

## **Part – A**

### **1) What is Binary Rendezvous?**

As said by the name, “binary” the communication between two processes are tracked and this study is called as Binary Rendezvous. In a Group Communication, there will exist a synchronous communication between any processes which are in asynchronous state at any point of time.

### **2) List the assumptions of Binary Rendezvous?**

- a) For the receive command, the sender must be specified. However, multiple receive commands can exist. A type check on the data is implicitly performed.
- b) Send and received commands may be individually disabled or enabled. A command is disabled if it is guarded and the guard evaluates to false. The guard would likely contain an expression on some local variables.
- c) Synchronous communication is implemented by scheduling messages under the covers using asynchronous communication. Scheduling involves pairing of matching send and receive commands that are both enabled. The communication events for the control messages under the covers do not alter the partial order of the execution.

### **3) Write the conditions in recording a global state**

How to distinguish between the messages to be recorded in the snapshot (either in a channel state or a process state) from those not to be recorded.

The answer to this comes from conditions C1 and C2 as follows:

Any message that is sent by a process before recording its snapshot, must be recorded in the global snapshot (from C1).

Any message that is sent by a process after recording its snapshot, must not be recorded in the global snapshot (from C2).

How to determine the instant when a process takes its snapshot. The answer to this comes from condition C2 as follows:

A process  $p_j$  must record its snapshot before processing a message  $m_{ij}$  that was sent by process  $p_i$  after recording its snapshot.

### **4) Explain the concept of Centralized algorithm for Total Ordering.**

When process  $P_i$  wants to multicast a message  $M$  to group  $G$ :  
send  $M(i, G)$  to central coordinator.

When  $M(i, G)$  arrives from  $P_i$  at the central coordinator:  
send  $M(i, G)$  to all members of the group  $G$ .

When  $M(i, G)$  arrives at  $P_j$  from the central coordinator:  
deliver  $M(i, G)$  to the application.

### **5) Explain the concept of Three-phase distributed algorithm in Total Ordering**

#### **Sender:**

**Phase I** In the first phase, a process multicasts the message  $M$  with a locally unique tag and the local timestamp to the group members.

**Phase 2** In the second phase, the sender process awaits a reply from all the group members who respond with a tentative proposal for a revised timestamp for that message M. The await call is non-blocking, i.e., any other messages received in the meanwhile are processed. Once all expected replies are received, the process computes the maximum of the proposed timestamps for M, and uses the maximum as the final timestamp.

**Phase 3** In the third phase, the process multicasts the final timestamp to the group.

#### **Receivers:**

**Phase 1** In the first phase, the receiver receives the message with a tentative/proposed timestamp. It updates the variable priority that tracks the highest proposed timestamp then revises the proposed timestamp to the priority, and places the message with its tag and the revised timestamp at the tail of the queue temp\_Q. In the queue, the entry is marked as undeliverable.

**Phase 2** In the second phase, the receiver sends the revised timestamp (and the tag) back to the sender. The receiver then waits in a non-blocking manner for the final timestamp

**Phase 3** In the third phase, the final timestamp is received from the multicaster. The corresponding message entry in temp\_Q is identified using the tag (and is marked as deliverable after the revised timestamp is overwritten by the final timestamp. The queue is then resorted using the timestamp field of the entries as the key. As the queue is already sorted except for the modified entry for the message under consideration, that message entry has to be placed in its sorted position in the queue. If the message entry is at the head of the temp\_Q, that entry, and all consecutive subsequent entries that are also marked as deliverable, are dequeued from temp\_Q, and enqueued in deliver\_Q in that order

#### **6) List the Variations of the Chandy–Lamport algorithm.**

- Spezialetti–Kearns algorithm
- Venkatesan’s incremental snapshot algorithm
- Helary’s wave synchronization method

#### **7) What is the use of Marker tag?**

A Marker tag is used to separate the message from a notification tag. Notification tag will give the current state of the process which sends a message. That is either the state that process is recorded or not recorded.

#### **8) Define Total Ordering (TO)**

For each pair of processes  $P_i$  and  $P_j$  and for each pair of messages  $M_x$  and  $M_y$  that are delivered to both the processes,  $P_i$  is delivered  $M_x$  before  $M_y$  if and only if  $P_j$  is delivered  $M_x$  before  $M_y$ .

#### **9) Define group communication**

Processes across a distributed system cooperate to solve a joint task. Often, they need to communicate with each other as a group, and therefore there needs to be support for group communication. A message broadcast is the sending of a message to all members in the distributed system.

