

README

Author: Isabella Samuelsson UnityId: insamuel

Files

auc_server.py : Auction server class. There are two types of auctions type 1 which is first price auction and type 2 which is second price auction. In a first price auction the participant with the highest bid wins, and pays exactly the price that he/she bids (the highest bid). In a second price auction the participant with the highest bid also wins, but only pays the price of the second highest bid. The server has two clients connecting Buyers and Sellers. The first client to connect is designated the seller and the others will be buyers. The server will accept incoming connections, designate seller and buyers and relay the result of the auction. Once the auction has been concluded the server will disconnect the seller and buyer clients, restart the auction state and will wait for more incoming connections to start another auction.

Seller Client: The first client to connect will be the seller. The server prompts for auction information. This includes Auction Type: 1 for a first price auction and 2 for a second price auction, Minimum Bid price: non-negative integer, Number of Bidders: non-negative integer less than 10 and Item Name: string. If invalid auction information is received the server will continue to prompt for valid information. If a client tries to connect while auction information is being received the client will be sent a busy message and disconnected.

Buyer Client: All subsequent connections will be designated buyers. The server will only allow the specified number of bidders sent in the auction information given by the seller. If additional clients try to connect a server busy message will be sent to the client and then it will be disconnected. The server will prompt for a bid. A bid should be a non-negative integer, if an invalid bid is given the client will be prompted again.

Run example: `python3 auc_server.py server_port_number`

auc_client.py : Auction client class. If you are the first to connect to the server you will be connected as a seller client, if not you will be connected as a buyer client. Bids must be an integer greater than zero.

Seller Client: Once connected you will be prompted to enter auction information. This includes Auction Type: 1 for a first price auction and 2 for a second price auction, Minimum Bid price: non-negative integer, Number of Bidders: non-negative integer less than 10 and Item Name: string. If you enter invalid auction information you will be prompted for valid information before continuing.

Buyer Client: Once connected you will be prompted to enter a bid. A bid should be a non-negative integer, if an invalid bid is given you will be prompted again.

Run example: `python3 auc_client.py server_ip_address server_port_number`

Compilation and Run Instructions

Once you are in the terminal at the directory with both the `auc_server.py` and `auc_client.py` class run: **`python3 auc_server.py server_port_number`** with `server_port_number` as the port you would like to use. Then in another terminal to start a client run: **`python3 auc_client.py server_ip_address server_port_number`** with `server_ip_address` as the servers ip address and `server_port_number` as the servers port number specified in the step above. The first client connection will be the seller, all subsequent client connections will be buyers.

Example Functionality

Example: Successful auction for auction type 1

Shows: invalid seller auction information + invalid bid

- #1 Start server on port 300 and ip address 152.7.177.13.
- #2 Seller connects and gives incorrect auction information " sword -3 11 wolf" and "3 wol"
- #4 Seller client gives the auction information: "1 100 3 WolfPackSword".
- #5 Three buyer clients connect to the server.
- #6 A fourth buyer client tries to connect but the server is busy handling bidding.
- #7 The clients give invalid bids -50, -100, wolf.
- #8 The clients give valid bids 150, 120, 180.
- #9 Winner announced and the auction is over.
- #10 Auction server restarts state and is ready for another auction.

Server:

```
[[insamuel@vclvm177-13 ~]$ python3 auc_server.py 3000
Auctioneer is ready for hosting auctions!

Seller is connected from 152.7.177.15

>> New Seller Thread spawned

Auction request received. Now waiting for Buyer.

Buyer 1 is connected from 152.7.177.22

Buyer 2 is connected from 152.7.177.10

Buyer 3 is connected from 152.7.177.19

Requested number of bidders arrived. Let's start bidding!

>> New Bidding Thread spawned

Buyer 1 bid $150

Buyer 2 bid $120

Buyer 3 bid $180

<< Item sold! The highest bid is $180. The actual payment is $180.

Auction Restart: Auctioneer is ready for hosting auctions!

█
```

