**Aim** – To implement Election algorithm(Bully algorithm) in java.

**Theory** – Distributed Algorithm is an algorithm that runs on a distributed system. Distributed system is a collection of independent computers that do not share their memory. Each processor has its own memory and they communicate via communication networks. Communication in networks is implemented in a process on one machine communicating with a process on another machine. Many algorithms used in the distributed system require a coordinator that performs functions needed by other processes in the system.

Election algorithms are designed to choose a coordinator.

Election Algorithms: Election algorithms choose a process from a group of processors to act as a coordinator. If the coordinator process crashes due to some reasons, then a new coordinator is elected on other processor. Election algorithm basically determines where a new copy of the coordinator should be restarted. Election algorithm assumes that every active process in the system has a unique priority number. The process with highest priority will be chosen as a new coordinator. Hence, when a coordinator fails, this algorithm elects that active process which has highest priority number. Then this number is send to every active process in the distributed system. We have two election algorithms for two different configurations of a distributed system.

1. The Bully Algorithm – This algorithm applies to system where every process can send a message to every other process in the system. Algorithm – Suppose process P sends a message to the coordinator.

If the coordinator does not respond to it within a time interval T, then it is assumed that coordinator has failed.

Now process P sends an election messages to every process with high priority number.

It waits for responses, if no one responds for time interval T then process P elects itself as a coordinator.

Then it sends a message to all lower priority number processes that it is elected as their new coordinator.

However, if an answer is received within time T from any other process Q,

(I) Process P again waits for time interval T’ to receive another message from Q that it has been elected as coordinator.

(II) If Q doesn’t responds within time interval T’ then it is assumed to have failed and algorithm is restarted.

**Conclusion** – Successfully Implemented Bully Election algorithm in java.