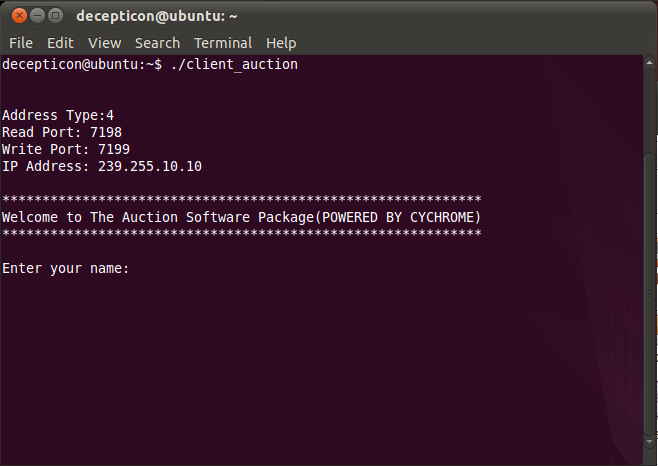
|  |  |  |
| --- | --- | --- |
| DST Assignment 2 | November 6  2011 | |
| U3030429, John Agbulos, 9.30AM Tuesday  U3036875, Cameron Ly, 5.30PM Wednesday  Questions attempted  A1  A2  A3  A4  A5  B6  B8  B9  B10 | |  |

**Client Program – PartA**

\*Note: Same Auction for client and server was used for all the screenshots in this document.

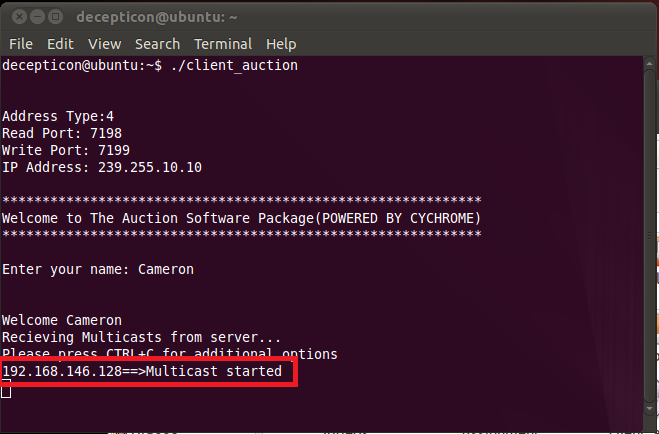
**Step 1:**

**Upon starting the client program, the user provides their name**

****

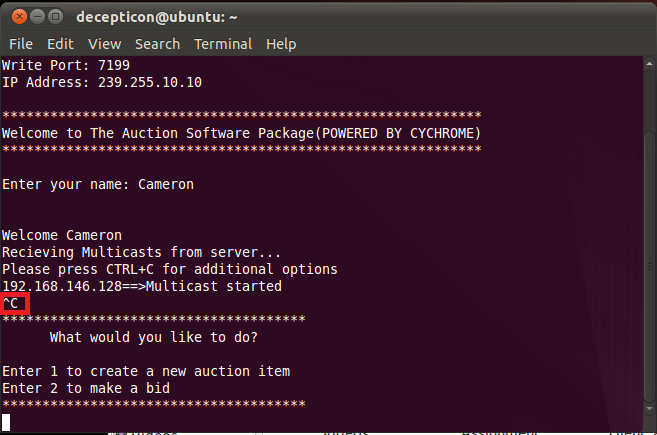
**Step 2:**

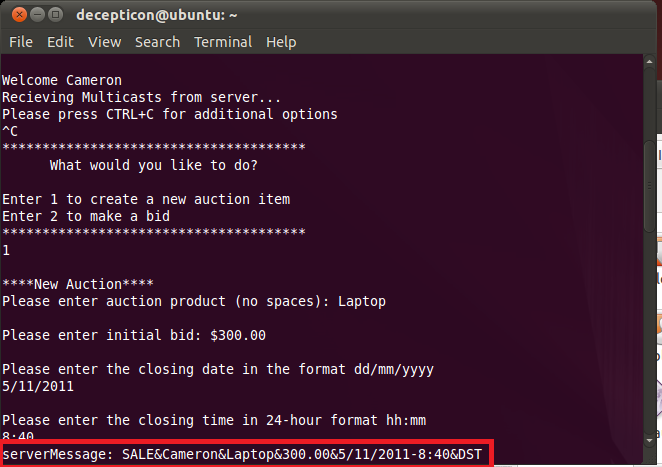
**Once the user has provided his name, the Client Program will continuously receive unicast messages from a server. The image below is a modified version of the Client Program which displays all messages sent to its receiving port**

