VETRI VINAYAHA COLLEGE OF ENGINEERING & TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

QUESTION BANK

SUBJECT CODE: CS6601

SUBJECT NAME: DISTRIBUTED SYSTEMS

REGULATION: 2013

YEAR / SEM: III/VI

UNIT I
INTODUCTION
PART A

1. Define distributed system.

A distributed system is one in which components located at networked computers communicate and coordinate their actions only by passing messages. A distributed system is a collection of independent computers that appears to its users a single coherent system.

2. What is the significance of distributed system?

a. Concurrency of computers.
b. No global clock.
c. Independent failures.

3. Why we do you need distributed system?

a. Functional distribution: Computers have different functional capabilities (i.e., sharing of resources with specific functionalities).

b. Load distribution/balancing: Assign tasks to processors such that the overall system performance is optimized.

c. Replication of processing power: Independent processors working on the same task.

d. Distributed system consisting of collections of microcomputers may have processing powers that no supercomputer will ever achieve.

e. Physical separation: Systems that rely on the fact that computers are physically separated (e.g., to satisfy reliability requirements).

f. Economics: Collections of microprocessors offer a better price/performance ratio than large mainframes.mainframes:10 times faster, 1000 times as expensive.

4**. Examples of distributed system?**

a. Internet
b. Intranet
c. Mobile and ubiquitous computing.

5. What are the two type of resource sharing?

a. Hardware sharing: Printers. plotters and large disks and other peripherals are shared to reduce costs.

b. Data sharing is important in many applications:

Software developers working in a team need to access each other's code and share the same development tools.

6. List the importance of data sharing?

-Software developers working in a team need to access each other's code and share the same development tools.

-Many commercial applications enable users to access shared data objects in a single active database.

-The rapidly growing area of group- ware tools enables users to cooperate with in a network.

7. Write the technological components of web?

HTML

HTTP-request-reply protocol

URL's

8. List the distributed systems challenges?

a. Heterogeneity: standards and protocols; middleware; virtual machine;

b. Openness: publication of services; notification of interfaces;

c: Security: firewalls; encryption;

d. Scalability: replication; caching; multiple servers;

e. Failure Handling. failure tolerance; recover/roll-back; redundancy;

f. Concurrency. concurrency control to ensure data consistency.

g. Transparency. Middleware; location transparent naming; anonymity

9. What are the three components of security?

Security for information resources has three components:

Confidentiality: production against disclosure to unauthorized individuals.

Integrity: production against or corruption.

Availability: production against interference with the means to access the resources.

10. What is the use of firewall?

A firewall can be used to form a barrier around an intranet to protect it from outside users but does not deal with ensuring the appropriate use of resources by users within the intranet.

11. What are the security challenges? List them.

a. **Denial of service attacks:** Another security problem is that the user may wish to disrupt a service for some reason. This can be achieved by bombarding the service with such a large number of pointless requests that the serious users are unable to use it. This is called a denial of service attack and there are many on well known web services.

b. **Security of mobile code:** Mobile codes needed to be handled with care. PC users sometimes send executable files as email attachments to be run by the recipient, but a recipient will not be able to run it.

12. List the challenges to be considered for designing scalable distributed system?

- Controlling the cost of physical resources
- Controlling the performance loss
- Preventing software resources running out
- Avoiding performance bottlenecks.

13. What are the types of transparencies?

a. **Access transparency:** enables local and remote resources to be accessed using identical operations.

b. Location transparency: enables resources to be accessed without knowledge of their location.

c. Concurrency transparency: enables several processes to operate concurrently using shared resources without interference between them.

d. Replication transparency: enables multiple instances of resources to be used to increase reliability and performance without knowledge of the replicas by users or application programmers. `

e. Failure transparency: enables the concealment of faults, allowing users and application programs to complete their tasks despite the failure of hardware or software components.

f. Mobility transparency: allows the movement of resources and clients within a system without affecting the operation of users or programs.

g. Performance transparency: allows the system to be reconfigures to implement performance as loads vary.

h. Scaling transparency: allows the system and applications to expand in scale without change to the system structure or the application algorithms.

14. What are the failures detected in DS?

Masking failures: Some detected failures can be hidden or made less severe. Examples of hiding failures:

1. Messages can be retransmitted when they fail to arrive

2. File data can be written to a pair of disks that if one is corrupted, the other may still be correct.

Tolerating failures: Most of the services in the Internet do exhibit failures. It would not be practical for them to detect and hide all the failures occur in such network. Their clients are designed to tolerate failures, which generally involve the users in that.

