- Solution - 02 - DC
Weekly test -

VIT Vidyalankar Institute of Technology
(Accredited A+ by NAAC)
(Autonomous Institute Affiliated to University of Mumbai)

Mid Semester Examination

Branch	Date	Sem.	Roll No. / Exam Seat No.	Subject	Student's Signature	Junior Supervisor's Name and Sign
MPH	19/3			DC		

Question No.	A	B	C	D	E	F	G	H	Total	Total out of (20 /30 / 40)
1										
2										
3										
4										

Examiners Signature	Student's Sign (After receiving the assessed answer sheet)

Week Test no - 02
Subject - DC
Date - 19/03/24.

Solution:-

Q1. Attempt any 5 (2 marks each).

1) Which are design issue in group communication.

→ A number of design issues for group communication are

(1) closed group Vs open group
(2) Peer group Vs hierarchal group
(3) centralised group membership Vs distributed membership

i) Closed groups - only group members may send a message to the group. This is useful when multiple processes need to communicate with others in solving a problem, such as in parallel processing applications. This alternative is open groups, where non-members can send a message to a group. An example use of this type of group is an implementation of a replicated server.

Peer groups vs. hierarchical groups - with peer groups, every member communicates with each other. The benefits are that this is a decentralized, symmetric system with no point of failure. However decision-making may be complex since all decisions must be made collectively

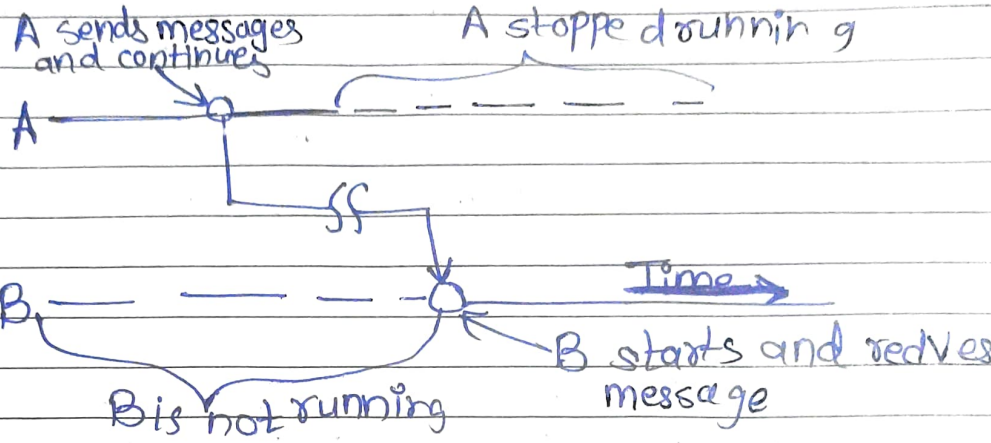
Centeralized group membership vs distributed membership
If the control of group membership is centeralized, we will have one group server that is responsible for getting all membership requests. It maintains a database of group members.

b) In the context of distributed systems, compare message oriented and stream oriented communication?

	Message-oriented communication	Stream-oriented communication
1	UDP (user-datagram protocol) uses message-oriented communication	TCP (transmission control protocol) uses stream-oriented communication
2	Data is sent by applications in discrete packages called messages	Data is sent with no structure
3	Communication is connectionless, data is sent without acknowledgment	It is reliable, data acknowledged. High overhead
4	Low overhead	Lost data is reframed automatically
5	No flow control	High overhead
6	Retransmission is not performed	Lost data is reframmed automatically.

c) In Message-oriented explain transient asynchronous communication.

→ Since the message is transient, both entities must be running. Also, the sender doesn't wait for responses because it is asynchronous. UDP is an example



d)

e) ~~AA~~ In the context of distributed systems, what is the primary purpose of Mutual Exclusion algorithm

→ i) No Deadlock - Two or more sites should not endlessly wait for any message that will never arrive

ii) No Starvation - Every site who wants to execute critical section should get an opportunity to execute it in finite time. Any site should not wait indefinitely to execute critical section while other sites are repeatedly executing critical section.

iii) Fairness - Each site should get a fair chance to execute critical section. Any request to execute critical section must be executed in the order they are made, i.e., critical section execution requests should be executed in the order of their arrival in the system

iv) Fault tolerance - In case of failure, it should be able to recognize it by itself to continue functioning

without any disruption.

f) What is objective of network time protocol (NTP) Algorithm in a distributed system.

→ i) The NTP client initiates a time-requig exchange with the NTP server.
ii) The client is then able to calculate the link delay and its local offset and adjust its local clock to match the clock at the server's computer.
iii) As a rule, six exchanges over a period of about five to 10 minutes are required to initally set the clock.

g) When discussing clock synchronization, what does a "physical clock" typically represent?