Seller Client:

```
[[insamuel@vclvm177-15 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Seller]
Please submit auction request:
[-3 11 wolf
Server: Invalid auction request!
Please submit auction request:
[3 wol
Server: Invalid auction request!
Please submit auction request:
[1 100 3 WolfPackSword
Server: Auction start

Auction finished!
Success! Your item WolfPackSword has been sold for $180.
Disconnecting from the Auctioneer server. Auction is over!

[[insamuel@vclvm177-15 ~]$ █
```

Buyer 1:

```
[insamuel@vclvm177-22 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Buyer]

The auctioneer is still waiting for other Buyer to connect...

The bidding has started!
Please submit your bid:
-50
Server: Invalid bid. Please submit a positive integer!
Please submit your bid:
150
Server: Bid received. Please wait...

Auction finished!
Unfortunately you did not win the last round.
Disconnecting from the Auctioneer server. Auction is over!

[insamuel@vclvm177-22 ~]$
```

Buyer 2:

```
[insamuel@vclvm177-10 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Buyer]

The auctioneer is still waiting for other Buyer to connect...

The bidding has started!
Please submit your bid:
-100
Server: Invalid bid. Please submit a positive integer!
Please submit your bid:
120
Server: Bid received. Please wait...

Auction finished!
Unfortunately you did not win the last round.
Disconnecting from the Auctioneer server. Auction is over!

[insamuel@vclvm177-10 ~]$
```

Buyer 3:

```
[insamuel@vclvm177-19 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Buyer]

The bidding has started!
Please submit your bid:
wolf
Server: Invalid bid. Please submit a positive integer!
Please submit your bid:
180
Server: Bid received. Please wait...

Auction finished!
You won this item WolfPackSword! Your payment due is $180
Disconnecting from the Auctioneer server. Auction is over!

[insamuel@vclvm177-19 ~]$
```

Example: Successful auction for auction type 1:

Shows: client join during seller busy + client join during bidding

- #1 Continue from above example with server on port 300 and ip address 152.7.177.13.
- #2 Seller client connects.
- #3 Another client tries to connect but the server is busy with seller interaction.
- #3 Seller gives the auction information: "1 100 2 WolfPackSword".
- #5 Two buyer clients connect to the server.
- #6 A third buyer client tries to connect but the server is busy handling bidding.
- #8 The clients give valid bids 150 and 120.
- #9 Winner announced and the auction is over.
- #10 Auction server restarts state and is ready for another auction.

Server:

```
<< Item sold! The highest bid is $180. The actual payment is $180.
Auction Restart: Auctioneer is ready for hosting auctions!
Seller is connected from 152.7.177.15
>> New Seller Thread spawned
Auction request received. Now waiting for Buyer.
Buyer 1 is connected from 152.7.177.22
Buyer 2 is connected from 152.7.177.10
Requested number of bidders arrived. Let's start bidding!
>> New Bidding Thread spawned
Buyer 1 bid $150
Buyer 2 bid $120
<< Item sold! The highest bid is $150. The actual payment is $150.
Auction Restart: Auctioneer is ready for hosting auctions!
```

Seller Client:

```
[[insamuel@vclvm177-15 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Seller]
Please submit auction request:
[1 100 2 WolfPackSword
Server: Auction start

Auction finished!
Success! Your item WolfPackSword has been sold for $150.
Disconnecting from the Auctioneer server. Auction is over!
[[insamuel@vclvm177-15 ~]$ python3 auc_client.py 152.7.177.13 3000
```

Buyer 1:

```
[insamuel@vclvm177-22 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Buyer]

The auctioneer is still waiting for other Buyer to connect...

The bidding has started!
Please submit your bid:
150
Server: Bid received. Please wait...

Auction finished!
You won this item WolfPackSword! Your payment due is $150
Disconnecting from the Auctioneer server. Auction is over!

[insamuel@vclvm177-22 ~]$
```

