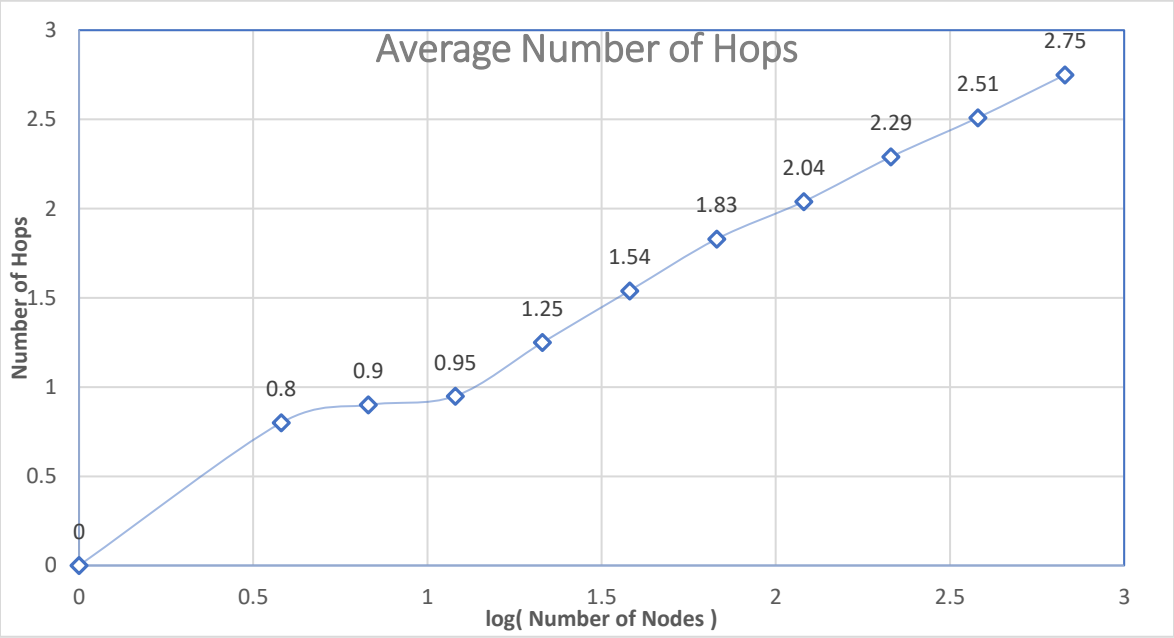
Project Report

Name : Mohit Israni UFID: 43384979

Name : Amit Kshirsagar UFID: 60032559

This document details the main part of the project. The network join and routing table was implemented as mentioned in the pastry protocol. Below are the findings for the average number of hops vs the number of nodes in the network.

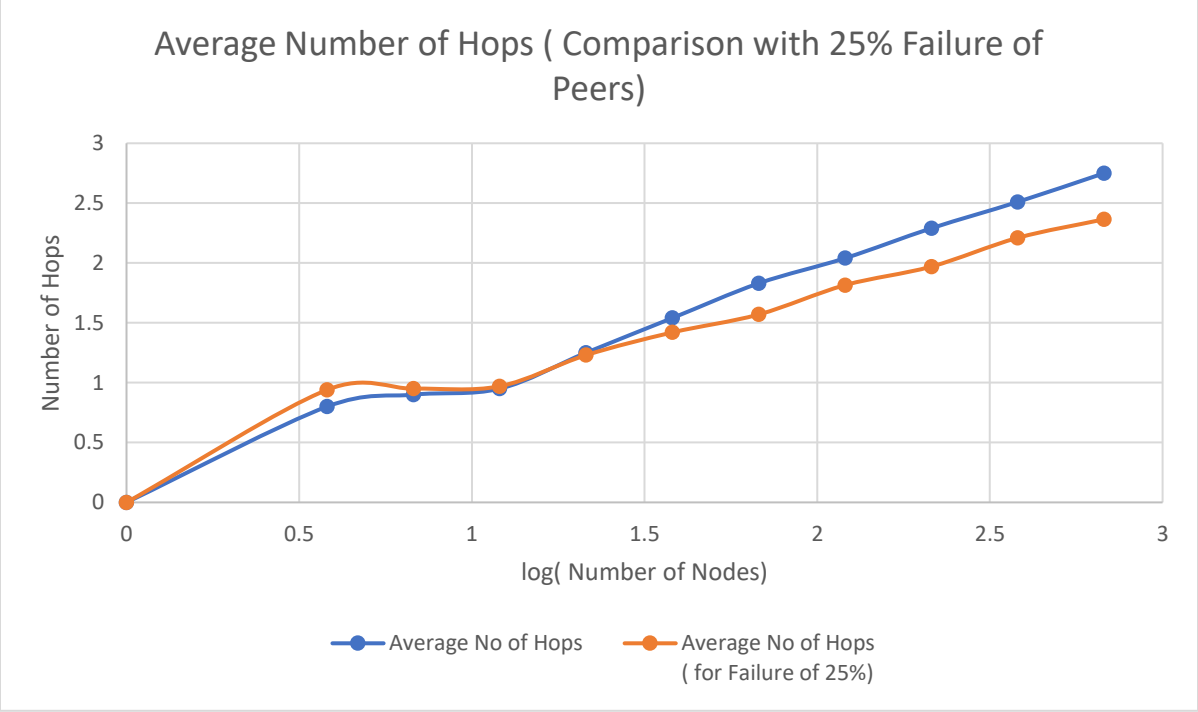



```
[MohitIsraniL.Mohit] ➤ ./pastry  2500 10

-----------------------------------------------
All Nodes STARTED...
Routing Tables and Leaf Sets UPDATED...
Peers will now start SENDING REQUESTS...
-----------------------------------------------
Requests Number: 1
Requests Number: 2
Requests Number: 3
Requests Number: 4
Requests Number: 5
Requests Number: 6
Requests Number: 7
Requests Number: 8
Requests Number: 9
Requests Number: 10
-----------------------------------------------
Average number of hops: 2.746458024952421
-----------------------------------------------
Thank you !! The evaluator process now EXIT..
-----------------------------------------------


** (EXIT from #PID<0.74.0>) killed
```