→ A computer clocks typically has a quartz crystal, a counter register, and a constant register. The constant register stores a constant value dependent on the quartz crystal's oscillation frequency.
The counter register decrements by 1 for each quartz crystal oscillation.
When the counter register reaches zero, an interrupt is issued, and its value is reinitialized to the constant register. Each interrupt is termed a clocktick.

h) Which are types of group communication used in distributed system?

→ i) Broadcast communication - It is used when host process tries to communicate with every process in a distributed system at the same time. Broadcast communication comes in handy when a common stream of information is to be delivered to every process in the most efficient manner

possible. Since it does not require any processing whatsoever communication is very fast in comparision to other modes of communication. However, it does not support many processes and cannot treat a specific process individually.

ii) Multicast Communication - It is used when the host process tries to communicate with a designated group of processes in a distributed system at the same time. This technique is mainly used to find a way to address problem of a high workload on host system and redundant information from process in system. Multitasking can significantly decrease time taken for message handling.

iii) Unicast communication - It is used when the host process tries to communicate with a single process in a distributed system at the same time. Although, same information may be passed to multiple processes. This works best for two processes communicating as only it must treat a specific process only. However, it leads to overheads as it must find exact process and then exchange information/data

Q2) Attemp any 2

a) Discuss the concept of Berkeley algorithm with a suitable example.

→ 1) Berkeley's Algorithm is a clock synchronization technique used in distributed systems.

2) The algorithm assumes that each machine node in the network either doesn't have an accurate time source or doesn't posse a UTC server. This algorithm overcomes limitation of faculty clock and malicious interference in passive time server and also overcomes limitation of active time-server algorithm.

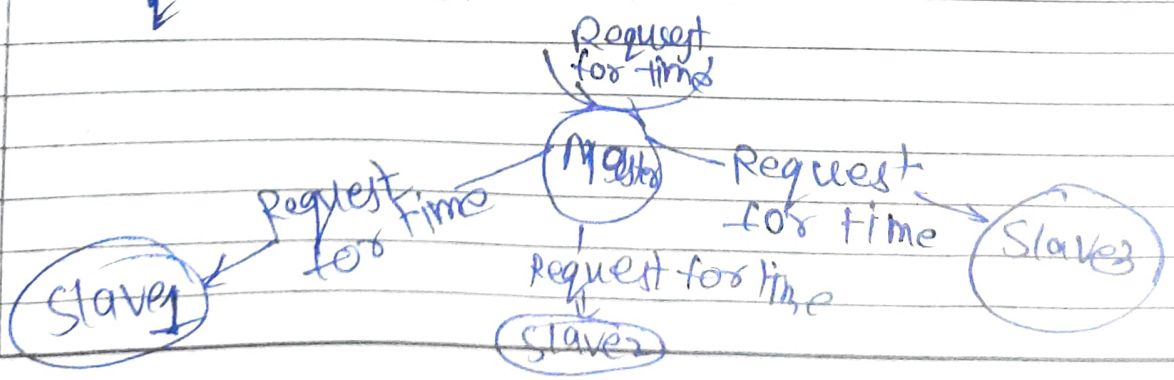
3) Time-server periodically sends a request message "time = ?" to all nodes in the system. Each node sends back its time value to the time-server.

4) Time-server has an idea of message propagation to each node and readjust the clock values in reply message based on it. Time server takes an average of other computer clock's value including its own clock value and readjusts its own clock accordingly

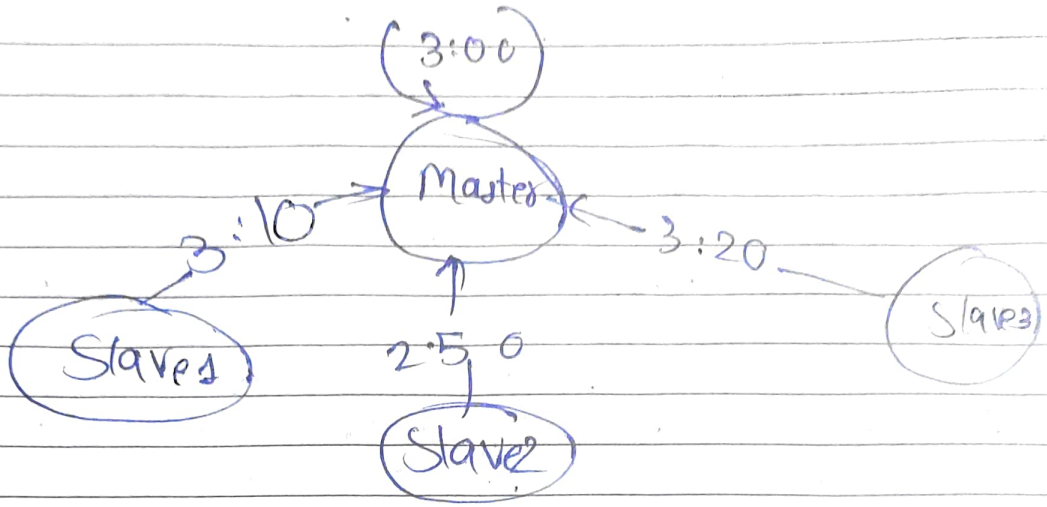
5) It avoids reading from unreliable clocks. For readjustment, time-server sends the factor by which other nodes require adjustment.

