**package** p1;

/\*\*

\* Class provided for ease of test. This will not be used in the project

\* evaluation, so feel free to modify it as you like.

\*/

**public** **class** Simulation

{

**public** **static** **void** main(String[] args)

{

**int** nrSellers = 1000;

**int** nrBidders = 500;

Thread[] sellerThreads = **new** Thread[nrSellers];

Thread[] bidderThreads = **new** Thread[nrBidders];

Seller[] sellers = **new** Seller[nrSellers];

Bidder[] bidders = **new** Bidder[nrBidders];

// Start the sellers

**for** (**int** i=0; i<nrSellers; ++i)

{

sellers[i] = **new** Seller(

AuctionServer.*getInstance*(),

"Seller"+i,

100, 50, i

);

sellerThreads[i] = **new** Thread(sellers[i]);

sellerThreads[i].start();

}

// Start the buyers

**for** (**int** i=0; i<nrBidders; ++i)

{

bidders[i] = **new** Bidder(

AuctionServer.*getInstance*(),

"Buyer"+i,

1000, 20, 150, i

);

bidderThreads[i] = **new** Thread(bidders[i]);

bidderThreads[i].start();

}

// Join on the sellers

**for** (**int** i=0; i<nrSellers; ++i)

{

**try**

{

sellerThreads[i].join();

}

**catch** (InterruptedException e)

{

e.printStackTrace();

}

}

// Join on the bidders

**int** biddersMoney=0;

**for** (**int** i=0; i<nrBidders; ++i)

{

**try**

{

bidderThreads[i].join();

biddersMoney+=bidders[i].cashSpent();

System.*out*.println("most items available for bidder:"+bidders[i].mostItemsAvailable());

}

**catch** (InterruptedException e)

{

e.printStackTrace();

}

}

// **TODO**: Add code as needed to debug

//check for consistency maintained by concurrent execution

//CHECK if total money spent by bidder is equal to total revenue collected at server

//System.out.println("total items available:"+AuctionServer.getInstance().getItems().size());

System.*out*.println("Total items sold in auction:"+AuctionServer.*getInstance*().soldItemsCount());

**if**(biddersMoney==AuctionServer.*getInstance*().revenue())

System.*out*.println("Auction server consistent with auctions made:"+"biddersmoney:"+biddersMoney+" revenue:"+AuctionServer.*getInstance*().revenue());

**else**

System.*out*.println("Auction server in inconsistent state:"+"biddersmoney:"+biddersMoney+"revenue:"+AuctionServer.*getInstance*().revenue());

System.*out*.println("highestbidsum:"+AuctionServer.*getInstance*().highestbidssum());

}

}