CS3551 - DISTRIBUTED SYSTEMS

2 MARKS AND 16 MARKS

QUESTIONS WITH ANSWERS

UNIT 1

2 MARKS:

1.What do you mean by message passing ?

In this model, data is shared by sending and receiving messages between co-operating processes, using system calls. Message passing refers to services performing a simple, one-way transfer operation between two programs.

2. Define distributed program.

Distributed program is composed of a set of "n" asynchronous processes like P1 P2 P3... Pi Pn that communicate by message passing over the communication network.

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

Synchronous execution means the first task in a program must finish processing before moving on to executing the next task. Asynchronous execution means a second task can begin executing in parallel, without waiting for an earlier task to finish.

4. List out the features of distributed systems.

Features of distributed systems are heterogeneity, openness, scalability, fault tolerance, transparency and resource sharing.

5. Write down the principles of distributed systems.

Distributed system consists of a collection of autonomous computers, connected through a network and distribution middleware, which enables computers to coordinate their activities and to share the resources of the system, so that users perceive the system as a single, integrated computing facility.

6. State the objectives of resource sharing model.

Ability to use any hardware, software or data anywhere in the system. Resource manager controls access, provides naming scheme and controls concurrency.

7. What are the significant consequences of distributed systems?

a. No global clock: The only communication is by sending messages through a network.

b. Independent failures: The programs may not be able to detect whether the network has failed or has become unusually slow.

c. Concurrency: The capacity of the system to handle shared resources can increased by adding more resources to the network.

8. Define transparency. What are its types?

A distributed system needs to hide the fact that its processes and resources are physically distributed across multiple computers.

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

Distributed system must be able to interact with services from other open systems, irrespective of the underlying environment. Systems should conform to well-defined interfaces and should support portability of applications.

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

Hardware resource: Memory cache server and CPU servers do some computation for their clients hence their CPU is a shared resource.

Software resource: File: File servers enable multiple clients to have read/write access to the same files. Database: The content of a database can be usefully shared. There are many techniques that control the concurrent access to a database.

11. List an example of distributed system.

Distributed system examples: Internet, an intranet which is a portion of the internet managed by an organization, mobile and ubiquitous computing.

12. Enlist the design issues and challenges of distributed systems.

Design issues and challenges of distributed systems are heterogeneity openness, security, scalability, failure handling, concurrency and transparency.

13. Define access transparency.

Enables local and remote information objects to be accessed using identical operations.

14. What is replication transparency ?

It enables multiple instances of information objects to be used to increase reliability and performance without knowledge of the replicas by users or application programs. Example: Distributed DBMS.

15. What is the goal of concurrency and failure transparency?

Enables several processes to operate concurrently using shared information objects without interference between them.

Failure transparency: Allows users and applications to complete their tasks despite the failure of other components.

16. Differentiate between buffering and caching.

Cache is made from static ram which is faster than the slower dynamic ram used for a buffer. A cache transparently stores data so that future requests for that data can be served faster. A buffer temporarily stores data while the data is the process of moving from one place to another, ie. the input device to the output device. The buffer is mostly used for input/output processes while the cache is used during reading and writing processes from the disk.

17. What is open distributed system?

Open distributed system is a system that offers services according to standard rules that describe the syntax and semantics of those services.

18. Give the example of relocation transparency?

When mobile users can continue to use their wireless laptops while moving from place to place without ever being disconnected.

19. Describe what is meant by a scalable system?

A system is scalable with respect to either its number of components, size or number and size of administrative domains, if it can grow in one or more of these dimensions without an unacceptable loss of performance.

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

To enhance the distribution transparency that is missing in network operating systems. In other words, middleware aims at improving the single system view that a distributed system should have.

13 MARKS:

1. Explain the difference between message passing and message sharing.

- Message passing
- Message sharing
- Emulating Message – passing systems on a shared memory systems.