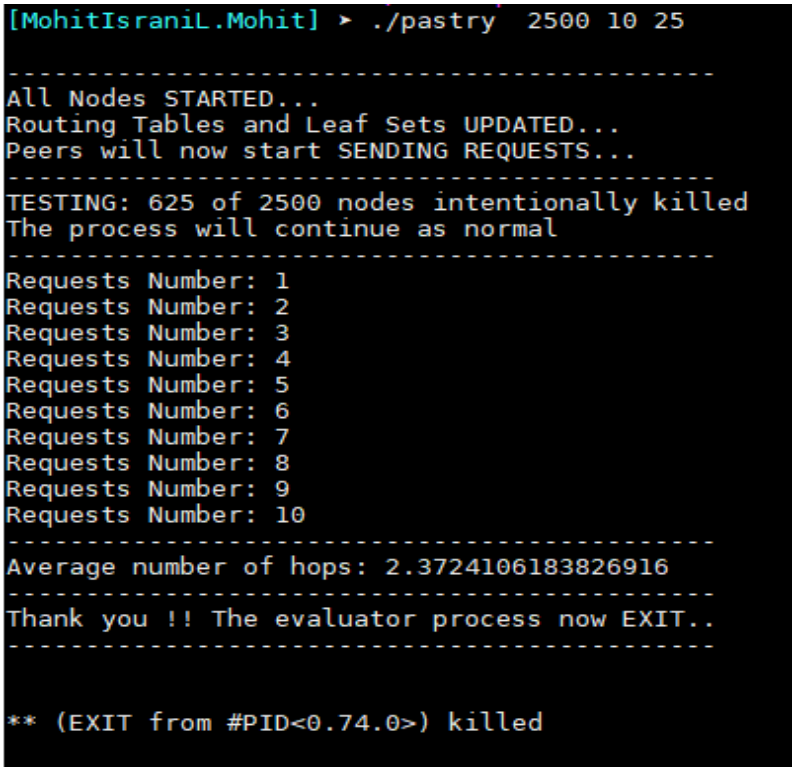
For Bonus Part:

This document details the findings and observations for the bonus part of the project. The Pastry protocol was implemented as given in the Pastry Paper. This was done by implementing the network join and the routing table. The bonus was implemented by introducing an additional parameter(main argument) called failure percentage. To analyze the failure part of this project we are randomly selecting nodes upto the given percentage of nodes(from command line arg) and intentionally kill them before starting the protocol. Below are the observations and analysis of the protocol.

To implement the failure check, we added a check before every request from a node whether the target node is alive or not. If the node is not alive, which in this case would be because the node has been hard killed by the command **Process.kill**, The node first removes the target from its leaf set or routing table and redistributes its leaf set in such a way that the workload is equally distributed among the workers. Below are the screenshots for the Bonus Part. However the system can only handle low failure percentage because on increasing the number of failure nodes it is possible that all of the leaves of particular node get killed. In this case the process gives an error. Below are the observations about the average number of hops for constant failure percentage (25%) and average number of hops when there are no failed nodes.



We see that, even after 25% failure the nodes are able to competitively deliver messages with almost the same number of average hops. We were able to deliver messages upto 80 % failed nodes



Screenshot of the protocol with 25% failed nodes.

The working and implementation steps are given in the Readme file.