Recovery from failures: involves the design of software so that the state permanent data can be rolled back after a server has crashed.

15. List the key design goals of DS?

a. Performance

b. Reliability

c. Scalability

d. Consistency

e. Security

16. List the technical design goals of DS?

a. Naming
b. Communication
c. Software structure
d. Workload allocation

17. 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 is 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 moving from one place to another, i.e. 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.

18. 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.

19. What is World Wide Web?

The World Wide Web (WWW) can be viewed as a huge distributed system with millions of clients and servers for accessing linked documents. Servers maintain collection of documents while clients provide users an easy to use interface for presenting and accessing those documents.

20. What is web browser?

A client interacts with web servers through a special application known as a browser. a browser is responsible for properly displaying a document.

PART B

1. Explain distributed system with an examples
2. List the various challenges in distributed system and explain them.
3. Elaborate on the recent trends in distributed system.
4. How resource sharing is done in distributed system.
5. Explain in detail about WWW.

UNIT II
COMMUNICATION IN DISTRIBUTED SYSTEM
PART A

1. **What is physical model in DS?**

   Physical model consider the types of computers and devices that constitute a system and their interconnectivity, without details of specific technologies.

2. **Define architectural model.**

   ` Architectural models describe a system in terms of the computational and communication tasks performed by its computational elements the computational elements being individual computers or aggregates of them supported by appropriate network interconnection.

3. **Define web services.**

   Web services are distributed web applications that provide discrete functionality and expose it in a well defined manner over standard internet protocols to other web application.

4. **What is latency?**

   It is a delay between the start of a message's transmission from one process and the beginning of its receipt by another.

5. **Define bandwidth.**

   The bandwidth of a computer network is the total amount of information that can be transmitted over it in a given time.

6. **How do you define marshalling?**

   Marshalling is the process of taking collection of data items and assembling them into a form suitable for transmission in a message.

7. **What do you mean unmarshalling?**

   Unmarshalling is the process of disassembling data items on arrival to produce an equivalent collection of data items.

8. **Difference between RMI and RPC.**

   In RPC, the parameters that are passed are ordinary data structures. Whereas in RMI, objects can be passed as parameters. RPC supports procedural programming where as RMI is object based.

9. **What is meant by inter process communication?**

   Interprocess communication is the set of tools provided by the OS to allow processes that do not share common memory segments to communicate with each other.

10. **Define datagram.**

    Sending process to transmit a single message to a receiving process. The independent packets containing message are called datagram.

11. **What is the use of UDP?**

    UDP datagram's are sometimes an attractive choice because they do not suffer from the overheads associated with guaranteed message delivery.

12. **What is meant by client server communication?**

    The client server model of computing is a distributed application structure that partitions tasks or workloads between the providers of a resource or service called servers and service requesters called clients.

13. What is multicast communication?

A service where data is delivered from a sender to multiple receivers is called multipoint communication or multicast.

14. What is socket?

Socket is the basic abstraction for network communication in the socket API. Socket defines an end point of communication for a process.

15. What is XML?

An extensible markup language refers to a textual encoding that represents both a text and details as to its structure or its appearance. XML was designed to describe structured documents and markup languages.

16. Write the characteristics of XML.

-XML is extensible in the sense that users can define their own tags.

-XML is self-describing.

-XML is textual so can be easily read by humans and computers.

17. What is publish-subscribe system?

Publishers publish events and subscribers subscribe to and receive the events they are interested in. The main characterization of pub/sub is in the way notifications flow from senders to receivers. Receivers are not directly targeted from publisher but indirectly addresse according to the content of notification.

18. List the type of EJB.

1. Session beans
2. Message driven bean
3. Entity bean

19. What is tuple space?

It is an associative, distributed shared memory. Processes can write tuples to the tuple space. Processes can then read a tuple from tuple space which also leaves a copy in the tuple space.

20. Define client stub and server stub.

- Client stub perform marshalling into request messages and unmarshalling from reply message.

- Server stub perform unmarshalling from request messages and marshalling into reply message.

PART B

1. Explain the types of system models in detail.
2. Illustrate TCP and UDP communication with suitable sample program.
3. Explain the marshalling operation in CORBA.
4. Explain in detail about RMI.
5. Explain in detail about RPC.
6. Write down the steps in JavaRMI with suitable example.
7. Explain in detail about publish subscibe system.
8. Explain EJB in detail.