Buyer 2:

```
[insamuel@vclvm177-10 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Buyer]

The bidding has started!
Please submit your bid:
120
Server: Bid received. Please wait...

Auction finished!
Unfortunately you did not win the last round.
Disconnecting from the Auctioneer server. Auction is over!

[insamuel@vclvm177-10 ~]$
```

Client Connect during server busy with seller communication:

```
[insamuel@vclvm177-19 ~]$ python3 auc_client.py 152.7.177.13 3000
Server is busy. Try to connect again later.

[insamuel@vclvm177-19 ~]$
```

Client connect during bidding:

```
[insamuel@vclvm177-19 ~]$ python3 auc_client.py 152.7.177.13 3000
Bidding on-going! Try to connect again later.

[insamuel@vclvm177-19 ~]$
```

Example: Unsuccessful auction for auction type 1:

- #1 Continue from above example with server on port 300 and ip address 152.7.177.13.
- #2 Seller client connects and gives the auction information: "1 100 2 WolfPackSword".
- #5 Two buyer clients connect to the server.
- #8 The clients give valid bids 50 and 90.
- #9 Winner announced and the auction is over.
- #10 Auction server restarts state and is ready for another auction.

Server:

```
<< Item sold! The highest bid is $150. The actual payment is $150.

Auction Restart: Auctioneer is ready for hosting auctions!

Seller is connected from 152.7.177.15

>> New Seller Thread spawned

Auction request received. Now waiting for Buyer.

Buyer 1 is connected from 152.7.177.10

Buyer 2 is connected from 152.7.177.22

Requested number of bidders arrived. Let's start bidding!

>> New Bidding Thread spawned

Buyer 1 bid $50

Buyer 2 bid $90

<< Item did not sell! The highest bid is $90 and the lowest price is $100.

Auction Restart: Auctioneer is ready for hosting auctions!

█
```

Seller Client:

```
[[insamuel@vclvm177-15 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Seller]
Please submit auction request:
[1 100 2 WolfPackSword
Server: Auction start

Auction finished!
Unfortunately your item WolfPackSword was not sold in the Auction.
Disconnecting from the Auctioneer server. Auction is over!

[[insamuel@vclvm177-15 ~]$ █
```

Buyer 1:

```
[[insamuel@vclvm177-10 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Buyer]

The auctioneer is still waiting for other Buyer to connect...

The bidding has started!
Please submit your bid:
[50
Server: Bid received. Please wait...

Auction finished!
Unfortunately you did not win the last round.
Disconnecting from the Auctioneer server. Auction is over!

[[insamuel@vclvm177-10 ~]$ █
```

Buyer 2:

```
[[insamuel@vclvm177-22 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Buyer]

The bidding has started!
Please submit your bid:
[90
Server: Bid received. Please wait...

Auction finished!
Unfortunately you did not win the last round.
Disconnecting from the Auctioneer server. Auction is over!

[[insamuel@vclvm177-22 ~]$ █
```

Example: Unsuccessful auction for auction type 2:

- #1 Continue from above example with server on port 300 and ip address 152.7.177.13.
- #2 Seller client connects and gives the auction information: "2 100 2 WolfPackSword".
- #5 Two buyer clients connect to the server.
- #8 The clients give valid bids 50 and 90.
- #9 Winner announced and the auction is over.
- #10 Auction server restarts state and is ready for another auction.

Server:

```
<< Item did not sell! The highest bid is $90 and the lowest price is $100.  
Auction Restart: Auctioneer is ready for hosting auctions!  
Seller is connected from 152.7.177.15  
>> New Seller Thread spawned  
Auction request received. Now waiting for Buyer.  
Buyer 1 is connected from 152.7.177.10  
Buyer 2 is connected from 152.7.177.22  
Requested number of bidders arrived. Let's start bidding!  
>> New Bidding Thread spawned  
Buyer 1 bid $50  
Buyer 2 bid $90  
<< Item did not sell! The highest bid is $90 and the lowest price is $100.  
Auction Restart: Auctioneer is ready for hosting auctions!  
█
```

