

Final Exam Topics

CS 3200, Fall 2010

SQL

- Compose and understand basic SQL queries
 - Might need intersection, union, negation
- Compose and understand Nested queries
- Compose and understand aggregate queries with grouping
 - Simple aggregate computation
 - Nested version, e.g., find name of oldest sailor for each rating
- Remember: create query step-by-step
- Don't worry about detailed issues with NULL values, but be aware of potential problems

Constraints

- What kind of constraints can primary keys, foreign keys, and NOT NULL (in table definition) not enforce?
- When do we need CHECK, ASSERTION, TRIGGER
 - Will try to avoid tricky scenarios
- What goes into Event, Condition, Action of a trigger

Relational Algebra and Calculus

- Don't need to be able to write queries in algebra or calculus
- Don't need to know details about calculus
- Know the algebra operators we discussed in class so that you understand a given query plan
- Why do we care about relational algebra and calculus?

DB App Development

- Will not ask you to write a JDBC program, but you should be able to tell what a given one is doing
- Know the basics
 - How does the basic JDBC architecture for a type 4 driver look like
 - What is a cursor
 - How do we declare transactions
 - How do we set isolation level
- Why are stored procedures useful

Storage, Indexing

- Cost of accessing disk versus memory
- What is an RID and what do we do with it
- What are heap file and sorted file
- Basic idea of B-tree
 - Number of leaves, how are they sorted, what do they contain
 - What is node fanout and how does it affect tree height
 - Abstract "algorithm" for finding data entries for a specific value or range
- Basic idea of hash index
 - Basic structure of static hashing
 - Algorithm for finding data entries for a specific value

Storage, Indexing (cont.)

- Different alternatives for what to store in a data entry
 - Distinction between data entry and data record
- For what type of query would one consider each of these storage options?
- Compute number of page accesses (like HW 7)
- When are composite search keys useful in a B-tree
- What are index-only plans

Query Evaluation

- Basic ideas
 - What is a query plan
 - What are the basic steps of query optimization
 - What access paths does a DBMS optimizer consider for a query
- Cost calculations like we discussed in class
 - Know nested loops, block nested loops, and sort-merge join
 - Know how to estimate number of matching tuples based on given information about data and query
 - Exact numbers (HW 7) vs. approximations (uniformity assumption, attribute independence)
- Be able to compare two given query plans

Transactions

- Why are they so important
- Possible anomalies when interleaving execution of transactions
- Serial vs. serializable schedules
- Locking to achieve serializability
 - Types of locks
 - Phantom problem
- How can deadlocks happen with 2PL and what can we do about them

Transaction (cont.)

- Why can locking lead to poor performance
- Tradeoffs between performance and isolation (HW 8)

Recovery

- Main ideas of the ARIES approach
 - Why do we need to log DB actions
 - What is logged
 - When do we log
 - How do we know from the log which transactions were complete when the system crashed
 - Which transactions crashed in the middle of doing their work
 - Main ideas of how to perform REDO and UNDO