Kyle Fauerbach

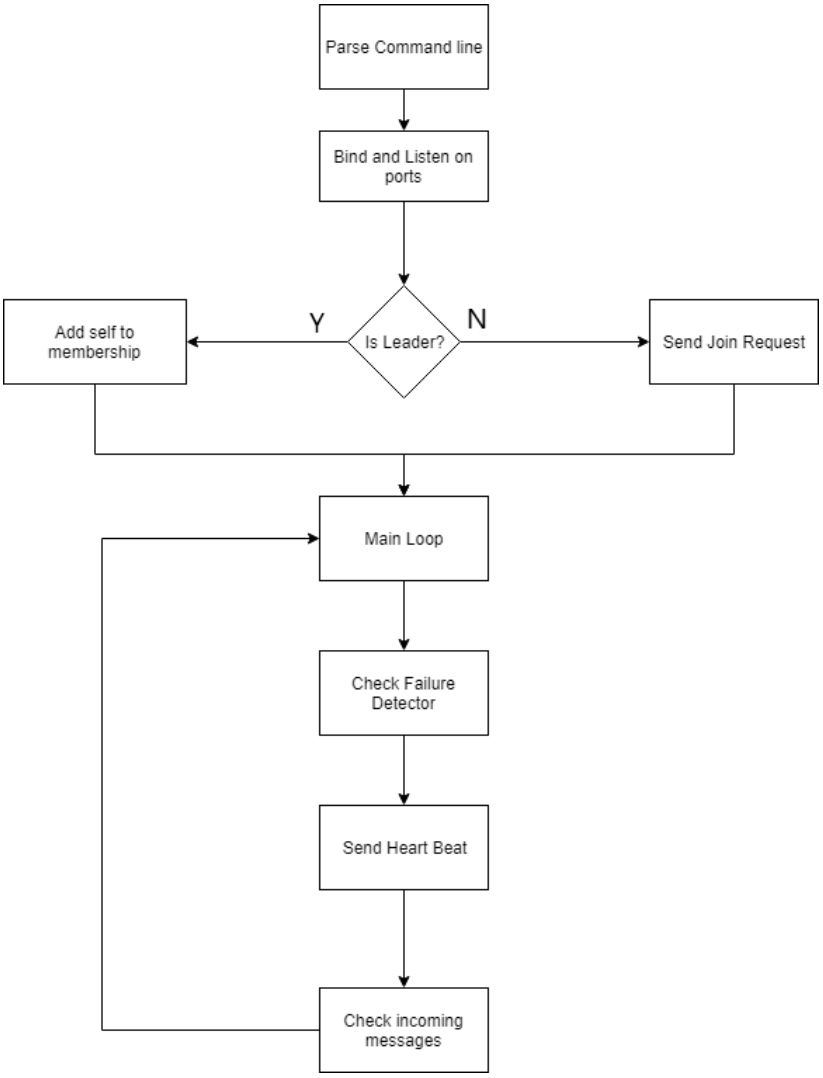
CS7610 Fall 2018

Project 2 Report

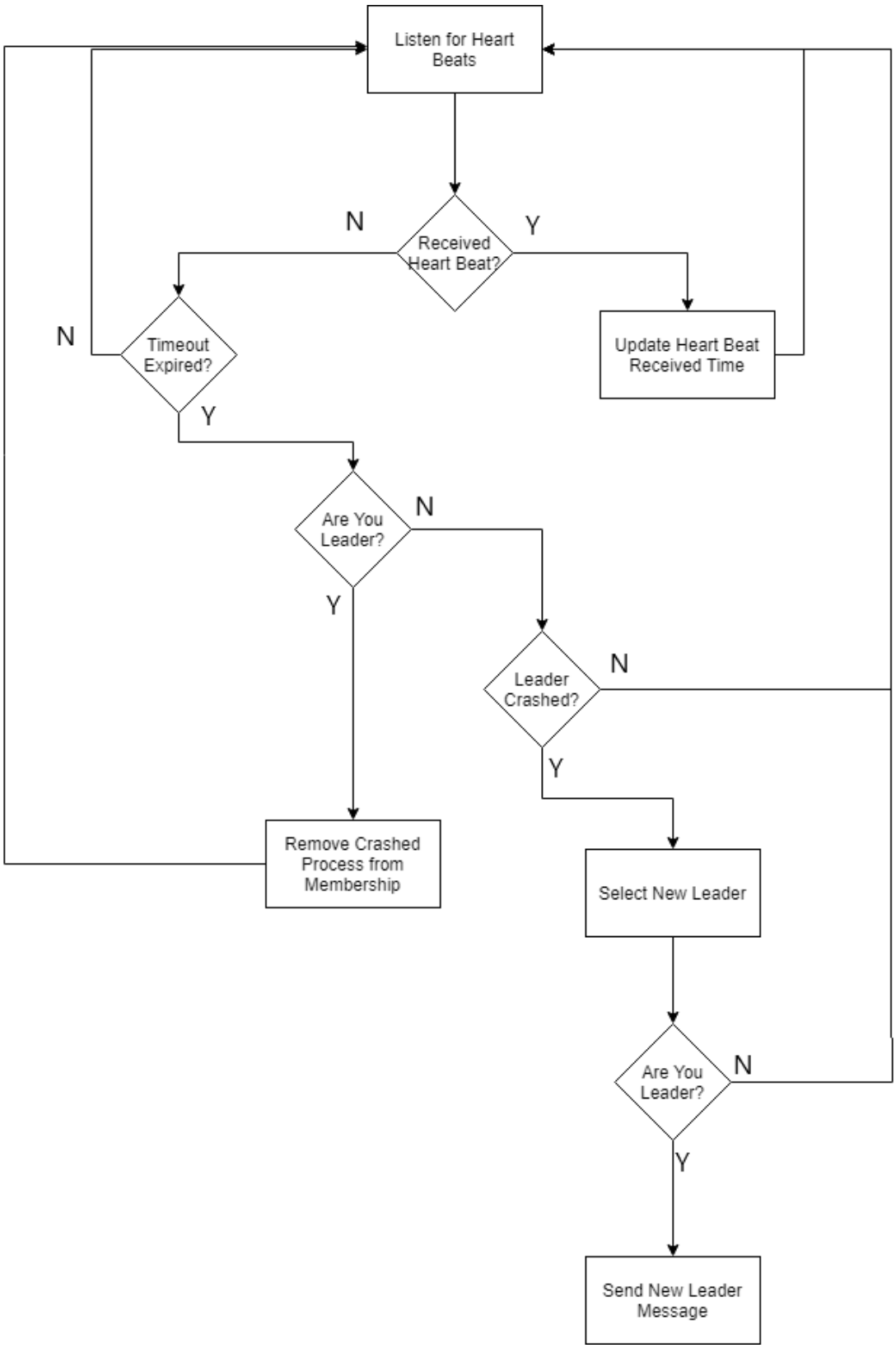
System Architecture

All code is written in C ad depends mainly on core libraries. Docker container testing completed on Ubuntu 18.04. Compilation and manual testing was also completed on the CCIS servers. Scripts to run all 4 test cases using Docker containers is included. See README for more documentation.