****

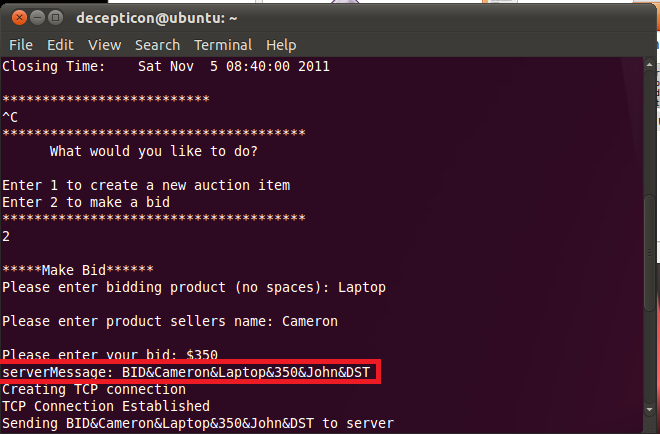
**Step 3:**

**The client program allows the user to make an auction or a bid by pressing CTRL+C (circled in red). When pressed, this raises an interrupt in the client program (new sales received will be displayed once the interrupt is complete.)**

****

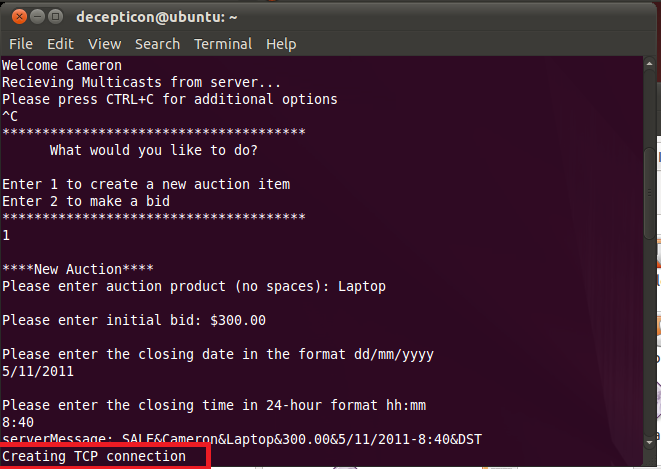
**The example below shows a user making an auction. Once the user has input the required details a formatted message will be created. The program below has been modified to display the message string that is about to be sent out to the server (circled in red)** 

**The example below shows a new user (John) on a separate client program making a bid. Once the user has input the required details a formatted message will be created. The program below has been modified to display the message string that is about to be sent out to the server (circled in red)**



**Step 4:**

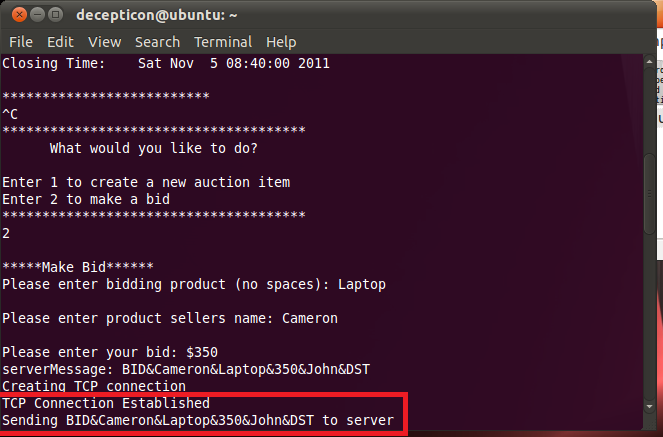
**Once the user has provided the details, the client program will send a request for a TCP connection. The program below has been modified to display when the connection occurs in the console**



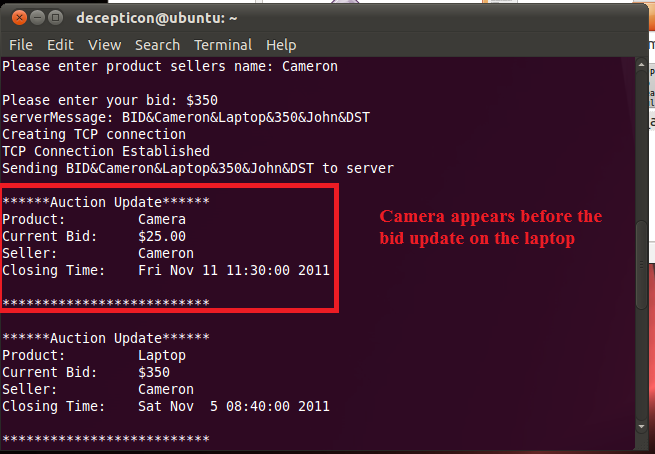
**Step 5:**

**When the connection has been established to the server via TCP, the client program will send the server message string to the server and end the interrupt routine.**

