**Summary:**

This java program is an implementation of Ring Coordinator Election Algorithm for the course CSE 5306 Spring 2014 based on the requirement given in Project 2.

Designed and Developed by Samuel Benison Jeyaraj Victor.

Email ID: sambenison66@gmail.com

**Instructions:**

1. Run the java program (preferably in Eclipse) as many times as you want in order to decide the number of processes (restricted to the maximum count of 5)
2. Start the Ring process by clicking ‘Manual Election’ in any one of the process gui.
3. Once the election is completed, the elector will update the coordinator to all other process in the ring.
4. Now the ‘ALIVE’ message is sent constantly into the ring from each process (THIS IS THE OFFICIAL MESSAGE) and get the confirmation from coordinator in order to keep the ring process in track
5. If any one of the non-coordinator process is crashed (by clicking Crash button manually), that process will be by-passed by all other active process.
6. If the coordinator process is crashed, then the latest message validation will receive negative status for ‘ALIVE’ message, so it will automatically invoke the election process and find the new coordinator
7. Any crashed process can be brought back into the ring structure (by clicking the Reset button), and it will automatically invoke the election process to make sure that the coordinator is up to date.
8. If the ALIVE message is not went down in any of the process, click the Refresh button manually from any one of the process to start sending the ALIVE token process again.

**Special Features:**

1. If any process has been exited by closing the window, the previous processor will automatically deduct the exit and it will invoke an automatic election to update the active process.
2. If a process is crashed (for example the 2nd processor), you can run the java program one more time (6th time), this time it will be allowed and the 6th processor will take the 2nd processor’s position since it is already crashed. But as per the coordinator, the 2nd processor is updated as crashed and it has not been restarted yet. So the new 2nd processor will invoke an automatic election in order to get itself into the ring as processor 2. Now if you try to Restart the old processor 2, the port will not be available for it, the system will automatically close the old processor 2.
3. Multiple concurrent elections are possible. Try to do ‘Manual Election’ from 2 processors at a time (for example 2nd and 5th processor) and both the processor will select the one eligible coordinator correctly and update others. (Timer is will also initiate the election concurrently, but to replicate it, the waiting time is huge)