9,1,bid
u,11,5,ask
q,best_bid
u,10,2,bid
q,best_bid
o,sell,1
q,size,10
u,9,0,bid
u,11,0,ask
```

Output file

Example of output file (for this input file):

```
9,1
10,2
1
```

Please, follow output format as closely as possible.

Additional requirements

- use Java
- send us source code as archive (do not publish it)
- performance matters
- your solution will be tested automatically with different inputs

Contacts

Send solution by email: hr@bookmap.com

Do you need any extra information to complete this task?

If you will have any additional questions later - feel free to ask, email: hr@bookmap.com