UNIT -1

TWO MARKS

1. What is a distributed system?

A distributed system is one in which components located at networked computers communicate and coordinate their actions only by passing messages. The components interact with each other in order to achieve a common goal.

2. What do you mean by message passing?

Message passing is a fundamental mechanism for communication in distributed systems. It enables processes or nodes to exchange messages and coordinate their actions. There are several types of message-passing models, including synchronous, asynchronous, and hybrid approaches.

3. Define Distributed Program?

A computer program that runs within a distributed system is called a distributed program, and distributed programming is **the process of writing such programs**.

4. What do you mean by synchronous and asynchronous execution?

Asynchronous is a non-blocking architecture, so the execution of one task isn't dependent on another. Tasks can run simultaneously. Synchronous is a blocking architecture, so the execution of each operation depends on completing the one before it. Each task requires an answer before moving on to the next iteration.

5. List out the features of distributed systems?

- Performance. ...
- Scalability. ...
- High availability. ...
- Data integrity. ...
- High reliability. ...
- Security. ...
- User mobility.

6. Write down the principles of distributed systems?

Distributed file systems are an important part of any organization's data storage and access needs. The design of the system should be based on the principles of scalability, availability, reliability, performance, and security.

7. State the objectives of resource sharing model?

The primary objective of resource sharing is to maximize the resource base,i.e., collection, staff, infrastructure, as well as services of the participating libraries. They would be benefited by the resources of other libraries adding to their own resources.

8. What are the significant consequences of distributed systems?

The components of a distributed system interact with one another in order to achieve a common goal. Three significant challenges of distributed systems are: maintaining concurrency of components, overcoming the lack of a global clock, and managing the independent failure of components.

9. What are the challenges of distributed systems?

The main challenges of distributed system are:

Heterogeneity
Openness
Security
Scalability
Failure handling
Concurrency
Transparency
Quality of service

10. Define Transparency. What are its types?

Transparency is defined as the concealment from the user and the application programmer of the separation of components in a distributed system, so that the system is perceived as a whole rather than as a collection of independent components.

Its types are:

Access transparency
Location transparency
Concurrency transparency
Replication transparency
Failure transparency
Mobility transparency
Performance transparency
Scaling transparency

11. What is the need of openness in distributed system?

Openness: The openness of the distributed system is determined primarily by the degree to which new resource-sharing services can be made available to the users. Open systems are characterized by the fact that their key interfaces are published

12. List any two resources of hardware and software, which can be shared in distributed systems with examples

Five types of hardware resource and five types of data or software resource that can usually be shared are printer,plotter,storage space,cd drive,dvd drive,processing power. For example printer which takes graphics and texts from the computer and later it gets transferred into a paper which is of standard size.

13. Differentiate between buffering and caching

Buffering is a process of temporarily holding data in memory or a buffer before writing it to a permanent storage location. Caching is a process of temporarily storing data in memory for quick access or retrieval. Cache stores copy of the data. Cache is in processor, and can be also implemented with ram and disk.

14. Differentiate between synchronous and asynchronous execution?

Synchronous code executes one line of code after the other, while asynchronous code allows multiple lines of code to run at the same time. Asynchronous code can be much more efficient than synchronous code for certain types of programs, but it is also more complex and harder to debug.

15. What is the role of middleware in a distributed system?

Middleware is an intermediate layer of software that sits between the application and the network. It is used in distributed systems to provide common services, such as authentication, authorization, compilation for best performance on particular architectures, input/output translation, and error handling.

16. Name some services and examples of middleware?

Common middleware examples include database middleware, application server middleware, message-oriented middleware, web middleware, and transaction-processing monitors.

17. What is open in distributed system?

An Open Distributed System is made up of components that may be obtained from a number of different sources, which together work as a single distributed system.

In 1988 the International Standards Organization (ISO) began work on preparing standards for Open Distributed Processing (ODP).

18. Describe what is meant by a scalable system?

A system is scalable when it has the capacity to accommodate a greater amount of usage. Some systems aren't at all scalable, and can only handle exactly the amount of usage they were designed for. Scalable systems can handle extra usage, but theircapacity varies.

19. What is replication transparency?

Replication transparency is the ability to create multiple copies of objects without any effect of the replication seen by applications that use the objects. It should not be possible for an application to determine the number of replicas, or to be able to see the identities of specific replica instances.

20. Define access transparency?

Access Transparency **allows the same operations to be used to access local and remote resources**.