2. Describe about Design issues and challenges in Distributed Computing.

- Challenges from system perspective.
- Challenges.
    - Heterogeneity
    - Openness
    - Security

        - Scalability
        - Failure handling
        - Concurrency
        - Transparency

3. Explain about the model of Distributed Computations: A distributed program

- A model of Distributed Execution
    - Casual precedence relation
    - Logical vs Physical concurrency
- Models of Communication Networks

4. What is Global State? Explain about the global state of Distributed Systems.

- Definition
- Requirements of global state
    - Garbage collection
    - Deadlock
    - Termination
    - Distributed debugging

5. Explain the applications of Distributed Computing and Challenges.

- Applications
    - Mobile systems
    - Pervasive computing
        - Intranet
    - Multimedia system
        - Web casting

UNIT 2

2 MARKS:

1.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.

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

Group communication offers a service whereby a message is sent to a group and then this message is delivered to all members of the group. The sender is not aware of the identities of the receivers.

3. Write application of casual order.

Causal ordering is used for implementing distributed shared memory, fair resource allocation. Other applications are updating replicated data, synchronizing multimedia streams and allocating requests in a fair manner.

4. What is synchronous order?

When all the communication between pairs of processes is by using synchronous send and receives primitives, the resulting order is synchronous order.

5. Define scalar time.

Scalar time is designed by Lamport to synchronize all the events in distributed systems. Time domain is the set of non-negative integers.

6. List the issue related with implementation of logical clocks.

Addressing following issues: Data structures local to every process to represent logical time.

Protocol to update the data structures to ensure the consistency condition.

7. List the properties of scalar time.

Properties of scalar time are consistency, total ordering, event counting and system of scalar clocks is not strongly consistent.

8. What is Rendezvous?

Rendezvous is an architecture for creating multi-user applications. It provides support for managing a multi-user session, for performing fundamental input and output activities and for controlling the degree to which the multiple users either share or do not share both information and control.

9. What is clock tick?

When the counter gets to zero, an interruption is generated and is called one clock tick.

10. What is clock skew?

With n computers, all n crystals will run at slightly different rates, causing the software clocks to gradually get out of sync. This difference in time values is called dock skew.

11. What is clock drift rate.

A clock 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.

12. List the name of modes the NTP servers synchronize.

NTP servers synchronize with one another in one of three modes multicast,procedure-call and symmetric mode.

13. What are the two modes of synchronization?

The two modes are:

1. External synchronization: For a synchronization bound D > 0, and for a source S of UTC time, IS(0)-C, (0) T. for i = 1,2 N and for all real times t in L

2 Internal synchronization: For a synchronization bound D>0.(C0)-C < D. for 1.1 1.2 N and for all real times t in 1

14. What is logical clock?

Logical clock is a monotonically increasing software counter, whose value need bear no particular relationship to any physical clock. Each process P, keeps its own logical clock L., which it uses to apply so called Lamport timestamps to events.

15. What is global state of the distributed system?

The global state of the distributed system consists of the local state of each process, together with the messages which are in transit.

16. Write the happens-before relation?

The happens before relation can be observed directly in two situations:

1. If a and b are events in the same process, and a occurs before b then a 'b is true.

2. It is the event of a message being sent by one process, and b is the event of the message being received by another process, then a b is also true.

17. What is need of physical clock?

Lamport's algorithm for logical clock synchronization gives an unambiguous event ordering the tone values assigned to events are not necessarily close to the actual times at which they occur.

In some systems like real-time systems, the actual clock time is important these systems external physical clocks are required.

18. What problem with Lamport's clocks to vector clocks solve?

With Lamport's clocks, you cannot tell whether two events are causally related or concurrent by looking at the timestamps. Just because L(a) L(b) does not mean that a->b Vector clocks allow you to compare two vector timestamps to determine whether the events are concurrent or not.