UNIT III
PEER TO PEER SERVICES AND FILE SYSTEM
PART A

1. What is routing overlay?

A routing overlay is a distributed algorithm for a middleware layer responsible for routing requests from any client to a host that holds the object to which the request is addressed.

2. List out file system modules.

Directory module: Relates file names to file IDs
File module: Relates file IDs to particular files
Access control module: Checks permission for operation requested
File access module: Read or writes file data or attributes
Block module: Accesses and allocates disk blocks
Device module: Disk I/O and buffering

3. List out the UNIX file system Operations:

fieldes=open(name,mode)
fieldes=create(name,mode)
status=close(fieldes)
count=read(fieldes,buffer,n)
count=write(fieldes,buufer,n)
pos=Iseek(filedes,offset,whence)
status=unlink(nmae)
status=link(name1,nmae2)
status=stat(name,buffer)

4. What is meant by concurrency control?

Changes to a file by one client should not interfere with the operation of other clients simultaneously accessing or changing the same file. This is well-known issue of concurrency control .The need for concurrency control for access to shared data in many applications is widely accepted and techniques are known for its implementation, but they are costly .Most current file services follow modern UNIX standards in providing advisory or mandatory file or record-level locking.

5. What is file replication?

In a file service that supports replication, a file may be represented by several copies of its contents at different locations. This has two benefits-its enables multiple servers to share the load of providing a service to clients accessing the same set of files, enhancing the scalability of the service, and it enhances fault tolerance by enabling clients to locate another server that holds a copy of the file when one has failed. Few file services support replication fully, but most support the catching of files or portions of files locally, a limited form of replication.

6. What is meant by directory services?

The directory services provide a mapping between text names for files and their UFIDs. Client may obtain the UFIDs of a file by quoting its text name to the directory services. The directory services provide the function needed to generate directories, to add new file name to directories and to obtain UFIDs from directories. It is client of the flat file services; its directory is stored in files of the flat services. When a hierarchic file-naming scheme is adopted as in UNIX, directories hold references to other directories.

7. What are the timestamps in called caching?

i. Tc is the time when the cache entry was last validated.

ii. Tm is the when the block was last modified at the server.

iii. A cache entry is valid at time T if T-Tc is less than a freshness interval t, or if the value for Tm recorded t the client matches the value of Tm at the server (that is, the data has not been modified at the server since the cache entry was made).

8. When the name is resolved?

The name is resolved when it is translated into data about the named resource or object, often in order to invoke an action upon it. The association between a name and an object is called a binding. In general, names are bound to attributes of the named objects, rather than the implementation of the objects themselves. An attribute is the value of a property associated with an object.

9. What is meant by URI?

URI-Uniform Resource Identifiers came about from the need to identify resources on the web, and other internet resources such as electronic mailboxes. An important goal was to identify resources in a coherent way, so that they could all be processed by common software such as browser. URIs is 'uniform' in that their syntax incorporates that of indefinitely many individual types of resource identifier(i.e URI schemas),and there are procedures for managing the global namespace of schemas. The advantage of uniformity is that eases the process of introducing new types of identifier, as well as using existing types of identifier in new contexts without disrupting existing usage.

10. What is mean by URN?

Uniform Resource Names are URIs that are used as pure resource names rather than locators. For example, the URI:

Mid:0E4FC272-5C02-11D9-B115-000A95B55BC8@hpl.hp.com

Is a URN that identifies the email message containing it in its 'message-id' field. The URI distinguishes that message from any other email message. But it does not provide the message's address in any store, so a lookup operation is needed to find it.

11. What is global name services?

The Global name Service developed at the Digital Equipment corporation systems, Research Center,is a descentdant of Grapevine with ambitious goals, including:

- com- Commercial organization.
- edu- universities and other educational institutions.
- gov- US governmental agencies
- mil- US military organization
- net- major network support centers
- org- organizations not mentioned above

12. What is meant by navigation?

The process of locating naming data from than more than one name server in order to resolve a name is called navigation. The client name resolution software carries out navigation on behalf of the client. It communicates with name servers as necessary to resolve a name.

13. What is multicast navigation?

In multicast navigation, a client multicast the name to be resolved and required object type to the group of name servers. Only the server that holds the named attributes responds to the request. Unfortunately, however, if the name proves to be unbound, the request is greeted with silence.