PART -B

1. Explain how a parallel system differs from a distributed system (May 2022,Mark-13)
2. Illustrate the difference between message passing and shared memory processcommunication model (Dec 2022, Mark-13)
3. Discuss the design issuses and challenges in distributed system from a systemperspective. (May 2022, Mark-13)

UNIT -2

TWO MARKS

1. What are the issues in distributed system?

There is no global time in a distributed system, so the clocks on different computers do not necessarily give the same time as one another.

All communication between processes is achieved by means of messages. Message communication over a computer network can be affected by delays, can suffer from a variety of failures and is vulnerable to security attacks.

2. What is meant by group communication in distributed system?

Group Communication occurs when a single source process simultaneously attempts to communicate with numerous functions. A group is an abstract collection of interrelated operations. This abstraction hides the message passing such that the communication seems to be a standard procedure call.

3. What is meant by asynchronous programming?

Asynchronous programming provides opportunities for a program to continue running other code while waiting for a long -running task to complete.

4. Write application of casual order?

The causal ordering of messages describes the causal relationship between a message send event and a message receive event. For example, if send(M1) -> send(M2) then every recipient of both the messages M1 and M2 must receive the message M1 before receiving the message M2.

5. What is synchronous order?

Synchronous execution means the first task in a program must finish processing before moving on to executing the next task.

6. Define Scalar Time?

scalar time are independent (i.e., they are notcausally related), they can be ordered using any. arbitrary criterion without violating the causality. relation . Therefore, a total order is consistent with the. causality relation .

7. What is clock shew?

Clock skew (sometimes called timing skew) is a phenomenon in synchronous digital circuit systems (such as computer systems) in which the same sourced clock signal arrives at different components at different times due to gate or, in more advanced semiconductor technology, wire signal propagation delay.

8. What is clock drift rate?

Clock Drift: As mentioned, no two clocks would have the same clock rate of oscillations i.e; clock rate would be different. The difference of clock rate is called clock drift.

9. What is clock tick?

Clock Tick: after a predefined number of oscillations, the timer will generate a clock tick. This clock tick generates a hardware interrupt that causes the computer's operating system to enter a special routine in which it can update the software clock and run the process scheduler.

10. What is logical Clock?

Logical Clocks refer to implementing a protocol on all machines within your distributed system, so that the machines are able to maintain consistent ordering of events within some virtual timespan. A logical clock is a mechanism for capturing chronological and causal relationships in a distributed system.

11. What is global state of the distributed system?

The global state of a distributed system is the set of local states of each individual processes involved in the system plus the state of the communication channels. Determinism. Deterministic Computation.

12. Write the happen before relation?

- Happened before relation is an irreflexive partial ordering on the set of all events happening in the system i.e.; (a⇢ a) is not true for any event a.
- This relates back to Einstein's general theory of relativity where events are ordered in terms of messages that could possibly be sent.

13. What is vector clock?

Vector Clock is an algorithm that generates partial ordering of events and detects causality violations in a distributed system.

14. What is chandy lamport algorithm?

Chandy and **Lamport** were the first to propose a algorithm to capture consistent global state of a distributed system. The main idea behind proposed algorithm is that if we know that all message that have been sent by one process have been received by another then we can record the global state of the system.

Part -B

1. Explain about Message ordering paradigms (May 2022, Dec 2022 , Mark-13)
2. Explain the types of Group communication used in distributed system (Dec 2022 , Mark-13)
3. Elucidate on the total and casual order in distributed system with a neat diagram (Dec 2022, Mark- 13)
4. Explain about Chandy Lamport Snapshot algorithms for FIFO channels (May 2022, Mark-13)

UNIT -3

TWO MARKS

1. What is clock synchronization?

Nodes in distributed system to keep track of current time for various purposes such as calculating the time spent by a process in CPU utilization ,disk I/O etc so that the corresponding user can be charged. Clock synchronization means the time difference between two nodes should be very small.

2. Explain the term mutual exclusion

A program object that blocks multiple users from accessing the same shared variable or data at the same time. With a critical section, a region of code in which multiple processes or threads access the same shared resource, this idea is put to use in concurrent programming.

3. What is deadlock?

A Deadlock is a situation where a set of processes are blocked because each process is holding a resource and waiting for another resource occupied by some other process. When this situation arises, it is known as Deadlock. Deadlock.