**Below is user John bidding for Cameron’s Laptop for $350. You can see the successful connection and the server message that is being sent to the server below.**

****

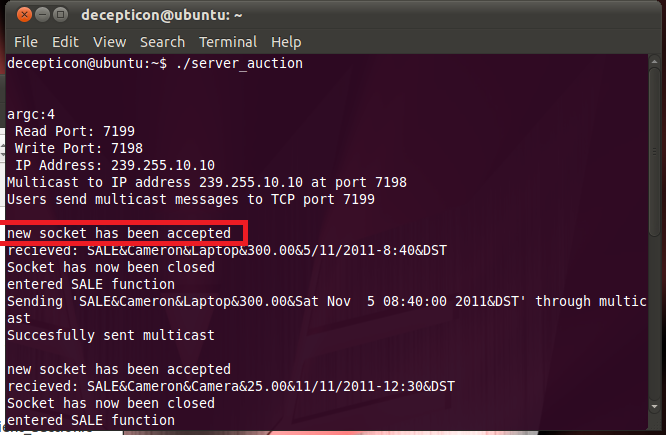
**Note also upon completion, the new sale ‘Camera’ is displayed after the completion of the interrupt**

****

**Server Program – PartB**

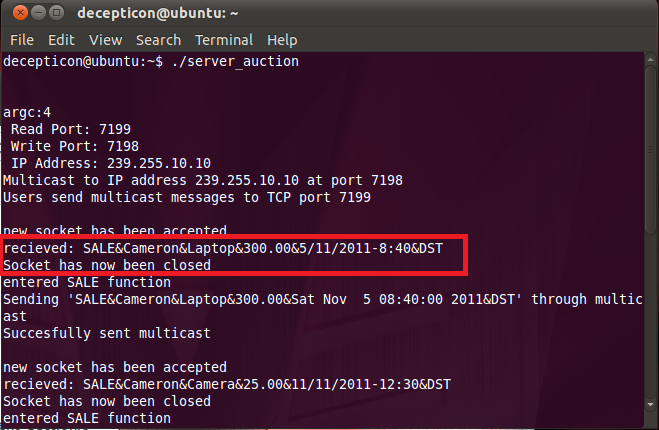
**Step 6:**

**Upon obtaining a request from a client, the server program establishes a TCP connection with the unique client (as seen in red)**

****

**Step 8:**

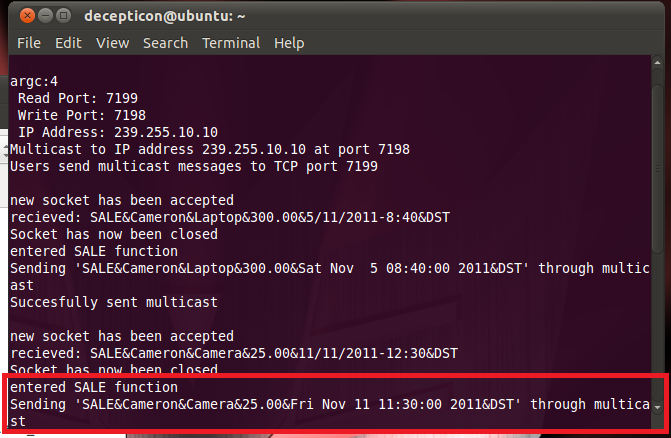
**Once the message is received the server program will close the connection (shown in red)**



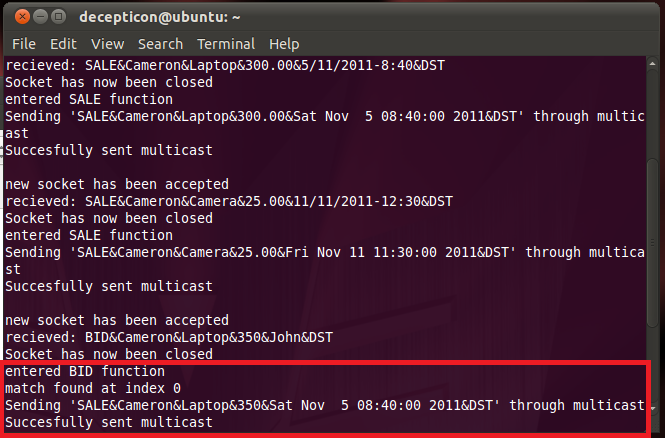
**Step 9:**

**Once the message is received, the server validates it with by the footer (DST) and header of either SALE or BID**

**The example below shows what happens when a SALE message is received. This is the same Auction displayed on the client side in step 5 (Camera Auction)**



