assignment2

Software Design:

1. Define the client and server classes that will handle the socket connections.
2. Define the messages that will be exchanged between the client and server.
3. Implement the LCR algorithm on the server.
4. Define the steps of the interaction process between the client and server, such as connection establishment, sending and receiving messages, and closing the connection.
5. Specify the number of nodes sent by the client to start the server LCR algorithm, and the server will simulate and send the simulation results
6. Design the server side to continuously listen for connections and serve multiple clients but sequentially, one after the other.
7. Increase the stability of the server, verify the parameters passed by the client, and ensure that the client's unexpected exit from the server is not affected and the log is printed.
8. Increase the ability of the client to proactively disconnect, detect unexpected server exit, and the client will automatically exit without blocking.

Software Testing:

test case:

Case Id	Test Case Description	Test Steps	Expected Results	Actual Results
1	Verify connect function	run server run client	print message and log	same
2	Verify persistent connection	exit client	1. server print log 2. listen other client	same
3	Verify simulation	client input node num	1. server print simulation log 2. client receive result	same
4	Verify input illegality	client input not a number	1. server use default number to simulate 2. client receive result	same
5	Verify exit	client input q or quit	1. client exit 2. server print log and listen other client	same
6	Verify exception exit	client exception exit	1. client exit 2. server print log and listen other client	same
7	Verify server exit	server exit	client print log and exit	same

Conclusion:

1. Using Java sockets can establish a TCP connection between the server and client, and transmit information over the network.

2. Using loops can achieve continuous monitoring of connections on the server.

3. The client can trigger simulation deployed on the server side and receive simulation results.