4. Name the two types of messages used in Ricart-Agrawala's algorithm

The algorithm uses two types of messages: REQUEST and REPLY. A process sends a REQUEST message to all other processes to request their permission to enter the critical section. A process sends a REPLY message to a process to give its permission to that process.

5. What are the conditions for deadlock?

The four necessary conditions for a deadlock situation are mutual exclusion, no preemption, hold and wait, and circular set. There are four methods of handling deadlocks - deadlock avoidance, deadlock prevention, deadline detection and recovery, and deadlock ignorance.

6. Which are the three basic approaches for implementing distributed mutual exclusion?

- Token-based approach.
- Non-token-based approach.
- Quorum-based approach.

7. What are the requirements of mutual exclusion algorithms?

- No Deadlock: Two or more site should not endlessly wait for any message that will never arrive.
- No Starvation: Every site who wants to execute critical section should get an opportunity to execute it in finite time.

8. What is response time?

Response time includes the time taken to transmit the inquiry, process it by the computer, and transmit the response back to the terminal.

9. What is wait for graph?

A wait-for graph in computer science is a directed graph used for deadlock detection in operating systems and relational database systems.

10. What do you mean by deadlock avoidance?

Deadlock avoidance is another technique used in operating systems to deal with deadlocks. Unlike deadlock prevention, which aims to eliminate the possibility of deadlocks, deadlock avoidance focuses on dynamically detecting and avoiding situations that could lead to deadlocks.

11. Define deadlock detection in distributed system?

Deadlock detection involves two basic tasks: maintenance of the state graph and search of the state graph for the presence of cycles. Because in distributed systems a cycle may involve several sites, the search for cycles greatly depends on how the system state graph is represented across the system.

Part -B

1. Explain about Ricart Agrawala's Algorithm with an example
2. Analyse suzuki kasami's broadcast algorithm for mutual exclusion in distributedsystem
3. Discuss with suitable example to show that a deadlock cannot occur if any one ofthe four conditions is absent

Name and explain the different types of deadlock models in distributed systemwith the commonly used strategies to handle deadlocks with a neat diagram.

UNIT - 4

TWO MARKS

1. What do you mean by clock skew and clock drift?

- **Clock skew** – Instantaneous difference between the readings of any two clocks is called clock skew. Skew occurs since computer clocks like any others tends not be perfect at all times.

Clock drift – Clock drift occurs in crystal based clocks which counts time at different rates and hence they diverge. The drift rate is the change in the offset between the clock and a nominal perfect reference clock per unit of time measured by the reference clock.

2. What do you mean by Coordinated Universal Time?

Coordinated Universal Time generally abbreviated as UTC is an international standard for timekeeping. It is based on atomic time. UTC signals are synchronized and broadcast regularly from land based radio stations and satellites covering many parts of the world.

3. Define External Synchronization.

Generally it is necessary to synchronize the processes' clocks C_i with an authoritative external source of time. It is called as External Synchronization. For a synchronization bound D>0, and for a source S of UTC time, $| S(t) – C_i(t)| < D$ for i=1,2..N for all real times t in I where I is the time interval.

4. When an object is considered to be garbage?

An object is considered to be garbage if there are no longer any references to it anywhere in the distributed system. The memory taken up by the object

can be reclaimedonce it is known to be garbage. The technique used here is distributed garbage collection.

5. What do you meant by Distributed debugging?

In general, distributed systems are complex to debug.A special care needs to be taken in establishing what occurred during the execution. Consider an application with a variable x_i(i=1,2..N) and the variable changes as the program executes but it is always required to be within a value $ of one other. In that case, relationship must be evaluated for values of the variables that occur at the same time.

6. Define marker receiving rule.

Snapshot algorithm designed by Chandy and Lamport is used for determining global states of distributed systems. This algorithm is defined through two rules namely marker sending rule and marker receiving rule. Marker receiving rule obligates a process that has not recorded its state to do so.

7. Define marker sending rule.

Snapshot algorithm designed by Chandy and Lamport is used for determining global states of distributed systems. This algorithm is defined through 2 rules namely marker sending rule and marker receiving rule. Marker sending rule obligates processes to send a marker after they have recorded their state ,but before they send any other messages.

8. Define the characteristics of serial equivalent transactions.

For any pair of transactions, it is possible to determine the order of pairs of conflicting operations on objects accessed by both of them. Read and write are the operations generally considered. For two transactions to be serially equivalent it is necessary and sufficient that all pairs of conflicting operations of the two transactions be executed in the same order at all of the objects they both access.