14. What is iterative navigation?

One navigation model that DNS supports is known as iterative navigation. To resolve a name, a client present the name to the local name server, which attempts to resolve it. If the local name server has the name, it returns the result immediately. If it does not it will suggest another server that will be able to help. Resolution proceeds at the new server, with further navigation as necessary until the name is located or is discovered to be unbounded.

15. What is meant by zone?

The DNS naming data are divided into zones. A zone contains the following data:

 Attributes data for names in a domain, less any subdomains administrates by lower level authorities.

 The names and address of at least two name servers that provide authoritative data for the zone. These are versions of zone data that can be relied upon as being reasonably up to date.

 The names of name servers that hold authoritative data for delegated sub domains; and 'glue' data giving the IP address of these servers.

 Zone-management parameters, such as those governing the catching and replication of zone data.

16. Define name space.

A namespace is the collection of all valid names recognized by a particular service. The service will attempt to look up a valid name, even though that name may prove not to correspond to any object. Name space requires a syntactic definition to separate.

17. Define immutable file.

An immutable file is one that, once created, cannot be changed. Immutable files are easy to cache and to replicate across servers since their contents are guaranteed to remain unchanged.

18. What is LDAP?

It is a protocol that runs over TCP/IP. The LDAP protocol standard includes low-level network protocol definitions plus data representation and handling functionality. A directory that is accessible through LDAP is commonly referred to as an LDAP directory.

19. List the file accessing model.

1. Remote service model 2. Data caching model

20. List the data transfer levels.

1. File 2. Block
3. Byte 4. Record

PART B

1. Write short notes on
   1. Napster and its legacy 2.Peer to Peer middleware
2. With neat sketch explain routing overlays in detail.
3. Explain file service architecture with neat diagram.
4. Describe in detail about Andrew file system.
5. Discuss on file sharing semantics.
6. Explain in detail about naming in file systems.
7. Write short notes on
   i) File model
   ii) File accessing model

UNIT IV
SYNCHRONIZATION AND REPLICATION
PART A

1. What is meant by hardware and software clock?

Clock devices can be programmed to generate interrupts at regular intervals in orders that, for example, time slicing can be implemented. The operating system reads the node's hardware clock value, H(t) , scales it and adds an offset so as to produce software clock $C(t)=\alpha Hi(t)+\beta$ that approximately measures real ,physical time t for process pi.

2. What is clock resolution?

Note that successive events will correspond to different timestamps only if the clock resolution-the period between updates of the clock-value-is smaller than the time interval betw4een successive events. The rate at which events occur depends on such factors as the length of the processor instruction cycle.

3. What is clock drift?

Clock drift, which means that they count time at different rates and so diverge. The underling oscillators are subject to physical variations, with the consequence that their frequencies of oscillation differ. Moreover ,even the same clock's frequency varies with temperature. Designs exist that attempt to compensate for this variation, but they cannot eliminate it. A clock's drift rate is the change in the offset (difference in reading) between the clock and a nominal perfect reference clock per unit of time measured by the reference clock.

4. What is IAT?

Computer clocks can be synchronized to external sources of highly accurate time. The most accurate physical clocks use atomic oscillators, whose drift rate is about one part in 1013.the most accurate physical clocks use atomic oscillators, whose drift rate is about one part in 1013. The output of these atomic clocks is used as the standard for elapsed real a time, known as International Atomic Time.

5. What is CUT?

Coordinated Universal Time-abbreviated as UTC (From the French equivalent)-is as international standard for timekeeping. It is based on atomic time, but a so-called 'leap second' is inserted-or, more rarely, deleted-occasionally to keep it in step with astronomical time.

6. What is meant by external synchronization?

In order to know at what time of day events occur at the processes in our distributed system –it is necessary to synchronize the processes' clocks, C , with an authoritative, external source of time. This is external synchronization. We define these two modes of synchronization more closely as follows, over an interval of real time I:

- For a synchronization bound $D>0$, $|Ci(t)-Cj(t)|<D$ for $i=0,1,2,\ldots N$ and for a source S of UTC time, $|S(t)-Ci(t)|<D$,for i=1,2,.. and for all real times t in I.
- The clocks Ci are accurate to within the bound D.

7. What is internal synchronization?

And if the clocks C are synchronized with one another to known degree of accuracy, then we can measure the interval between two events occurring at different computers by appealing to their local clocks, even though they are not necessarily synchronized to an external source of time. This is internal synchronization.

