COP5615 – Chord – P2P System and Simulation

Project – 3

Akhil Maddukuri (23309667), Dhanush Pakanati (28079405)

Overview: The idea of this project is to simulate a chord protocol using ERLANG in reference to the algorithm mentioned in the paper -
https://pdos.csail.mit.edu/papers/ton:chord/paper-ton.pdf

The working of chord protocol is based on the following functionalities:

1. Creating or joining nodes to build a network.
2. Ability to transfer messages between the nodes based on finger tables.
3. Updating finger tables
4. Stabilizing network
5. Finding the average hops needed for the network to stabilize.

Steps to Execute the Code:

The logic for the chord protocol have been written in a two module: main logic and helper functions.

To start the execution:

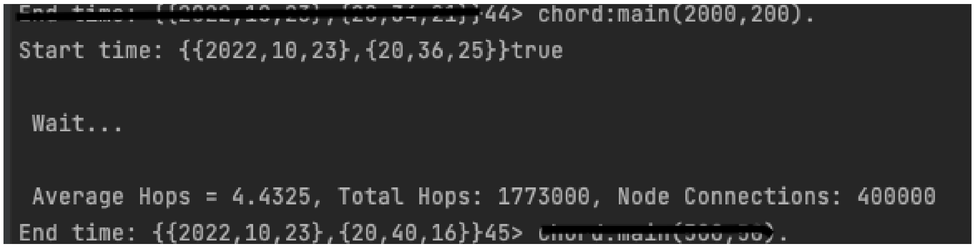
a. Open erlang shell: erl
b. Compile all erlang files: c(chordProtocol)., c(helper).
c. Run the main function: chordProtocol:main(1000,10)
   Here the parameters passed are the number of nodes and the number of requests for each node.

What is working?

1. Implementation of chord functionalities according to the algorithm mentioned in the paper - https://pdos.csail.mit.edu/papers/ton:chord/paper-ton.pdf
2. Functionalities: Create new ring, join, stabilize as mentioned in the above paper.\
3. Used hashing to create unique id's to all the nodes.

What is the largest network you managed to deal with?

The largest network we were able to deal with had 2000 nodes with 200 messages. The average number of hops for this network were: 4.435



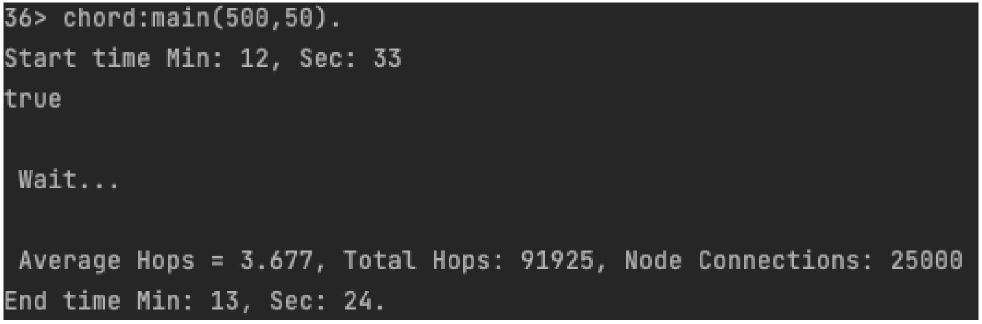
```
[redacted]44> chord:main(2000,200).
Start time: {{2022,10,23},{20,36,25}}true

 Wait...

 Average Hops = 4.4325, Total Hops: 1773000, Node Connections: 400000
End time: {{2022,10,23},{20,40,16}}45> [redacted].
```

Below is table showing the data about average hops, total hops, node connections and the time taken to stabilize.

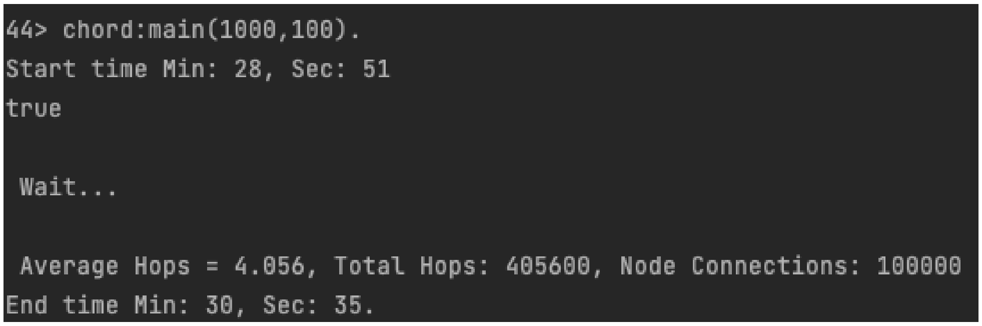
Number of Nodes	Requests	Average Hops	Time
500	50	3.677	51 sec
1000	100	4.056	1 min 44 sec
2000	200	4.435	3 min 51 sec

The average hops keep increasing with the increase in number of nodes and requests made.

Outputs:



```
36> chord:main(500,50).
Start time Min: 12, Sec: 33
true

 Wait...

 Average Hops = 3.677, Total Hops: 91925, Node Connections: 25000
End time Min: 13, Sec: 24.
```



```
44> chord:main(1000,100).
Start time Min: 28, Sec: 51
true

 Wait...

 Average Hops = 4.056, Total Hops: 405600, Node Connections: 100000
End time Min: 30, Sec: 35.
```