

## Distributed Data Management

1. Which properties should a good hash function have?
2. Explain the reasons behind using a DHT to index the data in a P2P Network?
3. Assume you are designing a DHT application for a large discount store chain. The DHT is supposed to store simple customer feedback and statistics. Each time a customer pays his bill, the cashier asks for the customer's ZIP-code and if he was satisfied with his shopping experience or not. Then a record is stored in the DHT with ZIP as key and satisfaction degree and monetary amount paid as data payload.  
Is  $hash = key \bmod numOfNodes$  a good hash function for this scenario? Why or why not?
4. Assume a file-sharing application using direct storage and a cryptographic hash function (e.g. SHA-1). Does such a system already have a load balancing mechanism? Why?
5. Please argue: A Chord ring has logarithmic routing complexity.
6. Explain the main differences between Chord and CAN.
7. What happens if a Chord ring does not use the stabilize function?
8. How are "stabilize" and "ping-pong" related?
9. Assume you are designing a pseudo-legal P2P system for sharing various works produced by university students (e.g. seminar papers, homeworks, Bachelor thesis, Master thesis, "Studienarbeit", etc.).  
The base idea is that each Student can share all his works with his fellow students. Whenever a student has to write a seminar paper or has to solve some homework's, he can look up in the system for similar works to obtain some "inspirational ideas". Which topology should such a system use (e.g. central P2P, pure P2P, hybrid P2P, DHT, etc.)? Why?
10. Assume you have decided to implement a DHT for the previous exercise, and you have two options for the implementation: Chord and SYMPHONY. Which one would you choose and why?