

Algorithms for choosing a Coordinator Node

Ring-based algorithm

- ① We arrange all processes/nodes in a logical ring and assume no failures. All nodes are set to be "non-participants".
- ② Initiating process makes himself "participant" and sends its identifier in an election message to its neighbour.
- ③ Receiver compares the received with its own identifier.
if ($ID_{message} > ID_{own}$): forward message unchanged
else: replace $ID_{message}$ with ID_{own} and forward message
Receiver is now a participant.
- ④ If ($ID_{message} = ID_{own}$)
The respective node has won and becomes the coordinator
- ⑤ Coordinator sends an elected message around the ring to tell everyone ("I'm the coordinator, this is my ID").

Bully algorithm

- ① P sends an election message to all processes/nodes with higher numbers. (P is the initiator).
- ② If no one responds P wins the election, becomes coordinator and informs all other nodes via a coordinator message
- ③ If a higher-numbered node Q answers P via an answer message, P doesn't win and Q begins the election process again. This repeats until one process wins.

Note 1: Failures are tolerated since wait-times for nodes can be controlled via time-outs.

Note 2: All nodes must know about the IDs (e.g. IP addresses) of the participating nodes and can communicate with them.