6) The readjustment value can either be +ve or -ve

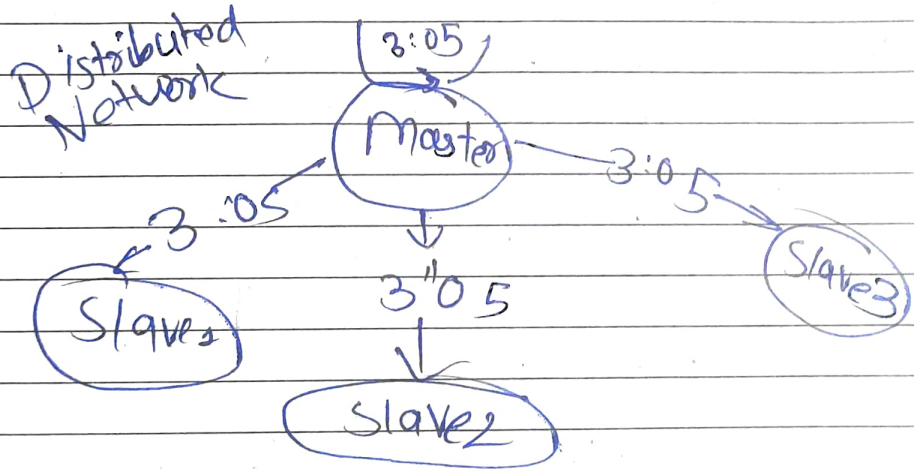
7) The following illustrate how the master sends request to slave nodes



Illustrates how slave nodes send back time given by system clock



Berkeley's Algorithm



b) Explain the significance of clock synchronization in distributed systems and discuss two methods used for achieving it.

→ clock synchronization is a method of synchronizing the clock values of any two nodes in a distributed system with the use of an external reference clock or internal clock value of the node.

A) Global Averaging Algorithm-

1. Each node in the system broadcast its local clock time in the form of special 'resync' message when its local time is $t + \Sigma R$ where I is for interrupt time and R includes the number of nodes in the system, maximum drift rate, etc
2. After broadcasting, the clock process of the node waits for some time period t'.
3. During this period, it collects all 'resync' message from other nodes and records its receiving time based on its local clock time.
4. At the end of the waiting period, it estimates the skew of its own clock with respect to all other nodes.

B) Localized Averaging Algorithm.
1. This algorithm overcomes the limitation of scalability of global averaging algorithms.
2. Nodes of the system are arranged logically in a specific pattern like a ring or grid.
3. Periodically each node exchanges its local clock time with its neighbors and then sets its clock time to the average of its own clock time and the clock time of its neighbours.

b) Dissuss concept to ring algorithm.

→ Algorithm uses a ring for its election but does not use any token. In this algorithm it is assumed that the processes are physically or logically ordered so each processor knows its successor.
i) When any process notices that a coordinator is not functioning, it builds an ELECTION message containing its own process number and sends the message to its successor.

the successor is down the sender skip over the successor and goes to next member along the ring until a process is located.

