Group Members:
Joseph S Rayan, (UFID: 61022245), josephsrayan@ufl.edu
Sai Krishna Anugu, (UFID: 42266064), saikrishnaanugu@ufl.edu

Contents of project2.zip:
Project2/ebin/mainpushsum.beam
Project2/ebin/maingossip.beam
Project2/ebin/topology.beam
Project2/src/topology.erl
Project2/src/maingossip.erl
Project2/src/mainpushsum.erl

Project Overview:

This project aims to disseminate information to other nodes with the usage of different architectures, both for the method of spreading information as well as the data that needs to be used to determine that all nodes has successfully received the information that is being spread.

Execution Details:

1) For Gossip algorithm,
    a. To execute the convergence for any Topology,
        i. maingossip:start(NumberofNodes, “full”).
        ii. maingossip:start(NumberofNodes, “line”).
        iii. maingossip:start(NumberofNodes, “2dgrid”).
        iv. maingossip:start(NumberofNodes, “imp2d”).
    b. Output format:
        i. Convergence time is “” milliseconds
        ii.
```
27> maingossip:start(100,"full").

start time is {11831275,3236}true

convergence time is 11 milliseconds
```
        iii.
```
9> maingossip:start(50, "2dgrid").

Num of Nodes rounded off to 49
start time is {1532851,11051}true

convergence time is 6 milliseconds
```
2) For Push-Sum algorithm,

a. To execute the convergence for any Topology,
    i. mainpushsum:start(NumberofNodes, “full”).
    ii. mainpushsum:start(NumberofNodes, “line”).
    iii. mainpushsum:start(NumberofNodes, “2dgrid”).
    iv. mainpushsum:start(NumberofNodes, “imp2d”).
b. Output format:
    i. Convergence time is “” milliseconds
    ii.

```
30> mainpushsum:start(10,"full").

start time is {11863772,28340}<0.16651.0>

convergence time is 1 milliseconds
```

    iii.

```
7> mainpushsum:start(50, "imp2d").

Num of Nodes rounded off to 49
start time is {1417058,118777}<0.30324.0>

convergence time is 21 milliseconds
```

What is Working

1) We can achieve full convergence for all topologies specified but with varying nodes values
    a. For the line topology, the convergence time increases drastically when more nodes are added.

2) Push-Sum Algorithm based gossip takes longer to converge as compared to the conventional Gossip Algorithm

3) Among the topologies, **Line** topology takes the longest to converge as the nodes are increased.

    a. From our observation, this is to be expected as the number of neighbors that the gossip can be spread is always at most 2 neighbors

4) For most Topologies, we were able to test with more than 2000 nodes
    a. With Line Topology however, where even going above 150 nodes was resulting in convergence times above 500seconds and counting.

Implementation Design Considerations

1) PID List
    a. All PIDs were stored in a list at the time of being spawned
    b. The PID list was used by all topologies to determine the neighbors of each PID
    c. For 2Dgrid and Imp2D, list was also used but an algorithm was implemented to impose the List into 2D array to determine the 2D neighbors for each index in the normal

2) Neighbors PID List
    a. The Neighbors list was determined based on the Topology given in the input.
    b. Neighbors list was queried each time a PID had to choose a neighbor to send the gossip.

3) Termination of Nodes
    a. As a design decision, the neighbors were not immediately terminated upon reaching the maximum required count for the individual algorithms mentioned.
    b. The Neighbors will continue to send gossip to other neighbors until all neighbors in the list have converged.
    c. Following which, when the final neighbor has converged, the '**killswitch'** function will then proceed to kill all processes and stopping the execution of the gossip program

Timing Measurement

1) Statistics Library was used to determine the convergence time. Using **statistics(wall_clock),** we were able to calculate the time from when all the actors were spawned, till the time it took for neighbors to reach its maximum count needed for convergence.