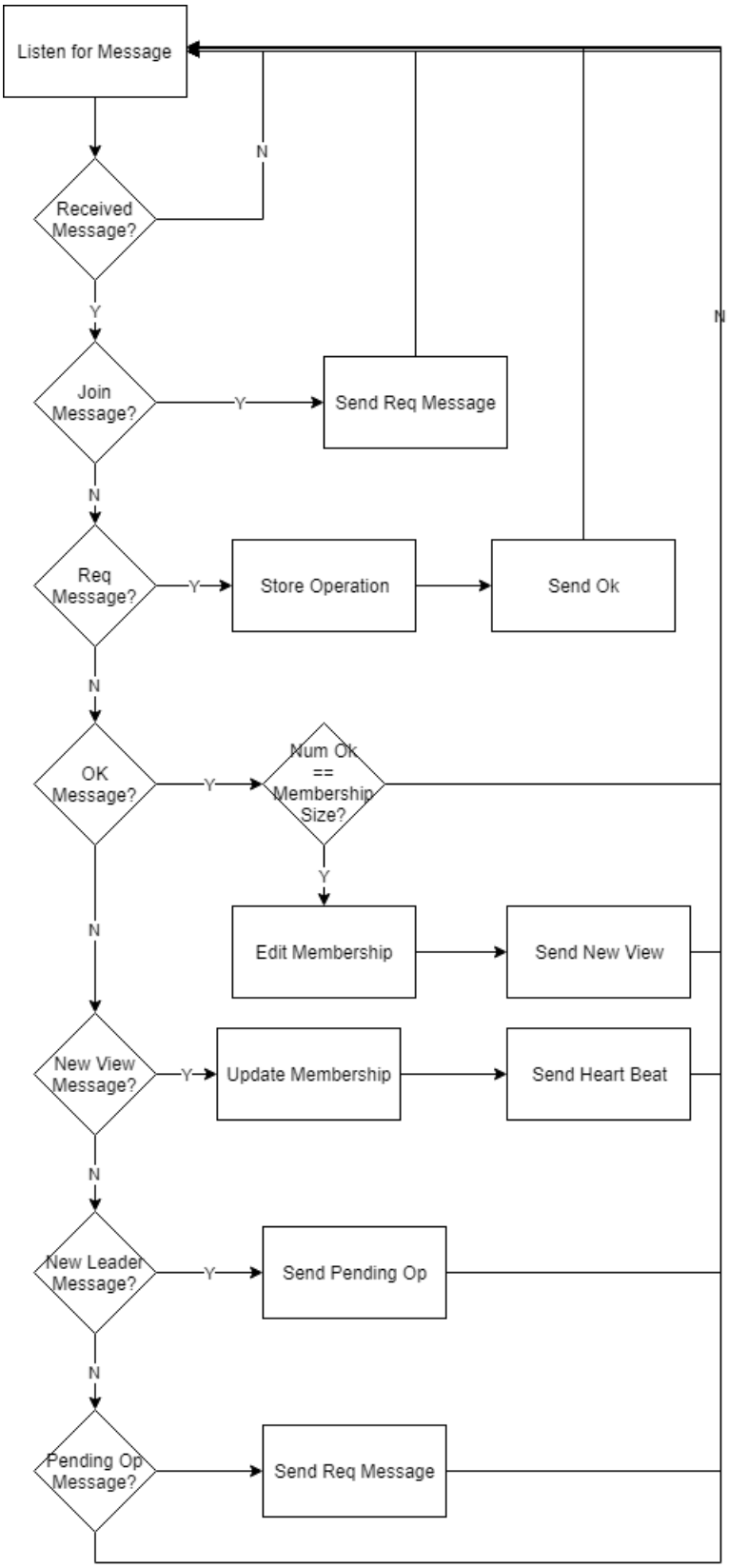
State Diagrams



High Level Overview



Check Failure Detector



Check Incoming Message

Design Decisions

All communication except for the failure detector is done over TCP, the failure detector is done with UDP. Testing instrumentation is done using the -t flag and specifying a level for the corresponding test. Further explanation of this can be found in the README. Selection of a new leader without a pending operation does not result in anything being printed to the terminal as all processes already know who the new leader will be and update their memberships implicitly. Leader ID is verified on receipt of new view to ensure that the leader did not crash while they were waiting to join.

Implementation Issues

Occasionally the failure detector will register a process to falsely declare that a process has failed. This happens very rarely during testing, and is resolved upon a subsequent run of the test. I believe this is due to load on the server and the reliance of the failure detector on gettimeofday(). This is only used for relative time within a process so clocks don't need to be synced but if the scheduler causes our process to block for too long this might be an issue. Increasing the timeout on the failure detector should fix this problem.