Seller Client:

```
[insamuel@vclvm177-15 ~]$ python3 auc_client.py 152.7.177.13 3000  
Connected to the Auctioneer server.  
  
Your role is: [Seller]  
Please submit auction request:  
2 100 2 WolfPackSword  
Server: Auction start  
  
Auction finished!  
Unfortunately your item WolfPackSword was not sold in the Auction.  
Disconnecting from the Auctioneer server. Auction is over!  
  
[insamuel@vclvm177-15 ~]$ █
```

Buyer 1:

```
[insamuel@vclvm177-10 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Buyer]

The auctioneer is still waiting for other Buyer to connect...

The bidding has started!
Please submit your bid:
50
Server: Bid received. Please wait...

Auction finished!
Unfortunately you did not win the last round.
Disconnecting from the Auctioneer server. Auction is over!

[insamuel@vclvm177-10 ~]$
```

Buyer 2:

```
[insamuel@vclvm177-22 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Buyer]

The bidding has started!
Please submit your bid:
90
Server: Bid received. Please wait...

Auction finished!
Unfortunately you did not win the last round.
Disconnecting from the Auctioneer server. Auction is over!

[insamuel@vclvm177-22 ~]$
```

Example: Successful auction for auction type 2

- #1 Continue from above example with server on port 300 and ip address 152.7.177.13.
- #2 Seller client connects and gives the auction information: "2 100 3 WolfPackSword".
- #5 Three buyer clients connect to the server.
- #8 The clients give valid bids 150, 120, 180.
- #9 Winner announced and the auction is over.
- #10 Auction server restarts state and is ready for another auction.

Server:

```
<< Item did not sell! The highest bid is $90 and the lowest price is $100.  
Auction Restart: Auctioneer is ready for hosting auctions!  
Seller is connected from 152.7.177.15  
>> New Seller Thread spawned  
Auction request received. Now waiting for Buyer.  
Buyer 1 is connected from 152.7.177.10  
Buyer 2 is connected from 152.7.177.22  
Buyer 3 is connected from 152.7.177.19  
Requested number of bidders arrived. Let's start bidding!  
>> New Bidding Thread spawned  
Buyer 1 bid $150  
Buyer 2 bid $120  
Buyer 3 bid $180  
<< Item sold! The highest bid is $180. The actual payment is $150.  
Auction Restart: Auctioneer is ready for hosting auctions!
```

Seller Client:

```
[insamuel@vclvm177-15 ~]$ python3 auc_client.py 152.7.177.13 3000  
Connected to the Auctioneer server.  
  
Your role is: [Seller]  
Please submit auction request:  
[2 100 3 WolfPackSword  
Server: Auction start  
  
Auction finished!  
Success! Your item WolfPackSword has been sold for $150.  
Disconnecting from the Auctioneer server. Auction is over!  
  
[insamuel@vclvm177-15 ~]$
```

Buyer 1:

```
[insamuel@vclvm177-10 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Buyer]

The auctioneer is still waiting for other Buyer to connect...

The bidding has started!
Please submit your bid:
150
Server: Bid received. Please wait...

Auction finished!
Unfortunately you did not win the last round.
Disconnecting from the Auctioneer server. Auction is over!

[insamuel@vclvm177-10 ~]$
```

Buyer 2:

```
[insamuel@vclvm177-22 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Buyer]

The auctioneer is still waiting for other Buyer to connect...

The bidding has started!
Please submit your bid:
120
Server: Bid received. Please wait...

Auction finished!
Unfortunately you did not win the last round.
Disconnecting from the Auctioneer server. Auction is over!

[insamuel@vclvm177-22 ~]$
```

Buyer 3:

```
[insamuel@vclvm177-19 ~]$ python3 auc_client.py 152.7.177.13 3000
Connected to the Auctioneer server.

Your role is: [Buyer]

The bidding has started!
Please submit your bid:
180
Server: Bid received. Please wait...
Auction finished!
You won this item WolfPackSword! Your payment due is $150
Disconnecting from the Auctioneer server. Auction is over!

[insamuel@vclvm177-19 ~]$
```