**Maekawa’s algorithm:**

1. Configure the test bed
   1. Setting up the network systems – need their Name or IP addresses
   2. Set up the shared file system
      1. Total of # of nodes
      2. Node addresses /Process ids
   3. Create the quorums and log the quorums – the quorums will actually be stored in class variables

Scenario 1 – assigning the msg in s predefined order

* 1. Read the shared file to find which process sends what to which process etc.
  2. Setup the counter for logical clock (global or local depending) to increment every time some msg is shared
  3. Msg queue is set – predefined case 1 – start sending msg and output:
     1. Msg sent from process X to process Y at time Z timestamped W
     2. Msg deleivered to Y from X with ……
     3. Have some way to show the sequence of events happening and also some way to show the ones that happen at the same time
     4. Show when the msg are over and system is idle

New Algorithm:

1. Configure the test bed
   1. Setting up the network systems – need their Name or IP addresses
   2. Set up the shared file system
      1. Total of # of nodes
      2. Node addresses /Process ids
   3. Create the quorums and log the quorums – the quorums will actually be stored in class variables

Scenario 1 – assigning the msg in s predefined order

* 1. Read the shared file to find which process sends what to which process etc.
  2. Setup the counter for logical clock (global or local depending) to increment every time some msg is shared
  3. Msg queue is set – predefined case 1 – start sending msg and output:
     1. Msg sent from process X to process Y at time Z timestamped W
     2. Msg deleivered to Y from X with ……
     3. Have some way to show the sequence of events happening and also some way to show the ones that happen at the same time
     4. Show when the msg are over and system is idle