1. Query Optimization 1
Given the following relations:

Customer(cusNr, name, PLZ, city, street) with 10 000 entries, distributed over 20 cities
Products(prodNr, description) with 5 000 entries
DeliveryScope(delNr ->Delivery, prodNr ->Products) with 200 000 entries and 5 products per delivery
Delivery(delNr, cusNr ->Customer, Date) with 40000 entries, 100 deliveries per day (in average)

(a) Formulate the following query with SQL: „Select all customer names, who received a delivery containing the product with number 101.“

(b) Create at least 3 different equivalent relational algebra expressions for this query.

(c) Find out how many operations are required for the database system to execute every single query (cf. slide 1-15ff). Which (algebra) expression is the optimum?

(d) Optimize the query „Select all customer names from Leipzig, who received a delivery on 01.03.2004 . The product descriptions of this delivery should be selected as well.“ like exercise a to c.

(e) Assume additionally, that the relations Customer and Delivery are ordered by the customer number (cusNr). Analyze the costs (number of operations) for the Nested-Loop-Join and the Merge-Join operation respectively.

2. DB Architecture
Depending on the administration level, different representations for database system architectures exist. Here, we want to compare the both best known. Furthermore, the 5-Layer architecture will be checked against the “Rules of Codd.”

(a) Point out similarities (intersections) and differences between the ANSI-SPARC-architecture (slide 1-7 / 2-30) and the 5-Layer Architecture (slides 2-31ff).
(b) “How does it fit to the Schema Architecture (logical/physical data independence from DB1)?

(c) Regarding the architectures, where are the following objects/components situated:
   - physical storage,
   - tables in form of relations,
   - access structures and
   - views on a database schema

(d) Describe at which positions in the 5-Layer architecture (and how) the following “Rules of Codd, are realized or have to be realized respectively.
   - Integration (ensures uniform, non-redundant management of data),
   - Operations,
   - Catalogue (contains data definition and description),
   - Multi user views (extract use-oriented views on data),
   - Consistency/Integrity (enforces keeping defined constraints),
   - (Data) Privacy (in form of access control),
   - Transactions ,
   - Synchronization (for multi user mode),
   - Data Backup.

Good Luck!