19. What is vector clock?

Vector clocks are used in a distributed system to determine whether pairs of events are causally related Using vector clocks, timestamps are generated for each event in the system, and their causal relationship is determined by comparing those timestamps.

20. How vector clock timestamps are assigned?

Vector clock timestamps are assigned as follows:

a Events: Every time an event is generated, a process increments its clock and assigns a timestamp to the event based on its knowledge of all the clocks in the system.

b. Sending messages: When a message is sent the timestamp of the sending event es given to the message.

c. Receiving messages: When a message is receivest, the process updates its knowledge of the system clock states by taking the maximum of each component of the message timestamp and its current knowledge of the system clock states.

13 MARKS:

1. Explain Logical Time.

- Event Ordering
    - Condition of happens before
    - Logical clock condition
- Lamport Timestamp
- Vector Timestamp

2. Discuss about Physical Clock Synchronization: NTP

- Synchronization in a Synchronous System
- Cristian's Method for Synchronizing Clocks
    - Christian's Algorithm
- Berkeley Algorithm
- Network Time Protocol
    - Localized Averaging Distributed Algorithm.
        - Network time protocol
        - Features of NTP

3. Explain the difference between Scalar Time and Vector Time.

- Scalar Time
    - Basic Properties
        - Consistency property
        - Total ordering
        - Event counting
- Vector Time
    - Definition

4. Describe and Explain the Group Communication.

- One to Many Communication
    - Group Management
    - Group addressing
    - Buffered and unbuffered multicast
- Many to One Communication
- Many to Many Communication
    - Message Ordering
        - Absolute ordering
        - Consistent/ Total Ordering
        - Causal ordering

5. Explain the global state and snapshot recording algorithm.

- Definition
- System Model
- Consistent Global State

UNIT 3

2 MARKS:

1.Explain the term mutual exclusion.

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

2. What is deadlock?

Deadlock is the problem of multiprogramming system. Deadlock can be defined as the permanent blocking of a set of processes that either complete for system resources.

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

Two type of messages used by Ricart-Agrawala are REQUEST and REPLY and communication channels are assumed to follow FIFO order. Site send a REQUEST message to all other site to get their permission to enter critical section. A site send a REPLY message to other site to give its permission to enter the critical section.

4. What are the conditions for deadlock?

Conditions should hold simultaneously for deadlock to occur are:

a) Mutual exclusion c) Hold and wait

b) No preemption d) Circular wait.

5. What is mutual exclusion?

Mutual exclusion in a distributed system states that only one process is allowed to execute the critical section (CS) at any given time. In a distributed system, shared variables or a local kernel cannot be used to implement mutual exclusion.

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

There are three basic approaches for implementing distributed mutual exclusion

1. Token based approach 2. Non-token based approach

3. Quorum based approach

7. What are the requirements of mutual exclusion algorithms?

Requirements of mutual exclusion algorithms are

a. Freedom from deadlocks

b. Freedom from starvation

c Strict fairness

d. Fault tolerance

8. What are the performance metric of mutual exclusion algorithm?

Performances metric are message complexity, synchronization delay, response time and system throughput.

9. What is response time?

The time interval a request waits for its CS execution to be over after its request messages have been sent out.

10. Which are the criteria for evaluating performance of algorithms for mutual exclusion?

Criteria for evaluating performance of algorithms for mutual exclusion are:

a. Bandwidth consumed which is proportional to the number of messages sent in each entry and exit operation.

b. Client delay incurred by a process at each entry and exit operation.

c. Throughput of the system.

10. What is the advantage if your server side processing uses threads instead of a single process?

An important property of threads is that they can provide a convenient means of allowing blocking system calls without blocking the entire process in which the thread is running. This property makes threads particularly attractive to use in distributed systems as it makes it much easier to express communication in the form of maintaining multiple logical connections at the same time.

11. What is a phantom deadlock?

A deadlock that is detected' but is not really a deadlock is called a phantom deadlock.

