# CS 677 Lab 1 Test Report

**Dan Cline, Kyle Stevens, Tian Xia**     *UMass Amherst*

---

This document is a report of all the tests ran on the system, and presents sample outputs of each test. To reproduce the results of the tests, run `runtest.sh` in the `test` directory on some `networkconfig.yml`.
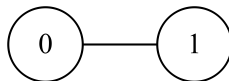
---

**Test 1**

Test Description: Assign one peer to be a buyer of fish and another to be a seller of fish. Ensure that all fish is sold and restocked forever. In `test1config.yml`, the number of fishes for the seller node is assigned to 10 and the unlimited flag is set to true, so that fish is sold throughout.

Node Assignment:

| nodes | address | role | sell | buy |
|-------|---------|------|------|-----|
| node0 | localhost:10000 | seller | fish (unlimited) | N/A |
| node1 | localhost:10001 | buyer | N/A | fish |

Network Topology:



Program Behaviour: Buyer node `node1` continuously buys from seller node `node0`. The number of fish remaining for the seller goes from 10 to 0 and gets restocked once it hits 0.
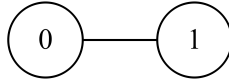
**Test 2**

Test Description: Assign one peer to be a buyer of fish and another to be a seller of fish and boar. Ensure that all fish is sold and restocked forever, but no boar is sold. In `test2config.yml`, the number of fishes for the seller node is assigned to 10 and the unlimited flag is set to true, so that fish is sold throughout. The number of boars for the seller node is also set to 10.

Node Assignment:

| nodes | address | role | sell | buy |
|-------|---------|------|------|-----|
| node0 | localhost:10000 | seller | fish/boar | N/A |
| node1 | localhost:10001 | buyer | N/A | fish |

Network Topology:

Program Behaviour: Buyer node `node1` continuously buys from seller node `node0`. The number of fish remaining for the seller goes from 10 to 0 and gets restocked once it hits 0. No boar is sold to `node1`.
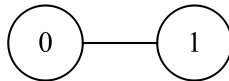
**Test 3**

Randomly assign buyer and seller roles. Ensure that items keep being sold throughout. In `test3config.yml`, the roles of the two nodes are set to `random`. The number of salt for the seller node is assigned to 1 and the unlimited flag is set to true, so that salt is sold throughout. The buyer target for the randomly chosen buyer node is set to salt.

Node Assignment:

| nodes | address | role | sell | buy |
|-------|---------|------|------|-----|
| node0 | localhost:10000 | random | salt(unlimited) | N/A |
| node1 | localhost:10001 | random | N/A | salt |

Network Topology:



Program Behaviour: Salt is sold throughout and restocked if one node is a buyer and the other is a seller. No products are sold if both nodes have the same role.
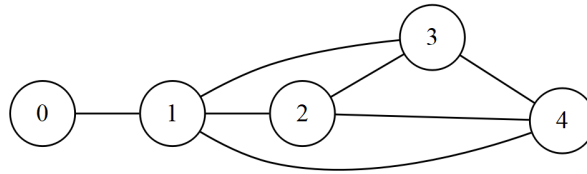
**Test 4**

One seller of boar, 3 buyers of boars, the remaining peers have no role. Fix the neighborhood structure so that buyers and sellers are 2-hop away in the peer-to-peer overlay network. Ensure that all items are sold and restocked and that all buyers can buy forever.

Node Assignment:

| nodes | address | role | sell | buy |
|-------|---------|------|------|-----|
| node0 | localhost:10003 | seller | boar | N/A |
| node1 | localhost:10004 | none | N/A | N/A |

2

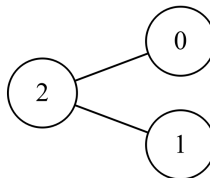| nodes | address | role | sell | buy |
|-------|---------|------|------|-----|
| node2 | localhost:10001 | buyer | N/A | boar |
| node3 | localhost:10002 | buyer | N/A | boar |
| node4 | localhost:10000 | buyer | N/A | boar |

Network Topology:



**Test 6**

Simulate a race condition in buy() wherein a seller has stock of 1 for an item but then replies to multiple buyers.

Node Assignment:

| nodes | address | role | sell | buy |
|-------|---------|------|------|-----|
| node0 | localhost:10002 | buyer | N/A | salt |
| node1 | localhost:10000 | buyer | N/A | salt |
| node2 | localhost:10001 | seller | salt | N/A |

Network Topology:



**Test 7**

Test Description: Stress test for larger networks with 5 buyers and 5 sellers for each item, and 20 additional generated nodes. See `test\test7config.yml` for more details.

Program Behaviors: Transactions goes smoothly between peers. No wrong items are sold or bought.

**Test 8**

Test Description: Stress test for larger networks with random nodes, with 50 generated nodes. See `test\test8config.yml` for more details.

Program Behaviors: Network random generation is successful. Transactions goes smoothly between peers. No wrong items are sold or bought.

**Test 9**

Test Description: Assign two nodes to have no role. Ensure that attempts are made to buy or sell. See `test\test9config.yml` for more details.

Program Behaviors: Network generation successful. No items are ever sold.

**Test 10**

Test Description: Stress test for many nodes with multiple items. 30 static nodes and 20 generated nodes For each single good there are two buyers, an unlimited seller and a seller with 10 stock. For each pair of goods there are two buyers, an unlimited seller and a seller with 10 stock. For all three goods there are three buyers and three unlimited sellers. See `test\test10config.yml` for more details.

Program Behaviors: Network random generation is successful. Transactions goes smoothly between peers. No wrong items are sold or bought.