```
#order={ "side": B, "quantity":100, "price": 10, "symbol" : MQ, "Exchange": E1, "orderid":1}


class orderbook():
    def __init__(self):
        self.buy = []
        self.sell = []

    def add(self, order):

        if order['side'] == 'B':
            self.buy.append(order)


        if order['side'] == 'S':
            self.sell.append(order)

        else:
            print('ERROR')


    def cancel(self, order):
        for i in self.buy:
            if order['orderid'] == i['orderid'] and order['exchange'] == i['exchange']:
                del order[i]

        for i in self.sell:
            if order['orderid'] == i['orderid'] and order['exchange'] == i['exchange']:
                del order[i]

    def modify(self, order):
        for i in self.buy:
            if order['orderid'] == i['orderid'] and order['exchange'] == i['exchange'] and order['price'] !=
i['price'] or order['quantity'] != i['quantity']:
                # replace order['price'], replace order['quantity']
                order['price'] = i['price']
                order['quantity'] = i['quantity']

        for i in self.sell:
            if order['orderid'] == i['orderid'] and order['exchange'] == i['exchange'] and order['price'] !=
i['price'] or order['quantity'] != i['quantity']:
                # replace order['price'], replace order['quantity']
                order['price'] = i['price']
                order['quantity'] = i['quantity']

    def top_of_book(self, order):
        # return the best bid and offers
        # use sorted() to reorganize the book by key 'price', one of the books is reversed
        self.sell = sorted(self.sell, order['price'], reverse=True)
        self.buy = sorted(self.buy, order['price'])
```

```
        # print the first item in each list
        print(self.sell[0], self.buy[0])
#http://stackoverflow.com/questions/72899/how-do-i-sort-a-list-of-dictionaries-by-values-of-the-dictionary-in-python
```