12. What is wait for graph?

The state of process-resource interaction in distributed systems can be modeled by a bi-partite directed graph called a resource allocation graph. The nodes of this graph are processes and resources of a system, and the edges of the graph depict assignments or pending requests. A pending request is represented by a request edge directed from the node of a requesting process to the node of the requested resource.

13. Explain Wait-die" method.

A non-preemptive approach. If a younger process is using the resource, then the older process waits. If an older process is holding the resource, the younger process kills itself. This forces the resource utilization graph to be directed from older to younger processes, making cycles impossible. This algorithm is known as the wait-die algorithm.

14. List the deadlock handling strategies in distributed system.

There are three strategies for handling deadlocks, viz, deadlock prevention, deadlock avoidance, and deadlock detection.

15. What do you mean by deadlock avoidance?

Deadlock avoidance depends on additional information about the long term resource needs of each process. The system must be able to decide whether granting a resource is safe or not and only make the allocation when it is safe. When a process is created, it must declare its maximum claim, i.e. the maximum number of unit resource The resource manager can grant the request if the resources are available.

16. Define deadlock detection in distributed systems.

Deadlock detection requires examination of process-resource interaction for the presence of cyclic wait.

17. What is Chandy-Misra-Haas Algorithm?

A blocked process determines if it is deadlocked by initiating a diffusion computation. processes in its dependent set. If an active process receives a query or reply message, it discards it. all the query messages it has sent out.

18. What is OR Model?

Set of Deadlocked processes, where each process waits to receive messages from other processes in the set.

19. What is AND Model?

Set of deadlocked processes, where each process waits for resource held by another process.

Use AND condition.

The condition for deadlock in a system using the AND condition is the existence of a cycle.

20. Define Deadlock Avoidance.

Decision made dynamically, before allocating a resource, the resulting global system state is checked, if it is safe state then allow for allocation.

Because of the following drawback, deadlock avoidance can be impractical in distributed system.

13 MARKS:

1. What is Lampot’s Algorithm and Explain it?

- Definition
- Requesting the critical session
- Conditions for entering CS
- Releasing the CS
- Correctness
- Optimization
- Lamport evalution

2. Explain Ricart – Agarwala’s Algorithm.

- Definition
- Algorithm
- Requesting the Critical Session
- Executing the Critical Session
- Releasing the Critical Session

3. Explain Token Based Algorithm

- Definition
- Suzuki – Kasami’s Broadcast Algorithm.
    - Major Design Issues
    - Important Data Structures
    - Algorithm
        - Requesting CS
        - Executing CS
        - Releasing CS
        - Theorem: A requesting site enters CS in finite time
        - Performance

4. Discuss the Deadlock Detection in Distributed Systems: Introduction.

- Deadlock
- Necessary Condition
    - Mutual exclusion
    - Hold and wait
    - Circular waiting
    - No preemption

5. Explain Preliminaries: Deadlock Handling Strategies.

- Deadlock Prevention
    - First Method
    - Second Method

    - Third Method
- Dead Avoidance
    - Disadvantage
- Deadlock Detection
    - Principle of operation
    - Resolution
    - Observation

UNIT 4

2 MARKS:

1. State the use of Rollback recovery.

Restore the system back to a consistent state after a failure

. • Achieve fault tolerance by periodically saving the state of a process during the failure-free execution.

- Treats a distributed system application as communicate over a network.

2. What is consensus in distributed system?

Each process has an initial value and all the correct processes must agree on a single value.

3. Write the purpose of using checkpoints.

Check pointing is most typically used to provide fault tolerance to applications. Check pointing techniques are useful not only for availability, but also for program debugging, process migration, and load balancing.

4. What do you mean by agreement problem in distributed system?

In the agreement problem, to achieve overall system reliability in the presence of a number of faulty processes and single process has the initial value.

