5. Implement token ring based mutual exclusion algorithm.

* Elements
  + Token
    - initialized at zero
    - will rotate in ring to find the sender
    - after sender is found , token will append the data in it
    - now it will find receiver for the data in a ring
    - will go to sender and confirm that data is sent
    - token will now rotate till another request is received
  + Sender
  + Receiver
  + Data
* A site in distributed system do not have complete information of state of the system due to lack of shared memory and a common physical clock. Message passing is a way to implement mutual exclusion. The token ring algorithm ensures mutual exclusion by using a special token that circulates among the nodes in a logical ring. For example, we can identify each process by its machine address and process ID to obtain an ordering. Using this imposed ordering, a logical ring is constructed in software. Each process is assigned a position in the ring and each process must know who is next to it in the ring.
* How mutual exclusion works
  + Single Token: There's only one token circulating in the ring.
  + Token Possession: Only the node holding the token is allowed to enter the critical section (send data).
  + Token Passing: Nodes pass the token to their logical successor in the ring after finishing their critical section or if they don't need it. process n passes it to process (n+1) mod ringsize.
* . Token Based Algorithm:
  + A unique token is shared among all the sites.
  + If a site possesses the unique token, it is allowed to enter its critical section
  + This approach uses sequence number to order requests for the critical section.
  + Each requests for critical section contains a sequence number. This sequence number is used to distinguish old and current requests.
  + This approach insures Mutual exclusion as the token is unique
* Mutual exclusion algorithm should satisfy
  + No deadlock
  + No starvation
  + Fairness
  + Fault tolerance
* Only one process has the token, and hence the lock on the resource, at a time. Therefore, mutual exclusion is guaranteed. Order is also well-defined, so starvation cannot occur. The biggest drawback of this algorithm is that if a token is lost, it will have to be generated. Determining that a token is lost can be difficult.
* Sender ==receiver , multiple sender , multiple receiver