9. What are the advantages of nested transactions?

The outermost transaction in a set of nested transactions is called top level transaction. Transactions other than the top level transaction are called subtransactions.

Advantages of nested transactions are:

- Subtransactions at one level may run concurrently with other subtransactions at the same level in the hierarchy. This can allow additional concurrency in a transaction.
- Subtransactions can commit or abort independently.

10. What are the rules of committing nested transactions?

Rules for committing of nested transactions are:

- A transaction may commit or abort only after its child transactions have completed.
- When a subtransaction completes, it makes an independent decision

  either to commitprovisionally or to abort.
- When a parent aborts, all of its transactions are aborted.

When a subtransaction aborts, the parent can decide whether to abort or not

11. Write short notes on strict two phase locking

A simple mechanism of a serializing mechanism is the use of exclusive locks. Under a strict execution regime, a transaction that needs to read or write an object must be delayed until other transactions that wrote the same object have committed or aborted. To enforce this rule, any locks applied during the progress of a transaction are held until the transaction commits or aborts. This is called *strict two-phase locking*. The presence of the locks prevents other transactions reading or writing the objects.

12. Define the approach of two phase commit protocol.

Two phase commit protocol is designed to allow any participant to abort its part of a transaction. In the first phase of the protocol, each participant votes for the transaction to be committed or aborted. In the second phase of the protocol, every participant in the transaction carries out the joint decision

PART - B

1. List the agreement statement that should be followed in synchronous system with failure?
2. Illustrate briefly the two kinds of checkpoints for checkpoint algorithm?
3. Discuss the issues of failure recovery with an example?
4. Illustrate the different types of failure in distributed systems?

UNIT V

PART A

1. Define Cloud Computing.
2. List out the essential characteristics of cloud computing.
3. Define On-demand self service
4. Define Resource pooling
5. What is Rapid elasticity.
6. What is multi-tenancy.
7. List and define the services provided by the cloud models.
8. Define IaaS.
9. Define PaaS.
10. Define SaaS.
11. List and explain the cloud deployment models.
12. List out the Benefits, characteristics, adoption and examples of IaaS.
13. List out the Benefits, characteristics, adoption and examples of PaaS.
14. List out the Benefits, characteristics, adoption and examples of SaaS.
15. Explain Google App Engine.
16. Explain Sales force.
17. Explain Amazon EC2.
18. What is S3
19. List out the cloud based services and applications.
20. Define Virtualization.
21. What is Hypervisor.
22. Explain Full Virtualization, Para-Virtualization and Hardware Virtualization.
23. What is Guest OS.
24. What is scalability.
25. What is Load balancing.
26. What is the goal of load balancing techniques.
27. List out the popular hypervisors.
28. List out the algorithms used for load balancing .
29. What is Round Robin.
30. Weighted Round robin.
31. Explain about low latency in load balancing.
32. What is overflow in load balancing.
33. List and explain the persistence approaches used in load balancing.
34. What is sticky sessions.

35. Explain Session Database.
36. Explain cookies.
37. Explain re-writing.
38. Explain Scalability & Elasticity.
39. What is Capacity planning.
40. List out the examples of popular load balancers.
41. What is scaling in and scaling out.
42. What is over-provisioning and under provisioning.
43. Define Replication and its types.
44. Define Array-based Replication and its merits and demerits.
45. Define Network-based Replication and its benefits.
46. Define Host-based Replication.
47. What is Block-based replication and File-based replication.
48. Define Monitoring.
49. Define Compute services and list out its features.
50. Define Storage services and its features.
51. List out the providers of compute services.
52. List out the providers of storage services
53. Define Application services and list out its providers.
54. List out the providers of database services.
55. Explain Queuing services and its providers.
56. Explain Email services and its providers.
57. Explain Notification services and its providers.
58. Explain Media services and its providers.

PART B & C

1. Explain in detail about service models with neat diagram.
2. Discuss about deployment models with neat diagram.
3. Briefly explain about cloud services-IaaS,PaaS, SaaS.
4. Explain in detail about virtualization with neat diagrams.
5. Briefly explain about Load balancing.
6. Detaily explain about scalability and elasticity.
7. Explain in detail about Replication.
8. Briefly explain about Monitoring.
9. Explain in detail about compute services.
10. Explain in detail about storage services.
11. Explain in detail about Application services.