

1 STRUCTURE OF THE CODE

In many scenarios, auction may not be restricted within the form of English Auction, Dutch Auction, etc. A combination of such forms is used, for example, in fish market, sellers first open-cry and descend the price for their fish, within a very short time, if more than one buyers express their willingness to buy, the seller then start to open-cry and ascend the price with only those bidders who bid in the last round of the Dutch auction.

Intuitively, such selling process is a combination of Dutch Auction and English Auction, where Dutch Auction was executed firstly and then English Auction. Such simple behaviour can be quite challenging to realize in software agents, which is the aim of this assessment.

1.1 INPUT

Auctioneer Input:

1. The initial item price (the starting Dutch auction price which is higher than Auctioneer's reserved price.). The initial input is obtained by defining it in cmd.
2. The reserved price is defined by the following code. Where RESERVE_RATE could be adjusted. Here the RESERVE_RATE was defined as 0.4.

```
private void setReservePrice() {  
    reservePrice = initialPrice * RESERVE_RATE;  
}
```

Bidder Input:

1. The lowest reserved price. It is obtained by input through cmd. When the CFP item price is lower than "lowest reserved price", the bidder will send PROPOSE to Auctioneer. To simplify the code, the entry for each bidder's "lowest reserved price" was set the same. More sophisticated scenarios could be simulated in the future.
2. The highest reserved price. It is defined by the following code, where every bidder's highest reserve price is different. When the item price is higher than "highest reserve price", the Bidder will send REFUSE to Auctioneer.

```
final double d = Math.random();  
final int i = (int) (d*5)+1;
```

```
priceToBuy = Double.parseDouble(args[0].toString());  
highpriceToBuy=priceToBuy*i;
```

2 OUTCOME

2.1 TEST CASES

- 5 test cases were done with different combination of Inputs mentioned in Section 1.1. The result is shown in Table

Table 1

Initial Item Price=500	Result
Bidder 1=400, Bidder 2=400	Both go to the English Auction, and Bidder 2 wins (as it has higher “highest reserved price”)
Bidder 1=400, Bidder 2=350	Bidder 1 goes to the English Auction, and wins.
Bidder 1=400, Bidder 2=350, Bidder 3=400	Bidder 1 and 3 go to the English Auction, and Bidder 1 wins (as it has higher “highest reserved price”)
Bidder 1=400, Bidder 2=400, Bidder 3=350	Bidder 1 and 2 go to the English Auction, and Bidder 1 wins (as it has higher “highest reserved price”)
Bidder 1=550, Bidder 2=550, Bidder 3=350	Bidder 1 and 2 go to the English Auction, and Bidder 2 wins (as it has higher “highest reserved price”)

The test reveals the following bug:

When there is only one bidder winning from Dutch Auction, the bidder will keep increasing its bid in English Auction turn which is contrary to common sense.

2.2 TRANSITION FROM DUTCH AUCTION TO ENGLISH AUCTION

By defining the different behaviour modules in Finite State Machines, the transition could be realized.

As shown in Figure 1, When more than one bidder sends PROPOSE for an item during the Dutch Auction, the Auctioneer will broadcast to all the participants “Dutch Auction finished and [xxx] will continue for English Auction”. Afterwards the process goes into English Auction Protocol and a English CFP will be sent.

```

Buyer [Bidder1@100.67.0.30:1099/JADE] answered : My decision for this round is PROPOSE
Buyer [Bidder2@100.67.0.30:1099/JADE] answered : My decision for this round is PROPOSE
Bidder2: I will continue the Auction!
Auctioneer: Dutch Auction finished and Bidder1@100.67.0.30:1099/JADE will continue for English Auction!
Bidder1: I will continue the Auction!
Auctioneer: Dutch Auction finished and Bidder2@100.67.0.30:1099/JADE will continue for English Auction!
Auctioneer: Sending English Auction CFP to [Bidder1]
Auctioneer: Sending English Auction CFP to [Bidder2]

```

Figure 1

2.3 THE ENDING OF THE CODE

As shown in Figure 2, when the item price is higher than bidder's "highest reserved price", the bidder will say "It's too high I will not bid". When no proposals received, the Auctioneer will end the auction.

As pointed out in the Limitation part, the software can not record the bidder with the highest bids. However the Auctioneer know the highest bid thus the benefits could be calculated as the gap between the Auctioneer's reserved price and the amount the bidder paid.

```

Auctioneer: Start collecting English Bids
Bidder2: It's too high I will not bid
Auctioneer: I received bid from Bidder1@100.67.0.30:1099/JADE as 505.0
Auctioneer(Thinking in head): I will increase the price to 555.5
Terminating buyer [Bidder2]...
Auctioneer: Start collecting English Bids
Auctioneer: no Proposal recived, the winner will be the one who bid the highest offer last turn
Auctioneer: Auction End
=====
| The number of messages exchanged is: 35
| The profit gained is: 355.5
=====

```

Figure 2

3 LIMITATION

In the English Auction Part, the Bidder with the highest Bidder can not be recorded. One way of realizing this function is by using HashMap to map the Bidder and its bid for each entry. Due to my limitation of coding skill, such function was not reached.