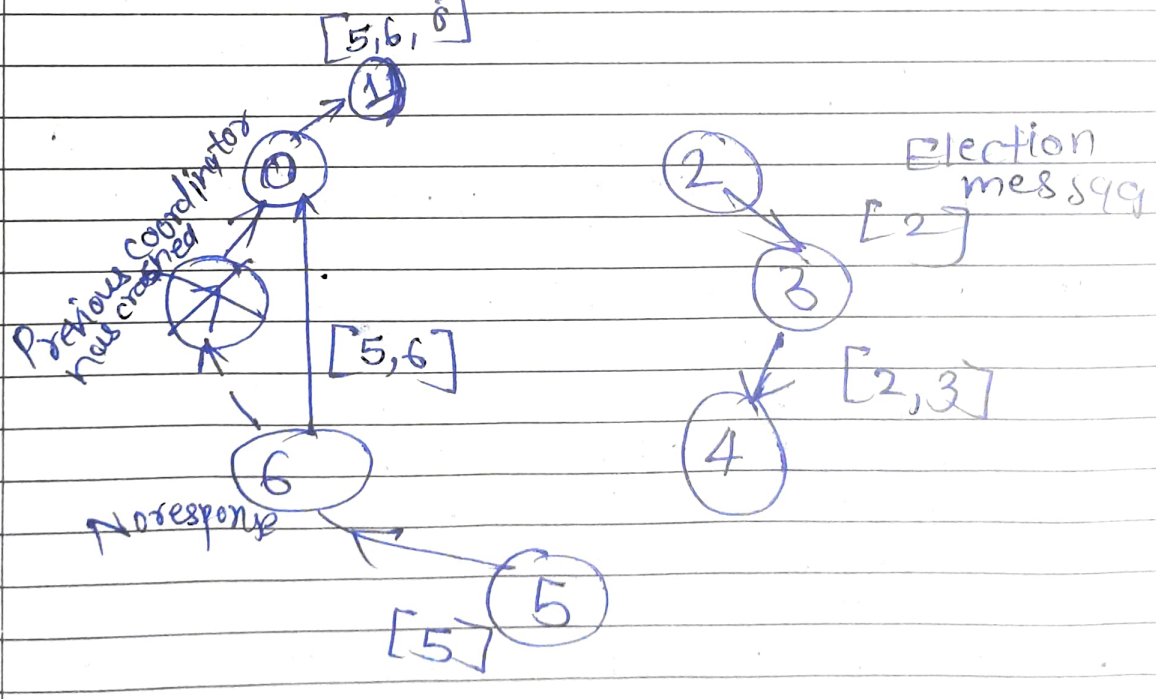
ii) At each step the sender adds its own process number to the list in the message making itself a candidate to elected coordinator.

iii) The message gets back to the process that started it and recognize this event as the message consists of its own process number.

iv) At that point the message type is changed to COORDINATOR and circulated once again to inform everyone who the coordinator is and who are the new members.

v) The coordinator is selected with the process having highest number

vi) When this message is circulated once it is removed and normal work is preceded



Q3) Attempt Any one.

a) Outline the fundamental requirements of Mutually Exclusion algorithms and explain the significance of performance measure in evaluating their effectiveness.

→ 1) No Deadlock: Two or more sites should not endlesly wait for any message that will never arrive

2) No starvation: Every site who wants to execute critical section should get an opportunity to execute it in finite time.
Any site should not wait indefinitely to execute critical section while other site are repeatedly executing critical section.

3) Fairness: Each site should get a fair chance to execute critical section. Any request to execute critical section must be executed in order they are made i.e, critical section execution requests should be executed in the order of their arrival in the system.

4) Fault tolerance. In case of failure, it should be able to recognize it by itself to continue functioning without any disruption

b) Discuss the concept of election algorithms in distributed system.

→ 1. Distributed system is a collection of independent computers that do not share their memory. Each processor has its own memory, and they communicate via communication networks.

2. Communication in networks is implemented in a process on one machine communicating with

a process on another machine. Many algorithm used in distributed system require a coordinator that performs functions needed by other process in the system.

3. Election algorithms are designed to choose a coordinator.

4. Election algorithm choose a process from group of processes to act as a coordinator. If the coordinator process crashes due to some reasons, then a new coordinator is elected on other processor.

5. Election algorithm basically determines where a new copy of coordinator should be restarted.

6. Election algorithm assumes that every active process in the system has a unique priority number.

7. The process with highest priority will be chosen as a new coordinator. Hence, when a coordinator fails, this algorithm elects that active process which has highest priority number. Then this number is sent to every active process in the distributed system. We have two election algorithms for two different configuration of distributed system - bully, Ring.

8) b) →

Which are crucial aspects of clock synchronization in distributed system.

- clock synchronization involves several critical aspects.

1. Accuracy
2. Precision
3. Fault tolerance
4. Scalability

Performance
Security
Compatibility.

⇒ Accuracy → Ensuring that the local clock of individual nodes are synchronized as closely as possible to a common reference time.

⇒ Precision → Achieving consistent and reliable timing across all nodes, allowing for precise ordering

⇒ Fault tolerance → Implementating mechanism to handle failures or discrepancies in clock synchronization. Fault detection algorithms

⇒ Scalability → Ensuring that clock synchronization solution can scale to accomodate large number of nodes in the distributed system without sacrificing accuracy performance.