

## Assignment No: 6

Aim: Implement Bully & Ring Algorithm for leader election.

Tools & Environment: C++ Programming Environment.

### Theory:

Election algorithms choose process from group of processors to act as co-ordinators. If the co-ordinator process crashes due to some reasons, new co-ordinator is elected. Election algorithm basically determines where new copy of co-ordinator should be restarted. The process with highest priority will be chosen as new co-ordinator. Hence when a co-ordinator fails, this algorithm elects that active process which has highest priority no. We have two election algorithms for two different configurations of distributed system.

⇒ **Bully Algorithm:** This algorithm applies to system where every process can send message to every other process in system.

### Algorithm:

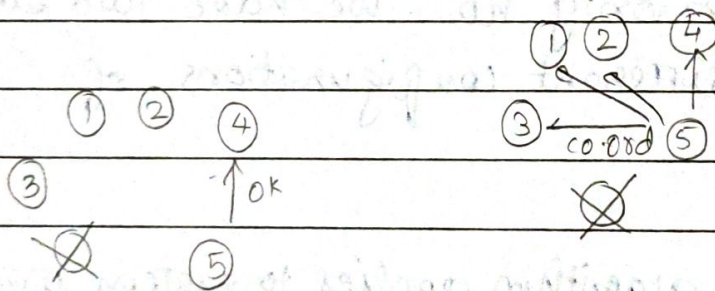
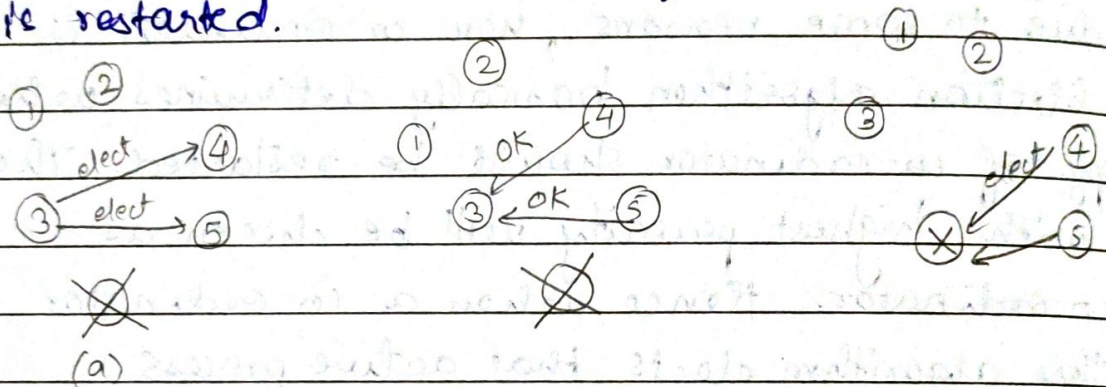
Suppose process P sends message to co-ordinator. If the co-ordinator does not respond to it within time T, then assume co-ordinator has failed. Now P sends election messages to every process with high priority no.



However if an answer is received within time  $T$  from other process  $q$ ,

i) Process  $P$  again waits for time interval  $T$  to receive another message from  $q$  that has been elected as co-ordinator.

ii) if  $q$  doesn't respond within time interval  $T$ , then it is assumed to have failed and algorithm is restarted.



II) Ring Algorithm: This algorithm applies to systems organised as rings. we assume these links are unidirectional & every process can message only towards its right. data structure used: active list, a list has priority no of all active processes in system.



### Algorithm:

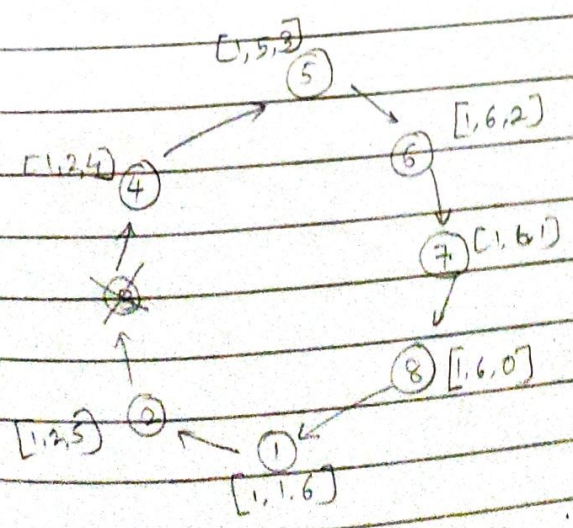
If process P1 detects a co-ord failure, it creates new active list which is empty initially. It sends election message to neighbours on right and adds no 1 to active list.

If process P2 receives message elect from processes on left, it responds in 3 ways:

- 1) If message received doesn't contain 1 in active list then P1 adds 2 to its active list forward message.
- 2) If this is first election message, received or sent P1 creates new active list with no 1 & 2
- 3) If process P1 receives its own election message, then active list P1 now contains no of all active processes in system. Now process P1 detects highest priority no from list and elect as new co-ordinator

Time complexity :  $O(n^2)$

space complexity :  $O(n)$





→ Conclusion: The bully algorithm is a type of election algorithm which is mainly used for choosing a co-ord. Hence in distributed system, we need some election algorithm such as bully & ring to get a co-ord that performs functions needed by other processes.