 For a synchronization bound D > 0 and for a source S of UTC times, |S(t)-Ci(t) | < D , for all real times t in I.

 Clocks C agree with in the bound D.

8. Define NTP and its design aims.

Christian's method and the Berkeley algorithm are intended primarily for use within intranets. The Network Time Protocol (NTP) [Mills1995] defines architecture for a time service and a protocol to distribute time information over the Internet.

NTP's chief design aims and features are as follows:

 To provide a service enabling clients across the Internet to be synchronized accurately to UTC:

 To provide a reliable service that can survive lengthy losses of connectivity:

 To enable clients to resynchronize sufficiently frequently to offset the rates of drift found in most computers:

9. Enumerate the mode of synchronization in NTP servers.

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

 Multicast mode is intended for use on a high-speed LAN. One or more servers periodically multicasts the time to the servers running in other computers connected by the LAN, Which set their clocks assuming a small delay. This mode can achieve only relatively low accuracies, but ones that nonetheless are considered sufficient for many purposes.

 Procedure-call mode is similar to the operation of Christian's algorithm. In this mode, one server accepts requests from other computers, which it processes by replying with its timestamp (current clock reading). This mode is suitable when higher accuracies are required than can be achieved with multicast, or where multicast is not supported in hardware.

 In, symmetric mode is intended for use by the servers that supply time information in LANs and by the higher levels of the synchronization subnet, where the highest accuracies are to be achieved.

10. What is filter dispersion?

NTP servers apply a data filtering algorithm to successive pairs which estimates the offset o and calculates the quality of this estimates as a statistical quantity called the filter dispersion.

11. What is synchronization dispersion?

Peers with lower stratum numbers are more favoured than those in higher strata because they are 'closer' to the primary time sources. Also, those with the lowest synchronization dispersion are relatively favoured. This is the sum of the filter dispersions measured between the server and the root of the synchronization subnet.

12. What is meant by HB relation?

 Lamport called the partial ordering obtained by generalizing these two relationships the happened-before relation. It is also sometimes known as the relation of causal ordering or potential causal ordering.

 We can define the happened-before relation, denoted -> by as follow:

HB1: If process pi:e->ie' , then e->e'

HB2: For any message m, send (m) receive (m)

HB3: IF e, e' and e" are events such that e->e' then e->e"

13. What is logical clock?

 Lamport [1978] invented a simple mechanism by which the happened before ordering can be captured numerically, called a logical clock.

 A Lamport 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 a events.

 We denote the timestamp of event e at pi by L (e) , and by L(e) we denote the timestamp of event e at whatever process it occurred at.

14. Define Vector clock

Vector clocks for a system of N processes is an array of N integers

• Shortcoming of Lamport clocks:

L(e) < L(e') doesn't imply e → e'

• Vector clock: an array of N integers for a system of N processes

– Each process keeps its own vector clock Vi to timestamp local events

– Piggyback vector timestamps on messages

• Rules for updating vector clocks:

– Vi[i]] is the number of events that pi has timestamped

– Viji] (j≠ i) is the number of events at pj that pi has been affected by

VC1: Initially, Vi[j] := 0 for pi, j=1.. N (N processes)

VC2: before pi timestamps an event, Vi[i] := Vi[i]+1

VC3: pi piggybacks t = Vi on every message it sends

VC4: when pi receives a timestamp t, it sets Vi[j] := max(Vi[j] , t[j]) for j=1..N (merge operation)

15. What do you meant by distributed garbage?

An object is considered to be garbage if there is no longer any reference to it anywhere in the distributed system. The memory taken up by that object can be reclaimed once it is known as to be garbage.

16. Define Global state predicate

 A Global State Predicate is a function that maps from the set of global process states to True or False.

 Detecting a condition like deadlock or termination requires evaluating a Global State Predicate.

 A Global State Predicate is stable: once a system enters a state where it is true, such as deadlock or termination, it remains true in all future states reachable from that state.

 However, when we monitor or debug an application, we are interested in non stable predicates.

17. List the assumption considered in snapshot algorithm

– Neither channels nor processes fail

– Reliable communications ensure every message sent is received exactly once

– Channels are unidirectional

– Messages are received in FIFO order

– There is a path between any two processes

– Any process may initiate a global snapshot at any time

18. Define Failure detector.

A failure detector is a service that processes queries about whether a particular process has failed .It is often implemented by an object local to each process that runs failure detection algorithms in conjunction with its counterparts at the other processes.

