Student's name:
Class:

Theoretical Exercises
Module: Distributed Systems
Chapter 1: Introduction and Architectures of Distributed Systems

Question 1: What is the role of middleware in a distributed system?

Question 2: Explain what is meant by (distribution) transparency, and give examples of different types of transparency.

Question 3: Why is it sometimes so hard to hide the occurrence and recovery from failures in a distributed system?

Question 4: Why is it not always a good idea to aim at implementing the highest degree of transparency possible?

Question 5: What is an open distributed system and what benefits does openness provide?

Question 6: Describe precisely what is meant by a scalable system.

Question 7: Scalability can be achieved by applying different techniques. What are these techniques?

Question 8: If a client and a server are placed far apart, we may see network latency dominating overall performance. How can we tackle this problem?

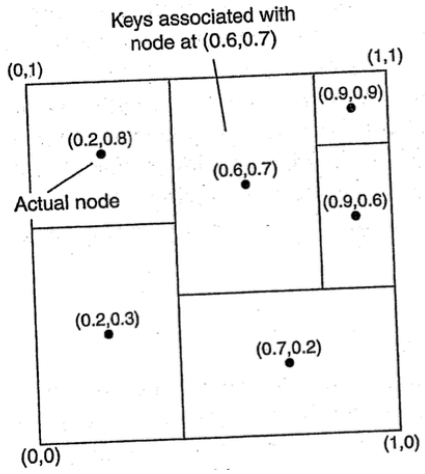
Question 9: What is a three-tiered client-server architecture?

Question 10: What is the difference between a vertical distribution and a horizontal distribution?

Question 11: In a structured overlay network, messages are routed according to the topology of the overlay. What is an important disadvantage of this approach?

Question 12: Consider a chain of processes P1 , P2 , ..., P n implementing a multitiered client-server architecture. Process Pi is client of process Pi+1, and Pi will return a reply to Pi−1 only after receiving a reply from Pi+1 . What are the main problems with this organization when taking a look at the request-reply performance at process P1?

Question 13: Considering that a node in CAN knows the coordinates of its immediate neighbors, a reasonable routing policy would be to forward a message to the closest node toward the destination. How good is this policy?



Question 14: What are the benefits of Microservices architecture compared to monolithic architecture?

Question 15: Design yourself an e-commerce system using Microservices architecture.