Refining the notion of broadcasting, there is multicasting wherein a message is sent to a certain subset, identified as a group, of the processes in the system. At the other extreme is unicasting, which is the familiar point-to-point message communication.

**10) Define Lost and Delayed message.**

**Lost messages:**

A message which is sent by the sender is recorded and this node is fine, but on the other end the message has been received, after sometimes due to the failure of that the node, it has been restarted to its previous checkpoint so this cause the message which is received to be deleted.

**Delayed messages:**

Messages whose receive is not recorded because the receiving process was either down or the message arrived after the rollback of the receiving process, are called delayed messages.

**11) When will a delayed message will turned out to an orphan message?**

When the sending process gets failed (when its message is in transmission or delayed (due to some technical issues in the communication channel or the receiving process)) and started recovering from its failure. In this mean time the receiving process will receive the message which is already sent by the sending process. Thus in this scenario delayed message will turned to an orphan message.

**12) When do a process must send a message to other end, and the order in which the receiver must receive it, in order to maintain consistency.**

A sender must send the message before taking a checkpoint, and at the receiver end the message must be received after its consistent state.

**13) Define Check Point and Rollback Recovery**

Check Point – A Point up to which the state of the particular process is saved is called as checkpoint.

Rollback Recovery – When a process is down due to some error during the execution of particular task, surely it will affect the overall outcome of the environment. Therefore, to avoid this, we have to do a process to bring back the system. This process is called as Rollback, and it is done in which that the system would have been in a consistent state.

---

**14) Why is Rollback Recovery of distributed systems complicated?**

Due to the existence of cascading rollback in the distributed environment. If a particular process failed, then it must be restarted (Roll Back) for bringing back to a consistent state, but as this is an distributed environment, where many process coordinated to a global task, we have to keep an eye on the other process which are involved in that coordinated activity. Even though the other system is in a good condition, it has to roll back to its previous consistent state. (i.e.) Chain Reaction

**15) What is Domino Effect?**

Upon a failure of one or more processes in a system, these dependencies may force some of the processes that did not fail to roll back, creating what is commonly called a rollback propagation. This phenomenon of cascaded rollback is called the domino effect

**16) List the different types of messages in rollback recovery of process failure**

- In-transit messages
- Lost messages
- Delayed messages
- Orphan messages
- Duplicate messages

**17) Difference between consistent and inconsistent state, and explain how it occurs.**

**Provide a valid answer, with your own example and diagram**

**18) What do you mean by Interactions with outside world?**

A process of message flow out of the distributed environment is said as interactions with outside world.

**Eg:** The Money which is withdrawn by the customer in ATM, A Ticket which is taken by the passenger, A content which is printed by the printer.

In this type of communication, there a complexity to maintain the consistent state to recover, when there is failure happens.

**19) Write the advantages and disadvantages of Uncoordinated Check pointing.**

Advantages	Disadvantages
Each process can take its own checkpoint during the particular time instant, without permission/knowledge of other node.	The process of roll backing to a consistent checkpoint when a failure occurs in a process, which has its dependencies with another process.
Marinating Local checkpoints will make the process to be in its own consistent state and to do its own computation task with its slave process	Too many checkpoint (unnecessary)

**20) What is the use of piggyback protocol in communication-induced check pointing?**

**Piggyback-** information about a processes states, id, Etc.

When passing the communication with the piggyback information, will do a great work in that receiving process. It can able to know the current state of that sender process, to which process it has made a last computations and which event made it to send a message to its process. So that while recovering from a failure, it is easy to backtrack the states/processes.

**21) Define coordinated check pointing and Uncoordinated Check pointing.**

**Coordinated check pointing** – All the process will be involved in taking a checkpoint, which will be consistent across the entire system

**Un-Coordinated check pointing** - All the process will be involved in taking a checkpoint, within itself (Independent of other process). Thus making their system to be consistent

**22) List the events of log based rollback recovery**

- Deterministic and non-deterministic events
- Pessimistic logging
- Optimistic logging
- Causal logging

**23) How to find the consistent checkpoints in a distributed environment**

**Provide a valid answer, with your own example and diagram**

**24) List the techniques that avoid domino effect.**

**a) checkpoint-based**

- coordinated
- uncoordinated
- communication-induced

**b) Log-based**

- Deterministic and non-deterministic events
- Pessimistic logging
- Optimistic logging
- Causal logging