John Malandrakis CUS1156

Lab 9

Part 1

1. The code is designed so that every time a new Bid is sent to the BidBus, all Bidders are notified and will check to see if their limit has been hit. If the Bidder's limit has not been hit, they will continue bidding along with the other Bidders until every Bidder's limit has been hit.
2. When a Bidder makes a Bid, it checks first to see if the Bidder's limit has been hit and if the Bidder's limit has not been hit, it next checks to see if the Bidder already placed the latest Bid. If the Bidder has both not hit their limit and not placed the latest Bid, it will generate a random number between 0.0 and 10.0 and send a new Bid to the BidBus of the latest bid plus the generated number.
3. BidBus first sets the state as changed, and then notifies all the Observers which in turn causes the Observers to run an update method which allows the Observer to “react” to the changed state set by the Observable.
4. The update method is called by the BidBus when it runs notifyObservers().
5. I can use the limitHit instance variable to signify that the limit was hit, and the limit instance variable to set the Bidder's limit.

Part 2

1. The program keeps track of the number of students in each class and then displays the number of students in the classes after the freshman center finishes scheduling and then displays again after the Dean's office finishes scheduling.