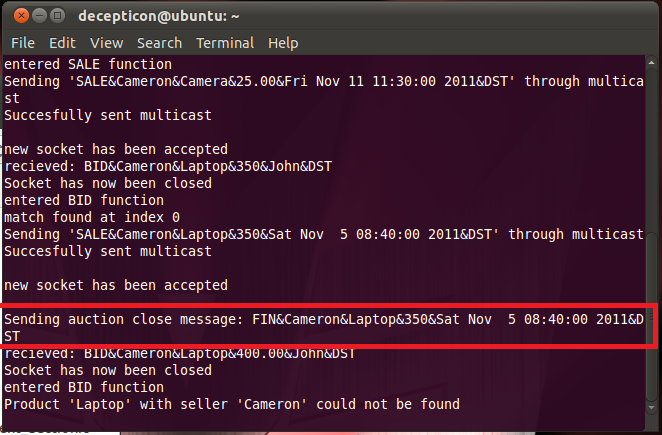
**The example below shows what happens when a BID message is received. This is the same Auction displayed on the client side in step 5 (Laptop bid)**



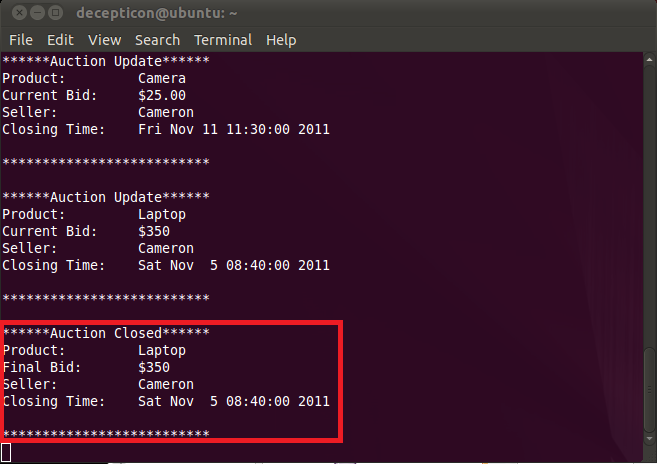
**Step 10:**

**If any auction has passed its closing time, the server multicasts a message which informs the auction session is closed.**

**This is the message being sent from the server program**



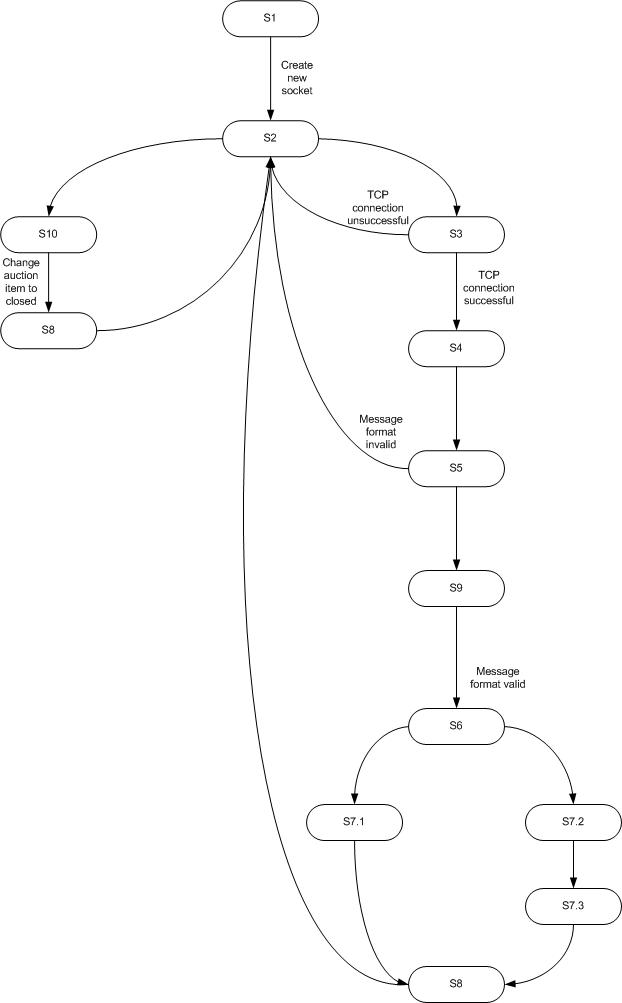
**And below is the message being displayed on the client side**



# Client state diagram

# clientstate.jpg

# Server state diagram



# Client flow chart

# FC_client.jpg

# Server flow chart

# serverstate.jpg