19. List the properties of failure detector

A failure detector is not necessarily accurate. Most fails into the category of **unreliable failure detectors.**

- A result of unsuspected
- A result of Suspected

20. What is meant by election?

Election: choosing a unique process for a particular role is called an election

– All the processes agree on the **unique** choice

– For example, server in dist. mutex

PART B

1. Explain clocks, events and process state in details
2. Discuss in details any two algorithms used for synchronizing clocks in intranet
3. Explain logical clocks in detail.
4. Explain distributed mutual exclusion algorithms in detail.
5. Explain ring based and bully's election algorithms.
6. Explain distributed transaction in detail.

UNIT V
PROCESS AND RESOURCE MANAGEMENT
PART A

1. What is process?

Process means a program in execution. Process execution must progress in sequential order.

2. What is process migration?

The phenomenon of shifting a process from one machine to another one which is called process migration.

3. What is preemptive process migration?

Preemptive process transfers involve the transfer of a process that is partially executed. This transfer is an expensive operation as the collection of a process's state can be difficult.

4. What is load balancing?

Load balancing is nothing but the allocation of tasks or jobs to processors to increase overall processor utilization and throughput. Actually load balancing is done by process migration.

5. What is load?

Load may define as number of tasks are running in queue, CPU utilization, load average, I/O utilization, amount of free CPU time/memory, etc.

6. List desirable features of good process migration mechanism.

Features are transparency, efficiency, minimal interference, minimize freezing time and minimal residual dependencies.

7. What is non-preemptive process migration?

Non-preemptive process transfers involve the transfer of processes that have not begun execution and hence do not require the transfer of the process's state. In both types of transfers, information about the environment in which the process will execute must be transferred to the receiving node.

8. List any three challenges of process migration.

1. Process state capturing and transfer.
2. Scheduling.
3. System call.

9. What are the strategies for the migration of files?

1. If the file is locked by the migrating process and resides on the same system, then transfer file with the process.
2. If the process is moved temporarily, transfer the file only after an access request was made by the migrated process.
3. If the file is shared by several distributed processes, do not transfers file.

10. What is sender initiated distributed heuristic?

This algorithm requires no coordinator whatsoever. if a machine decides that it should not run its job locally, it picks a machine random and sends it a probe message. If the randomly selected machine cannot run the job, another machine is picked random and a probe sent to it.

11. Define the term thread.

A minimal software processor in whose context a series of instructions can be executed. Saving a thread context implies stopping the current execution and saving all the data needed to continue the execution at a later stage.

12. What is user level thread?

User level thread uses user space for thread scheduling. These threads are transparent to the operating system. User level threads are created by runtime libraries that cannot execute privileged instructions.

13. What is kernal level thread?

In kernel level thread, thread management is done by kernel. OS support the kernel level thread. Since Kernel managing threads, Kernel can schedule another thread if a given thread blocks rather than blocking the entire processes.

14. What is address space in process migration?

Address space can be defined with respect to an existing execution environment. For example the newly created child process physically shares the parent's text region and has heap and stack regions that are copies of the parent's in extent. When parent and chile share a region, the page frames belonging to the parent's region are mapped simultaneously into the corresponding child region.

15. List the benefits of process migration.

1. Better response time and execution speed-up
2. Reducing network traffic.
3. Improving system reliability.
4. Higher throughput and effective resource utilization.

16. List the types of process scheduling techniques.

1. Task assignment approach
2. Load balancing approach
3. Load sharing approach

17. What is task assignment approach?

In task assignment approach, each process submitted by a user for processing is viewed as a collection of related tasks and these tasks are scheduled to suitable nodes so as to improve performance.

18. List the major groups of pthreads.

1. Thread management
2. Mutexes
3. Condition variable
4. Synchronization

19. List the operation of pthread.

1. pthread-create()
2. pthread-join()
3. pthread-self()
4. pthread-detach()
5. pthread-exit()
6. pthread-sigmask()
7. pthread-kill()

20. List the issues in designing load balancing approach.

1. Load estimation policy
2. Process transfer policy
3. Location policy
4. Priority assignment policy
5. Migration limiting policy

PART B

1. Explain in detail about process migration.
2. Explain thread, thread models and issues in detail.
3. Explain in detail about thread implementation.
4. Explain task assignment approach in detail.
5. Explain load balancing approach in detail.
6. Explain load sharing approach in detail.