[Time: Three Hours] [Marks:80]

Please check whether you have got the right question paper.

N.B: (1)Question No. **1 is Compulsory**.

(2) Attempt any **three** question out of remaining **five**.

1	(a)	Differentiate between homogeneous and heterogeneous distributed database management systems with example.	10
	(b)	Discuss the phases of distributed query processing with neat diagram.	10

2 Consider the following relations:

BOOKS(Book#, Primary_author, Topic, Total_stock, $price)

BOOKSTORE(Store#, City, State, Zip, Inventory_value)

STOCK(Store#, Book#, Qty)

Total_stock is the total number of books in stock and Inventory_value is the total inventory value for the store in dollars.

	(a)	Design a global schema for above database. Give an example of two simple predicates that would be meaningful for the BOOKSTORE relation for horizontal partitioning	08
	(b)	How would a derived horizontal partitioning of STOCK be defined based on the partitioning of BOOKSTORE?	04
	(c)	Show predicates by which BOOKS may be horizontally partitioned by topic. Show how the STOCK may be further partitioned from the partitions in (b) by adding the predicates in	04
	(d)	(c).	04

3	(a)	What is distributed data independence? Explain how distributed data independence is provided by the architecture of DDBMS.	10
	(b)	Discuss the algorithms used for distributed Deadlock preventions.	10

4	(a)	Compare various locking based concurrency control protocols.	10
	(b)	XML document of 'Restaurant Menu Card' has food items, categorized into Starters, Drinks, Chinese, South and Punjabi. Each food item element contains name, cost, calories, and veg/non-veg flag.	10
		i. Write DTD rules for above XML document.	
		ii. Write XML Schema for above XML document.	

5	(a)	Discuss the different communication structures for 2PC.	10
	(b)	Describe the distributed R* query optimization algorithm.	10

6	Write a short notes on **(Any two)**		20
	(a)	Distributed transaction management	
	(b)	Multi-version TO algorithm	
	(c)	Transparency in Distributed Database Design	
	(d)	Schema architecture of federated MDBS	