5. What is the difference between agreement and consensus problem?

The difference between the agreement problem and the consensus problem is that, in the agreement problem, a single process has the initial value, whereas in the consensus problem, all processes have an initial value.

6. Define recovery.

Recovery refers to restoring a system to its normal operational state. Once a failure has occurred, it is essential that the process where the failure happened recover to a correct state. Fundamental to fault tolerance is the recovery from an error.

7. Explain two types of checkpoints.

1. Tentative: A temporary checkpoint that is made a permanent checkpoint on the successful termination of the checkpoint algorithm.

2. Permanent: A local checkpoint at a process.

8. List drawback of synchronous check pointing.

1. Additional messages must be exchanged to coordinate check pointing

2. Synchronization delays are introduced during normal operations

3. No computational messages can be sent while the check pointing algorithm is in progress.

4. If failure rarely occurs between successive checkpoints, then the checkpoint algorithm places an unnecessary extra load on the system, which can significantly affect performance.

9. How shadow versions are helpful in recovery?

Shadow version uses a map to locate versions of the server's objects in a file called a version store. The map associates the identifiers of the server's objects with the positions of their current versions in the version store. The versions written by each transaction are shadows of the previous committed versions. The transaction status entries and intentions lists are stored separately. When a transaction commits, a new map is made by copying the old map and entering the positions of the shadow versions. To complete the commit process, the new map replaces the old map.

10. Define fault and failure. What are different approaches to fault-tolerance?

Fault: Anomalous physical condition, eg design errors, manufacturing problems, damage, external disturbances.

Failure of a system occurs when the system does not perform its service in the manner specified.

11. List the requirements of consensus algorithm to hold for execution.

The requirements of consensus algorithm to hold for execution are

1. Termination
2. Agreement and
3. Integrity.

12. What are the performance aspects of agreement protocols?

Following metrics are used

1. Time: No of rounds needed to reach an agreement.

2. Message traffic: Number of messages exchanged to reach an agreement.

3. Storage overhead: Amount of information that needs to stored at processors

during execution of the protocol.

13. What are the application of agreement algorithm?

Applications of agreement algorithms

- Fault-tolerant clock synchronization.
- Distributed systems require physical clocks to synchronized
- Physical clocks have drift problem.
- Agreement protocols may help to reach a common clock value. • Synchronizing distributed clocks:
- At any time, values of clocks of all non-faulty processes must be approximately equal.
- There is a small bound on amount by which the clock of a non-faulty process is

changed during re-synchronization.

14. State Byzantine agreement problem.

In the Byzantine agreement problem, n processors communicate with each other in order to reach an agreement on a binary value b. There are bad processors. that may collaborate with each other in order to prevent an admissible agreement. Each processor has an initial binary value. The agreement must reflect to a certain extent the majority among the initial value.

15. What is local checkpoints?

A process may take a local check point anytime during the execution. The local checkpoints of different processes are not coordinated to form a global consistent checkpoint.

16. What is forced checkpoints?

To guard against the domino effect, a communication induced checkpoint protocol piggybacks protocol-speci?c information to application messages that processes exchange. Each process examines the information and occasionally is forced to take a checkpoint according to the protocol.

17. Explain useless checkpoints.

A useless checkpoint of a process is one that will never be part of a global consistent state. Useless checkpoints are not desirable because they do not contribute to the recovery of the system from failures, but they consume resources and cause performance overhead.

18. What is checkpoint intervals?

A checkpoint interval is the sequence of events between two consecutive checkpoints in the execution of a process.

19. Define orphan messages.

Messages with receive recorded but message send not recorded are called the orphan messages.

20. What is the basic idea behind task assignment approach?

Basic idea:

a. A process has already been split up into pieces called tasks:

b. The amount of computation required by each task and the are known.

c. The cost of processing each task on every node is known.

d. The IPC costs between every pair of tasks is known..

e. Precedence relationships among the taks are known.

f. Reassignment of tasks is not possible. Mention some motivations for replication.

