Total No. of Questions :10]

SEAT No. :

P3989

[Total No. of Pages :4

[4959]-1134

B.E. (Information Technology)

DISTRIBUTED SYSTEM

(2012 Course) (Semester - II) (End - Semester) (414461)

Time : 2½ Hours] *[Max. Marks :70*

Instructions to the candidates:

1) ***Answer Q.1 or Q.2, Q.3 or Q.4, Q.5 or Q.6,Q.7 or Q.8, Q.9 or Q.10.***
2) ***Figures to the right indicate full marks.***
3) ***Neat diagrams must be drawn wherever necessary.***
4) ***Assume suitable data, if necessary.***

Q1) a) Distinguish between: **[6]**

i) Buffering and Caching

ii) RMI and RPC

b) Consider two communication services for use in asynchronous distributed systems. In service A, messages may be lost, duplicated or delayed and checksums apply only to headers. In service B, messages may be lost. Delayed or delivered too fast for the recipient to handle them, but those that are delivered arrive order and with the correct contents. **[4]**

i) Describe the classes of failure exhibited by service A and service B. Classify their failures according to their effect on the properties of validity and integrity. Can service B be described as a reliable communication service?

ii) Consider a pair of processes X and Y that use the communication service B from above to communicate with one another. Suppose that X is a client and Y a server and that an invocation consists of a request message from X to Y (that carries out the request) followed by a reply message from Y to X. Describe the classes of failure that may be exhibited by an invocation.

OR

P.T.O.

Q2) a) Explain Distributed Object Model with respect to: **[6]**

i) Actions and

ii) Garbage collection

iii) Exceptions

b) Explain in nutshell, the different techniques for failure handling in a Distributed system. **[4]**

Q3) a) Explain Remote object reference and Remote interface in Distributed Object Model with suitable example. **[5]**

b) What is marshaling? How marshaling and serialization is used in communication between a client and a server? **[5]**

OR

Q4) a) Explain three communication primitives of Request-reply protocol along with message structure used in information transmission. **[4]**

b) Describe general organization of CORBA system with the help of suitable diagram. Why there is no explicit data-typing in CORBA CDR? **[6]**

Q5) a) What are the Network Time Protocol's aims and features? Explain the modes through which NTP servers synchronize with one another. **[8]**

b) Explain following points related to fault tolerance issues in Distributed Systems: **[8]**

i) Availability

ii) Reliability

iii) Failure Models

iv) Tripple modular redundancy

OR

Q6) a) A client attempts to synchronize with a time server. It records the round-trip times and timestamps returned by the server in the table below. **[8]**

Round-trip (ms)	*Time (hr:min:sec)*
22	10:54:23.674
25	10:54:25.450
20	10:54:28.342

Which of these times should it use to set its clock? To what time should it set it? Estimate the accuracy of the setting with respect to the server's clock. If it is known that the time between sending and receiving a message in the system concerned is at least 8 ms, do your answers change?

b) Describe implementation of ordered multicast in a non-overlapping group. **[8]**

Q7) a) Explain the objectives and architecture of Hadoop Distributed File System in details. **[8]**

b) How is the X.500 directory service implemented? **[8]**

OR

Q8) a) List the different Distributed File System Requirements? Explain the abstract File Service architectural model with neat diagram. **[8]**

b) Write a detailed note on Domain Name System. **[8]**

Q9) a) Write short note on the following (Any 2): **[10]**

i) Cloud Computing

ii) Secure Channel

iii) Cryptographic Algorithms.

b) State and explain various security mechanisms for achieving security in distributed systems. **[8]**

OR

Q10)a) Write short note on the following: **[10]**

i) Applications of cryptography and political obstacles.

ii) Symmetric and Asymmetric Algorithms.

b) Explain the Secure Mobile Code in brief with reference to JAVA sandbox. **[8]**

EEE