DOS Project 3 Report(Bonus):

TEAM MEMBERS:
Name: Poornima Kumar
UFID: 5684-4925

Name: Harika Bukkapattanam
UFID: 3683-6895

PROGRAM EXECUTION:
Compilation:
mix escript.build
Running the project:
escript project3bonus numNodes numRequests numFail
Eg: escript project3bonus 100 10 20

IMPLEMENTATION:

- The implementation takes a parameter for number of failed nodes through command line as mentioned above.
- The FailureModule implementation is same as described for the PastryProtocol with the below described added functionalities:
    - Master:
        - It is also responsible for killing numFail number of nodes whose routing has already started.
    - Node:
        - Leaf sets update:
            - The killed node's left_leaf_set.max is updated in the place of the killed node for all the nodes in the right leaf set of killed node.
            - The killed node's right_leaf_set.min is updated in the place of the killed node for all the nodes in the left leaf set of killed node.
        - Routing table update:
            - Firstly, for all the nodes in node_list, the killed node is removed in their routing tables if an entry exists.
            - Secondly, each of the neighboring nodes of the killed node in the routing tables of each node from which the killed node was removed, is asked for a replacement node generally which lies close to the position of killed node in the routing table.
            - Finally, all the nodes whose routing tables had the killed node, will be updated with the new replacement node obtained.
    - The parameters chosen for the implementation are:
        - b = 2

- Number of digits in node id = 8

- Whenever one or more nodes leave the network, we cannot guarantee the delivery of the message and hence the convergence also can't be attained. In such cases when we try to update the tables for failed nodes, resulting in an inconsistent state.

OBSERVATIONS:

No. of Nodes	No. of Requests	No. of Failed Nodes	Avg. No. of Hops
18	10	2	2.5875
100	10	20	2.0262
100	50	20	2.0135
100	100	30	1.8308
200	50	60	1.8944
200	100	60	1.6121
500	10	100	1.4527
750	10	300	1.0626
1000	10	300	1.0901
2500	10	1000	1.0
2500	2	500	1.0062
5000	2	1000	1.0111

Largest number of nodes and number of requests is 5000.

Even though the average hop count fluctuates due to node failures, the destination arrival is always guaranteed.

NOTES:

1. We have made use of the Matrix library:
    a. https://hexdocs.pm/matrix/api-reference.html . The dependencies for the same has been added in the mix.exs file. The dependency can be fetched by running the command **mix deps.get**, if not added automatically.