13 MARKS:

1. Explain the Solution to Byzantine Agreement Problem.

- Impossible Scenario
- Lamport – Shostak – Pease Algorithm
  - Example

2. Explain the Consistent Set of Checkpoint.

- Definition
  - Strongly Consistent Set of Checkpoint.
  - Consistent Set of Checkpoint
  - Checkpoint Notation
- Synchronous Checkpoint and Recovery
  - Checkpointing Algorithm
    - Types
  - Synchronous Checkpointing Disadvantages
- The Rollback Recovery Algorithm
  - Phase one
  - Phase two
- Message Types

3. Discuss about the Checkpoint – Based Recovery.

- Uncoordinated Checkpointing
  - Direct dependency tracking technique
- Coordinated Checkpointing
  - Blocking Checkpointing
  - Non – Blocking Checkpointing
- Communication – induced Checkpointing

4. Describe the Issues in Failure Recovery.

- Basic Concept
- Recovery
    - System Failure
    - Erroneous System State
    - Error
    - Fault

5. Explain Byzantine Agreement Problem.

- Introduction
    - The Problem
    - Validity
- Consensus Problem
    - Agreement
    - Validity
- Interactive Consistency Problem

UNIT 5

2 MARKS:

1. Explain NIST definition of cloud computing.

NIST definition of cloud: Cloud computing is a pay-per-use model for enabling available, convenient, on-demand network access to a shared pool of configurable computing resources (eg, networks, servers, storage, applications, services) that can be rapidly provisioned and released with minimal management effort or service-provider interaction.

2. What is cloud service?

Cloud service is any service made available to users on demand via the Internet from a cloud computing provider's servers as opposed to being provided from a company's own on-premises servers.

3. What is public cloud?

Public cloud is built over the Internet and can be accessed by any by user who has paid for the service. Public clouds are owned by service providers and are accessible through a subscription.

4. What is private clouds?

A private cloud is built within the domain of an intranet owned by a single organization. Therefore, it is client owned and managed, and its access is limited to the owning clients and their partners.

5. Explain about virtual machines.

A Virtual Machine (VM) is a software construct that mimics the characteristics. of a physical server. VM is a software program or operating system that not only exhibits the behavior of a separate computer, but is also capable of performing tasks such as running applications and programs like a separate computer.

6. What is NIST definition of laaS?

The ability given to the infrastructure architects to deploy or run any software on the computing resources provided by the service provider. The end users are responsible for managing applications that are running on top of the service provider cloud infrastructure.

7. Explain characteristics of laas.

Characteristics of laaS

1. Resources are provided as a service

2. Allows for dynamic scaling and elasticity.

3. It has a variable cost, usage based pricing model (pay per go and pay per use)

4. It has multi-tenet architecture, includes multiple users on a single piece of hardware.

5. IaaS typically has enterprise grade infrastructure.

8. List the situations where PaaS may not be the best option.

- Integration with on-premise applications.
- Flexibility at the platform level.
- Customization at the infrastructure level.
- Frequent application migration.

9. What is Amazon EC2?

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers and system administrators.

10. List the function of EC2?

EC2 functions:

1. Load variety of operating system.

2. Install custom applications

3. Manage network access permission 4. Run image using as many/few systems as we desire.

11. What is Azure?

Windows Azure is a cloud computing platform and infrastructure, created by Microsoft, for building, deploying and managing applications and services through a global network of Microsoft-managed data centers.

12. What is Azure queues?

Azure queue storage is a service for storing large numbers of messages that can be accessed from anywhere in the world via authenticated calls using HTTP or HTTPS A single queue message can be up to 64 KB in size, and a queue can contain of messages, up to the total capacity limit of a storage account.

13. How virtualization employed in Azure?

Azure is a virtualized infrastructure to which a set of additional enterprise services has been layered on top, including a virtualization service called An AppFabric that creates an application hosting environment. AppFabric is a cloud-enabled version of the NET framework.

14. What is service cloud?

Service cloud refers to the service module in Salesforce.com. It includes accounts, contacts, cases, and solutions. It also encompasses features such as the publi knowledge base, web-to-case, call center, and self-service portal, as well as custome service automation.

15. What is Google Cloud Storage?

Google cloud storage allows world-wide storage and retrieval of any amount of data at any time. It can be used for a range of scenarios including serving website content, storing data for archival and disaster recovery, or distributing large data objects to users via direct download.

16. Define Storage Service.

Amazon S3 defines a bucket name as a series of one or more labels, separated by periods, that adhere to the following rules: The bucket name can be between 3 and 63 characters long, and can contain only lower-case characters, numbers, periods, and dashes.

17. What Scalability and Elasticity?

Scalability is the ability of a system or network to handle increased load or usage. At the same time, elasticity is the ability to automatically expand and contract resources to meet demand.

Cloud elasticity is a system's ability to manage available resources according to the current workload requirements dynamically. This is a vital feature of a system infrastructure. It comes in handy when the system is expected to experience sudden spikes of user activity and, as a result, a drastic increase in workload demand.

18. Define Load Balancing.

Load balancing can be defined as the process of task distribution among multiple computers, processes, disk, or other resources in order to get optimal resource utilization and to reduce the computation time.

Load balancing is an important means to achieve effective resource sharing and utilization.

19. Define Pros and Cons of Virtualization.

Pros:

1. Data center and energy-efficiency savings: As companies reduce the size of their hardware and server footprint, they lower their energy consumption.

2. Operational expenditure savings: Once servers are virtualized, your IT staff can greatly reduce the ongoing administration and management of manual work.

3. Reduced costs: It reduced cost of IT infrastructure. 4. Data does not leak across virtual machine.

5. Virtual machine is completely isolated from host machine and other virtual machine.

6. Simplifies resource management by pooling and sharing resources.

7. Significantly reduce downtime.

8. Improved performance of IT resources.

Cons:

1. Not all hardware or software can be virtualized.

2. Not all servers are applications are specifically designed to be virtualization-friendly.

20. Define Para-Virtualization.

Paravirtualization is a type of virtualization in which a guest operating system (OS) is recompiled, installed inside a virtual machine (VM), and operated on top of a hypervisor program running on the host OS.

- Para-virtualization refers to communication between the guest OS and the hypervisor to improve performance and efficiency.

- Para-virtualization involves modifying the OS kernel to replace non-virtualizable instructions with hyper-calls that communicate directly with the virtualization layer hypervisor.

13 MARKS:

1. Explain Cloud Deployment Models

- Public Cloud
    - Benefits
    - Risks
- Private Cloud
    - Benefits
    - Risks
- Community Cloud
- Hybrid Cloud
    - Benefits
    - Risks
- Difference between public and private Cloud

2. Explain Cloud Service Models

- Software as a Service(SaaS)
    - Characteristics
    - Benefits
- Platform as a Service(PaaS)
    - Characteristics
    - Benefits
- Infrastructure as a Service(IaaS)
    - Types
        - Physical Server
        - Dedicated Virtual Server
        - Shared Virtual Server
    - Advantage

3. Discuss about Virtualization.

- Hypervisor
- Para – Virtualization
    - Problems
- Full – Virtualization

        - Host Based Virtualization
    - Pros and Cons of Virtualization

4. Explain Cloud Services and Platforms: Compute Services.

    - Amazon Elastic Compute Cloud
        - Launching an EC2 instance
        - Stop instances
    - Windows Azure

5. Explain the Application Services of Cloud.

    - Application Framework and Runtime: Google App Engine
        - Major feature of Google App Engine
        - Key feature of GAE Programming mode using Java and Python
            - Python
            - Java
        - Queuing Service: Amazon Simple Queue Service
            - Example.