

CS 525 Advanced Database Organization - Spring 2017

Mon + Wed 1:50 - 3:05 PM, Room: Stuart Building 111

Instructor: Boris Glavic, Stuart Building 226 C, Phone: 312 567 5205, Email: bglavic@iit.edu

Office Hours: Monday, 12:00 pm - 1:00 pm

Instructor Webpage: www.cs.iit.edu/~glavic/

Course Webpage: www.cs.iit.edu/~cs525/

Course Description:

Databases management systems are a crucial part of most large-scale industry and open-source systems. This course provides comprehensive coverage of issues associated with database system development and an in-depth examination of structures and techniques used in contemporary database management systems (DBMSs). Students will learn about the inner workings of these exciting systems: Which algorithms are used? What are typical architectures used to build a system as complex as a DBMS? What are implementation strategies? These questions and more will be answered during the course.

The course is highly applied, emphasizing practical skills and habits through a series of programming assignments during which students will develop their own tiny DBMS like engine. We will cover the most important aspects/components of a DBMS: **storage and buffer management, indexing, query optimization, query execution, and concurrency control and recovery.**

Course Material:

The following text books will be helpful for following the course and studying the presented material. All four textbooks have their merits, but any one should be sufficient as reading material.

Elmasri and Navathe , **Fundamentals of Database Systems** , 6th Edition , Addison-Wesley , 2003

Ramakrishnan and Gehrke , **Database Management Systems** , 3rd Edition , McGraw-Hill , 2002

Silberschatz, Korth, and Sudarshan , **Database System Concepts** , 6th Edition , McGraw Hill , 2010

Garcia-Molina, Ullman, and Widom, **Database Systems: The Complete Book**, 2nd Edition, Prentice Hall, 2008

The slides will be made available on the course webpage. Furthermore, for the brave, the webpage will list research papers related to the topics covered in the course.

Prerequisites:

- *Courses:* CS425
- Programming experience in C, C++ or other low level languages
- Unix OS and file system knowledge is helpful
- Data structures (.e.g., CS401)

Course Details:

The following topics will be covered in the course:

- Introduction
 - Relational Algebra
 - DBMS Architecture
- Hardware Characteristics affecting DBMS Design
 - Read/Write Properties of Disks
 - RAID Storage
 - Memory Hierarchy
- Disk Storage and Buffer Management
 - Physical Tuple Layout
 - Page Layout
 - Tuple IDs
 - Buffer Replacement Strategies
- Indexing and Hashing
 - B-Tree-Family Indices
 - Hashing
- Query Optimization
 - Logical Optimization
 - Equivalence Transformations
 - Physical Optimization
 - Join Reordering
 - Cost Estimation
- Query Execution
 - Pipelining
 - Push vs. Pull based Execution
 - Access Methods
 - Join Methods
 - Grouping and Aggregation
 - Other Operator Implementations
 - External Sorting
- Recovery

- Write Ahead Log (WAL)
- Algorithms for Recovery and Isolation Exploiting Semantics (ARIES)
- Concurrency Control
 - Serializability
 - Two-Phased Locking (2PL)
 - Implementing of Locks
- Advanced Topics
 - Distributed Database Systems
 - Datawarehousing
 - Parallel Query Execution
 - Technics for Executing Nested Queries and Un-nesting
 - Additional Index Structures
 - Relation to Large-Scale Data Analytics

Workload and Grading Policies:

Programming Assignments:

There will be several programming assignments during the course. Starting from a storage manager you will be implementing your own tiny database-like system from scratch. You will explore how to implement the concepts and data structures discussed in the lectures and readings. The assignments will require the use of skills learned in this course as well as other skills you have developed throughout your program. Each assignment will build upon the code developed during the previous assignment. In the end there will be an *optional* assignment for extra credit. Each of the regular assignments will have optional parts that give extra credit. All assignments have to be implemented using C/C++.

- **Assignment 1 - Storage Manager:** Implement a storage manager that allows read/writing of blocks to/from a file on disk
- **Assignment 2 - Buffer Manager:** Implement a buffer manager that manages a buffer of blocks in memory including reading/flushing to disk and block replacement (flushing blocks to disk to make space for reading new blocks from disk).
- **Assignment 3 - Record Manager:** Implement a simple record manager that allows navigation through records, and inserting and deleting records.
- **Assignment 4 - B⁺-Tree Index:** Implement a disk-based B⁺-tree index structure.
- **Potential Optional Assignment:**
 - Implement a standard operator algorithm on top of your record manager, e.g., nested loop join, hash-aggregate,

Mid Term and Final Exam:

There will be a mid term and a final exam covering the topics of the course.

Quizzes:

There will be quizzes during the course. The main objective of the quizzes is for you and the instructor to evaluate how well you internalized the topics covered in the course.

Grading Policies:

See the course webpage for policies regarding late assignments and plagiarism.

- Programming Assignments: 50% (10% + 10% + 15% + 15%)
- Mid Term Exam: 20%
- Final Exam: 20%
- Quizzes: 10%

Course Objectives:

After attending the course students should:

- Understand the design decisions behind textbook DBMS architectures
- Know the trade-offs of various storage organization techniques
- Be able to build parts of a small-sized data processing system from scratch
- Understand the basics of query optimization
- Know standard implementations of relational operators such as join, aggregation, and set operations
- Be able to estimate the cost of executing an operator/query based on DB statistics
- Know standard database indexing techniques
- Understand concurrency control and recovery mechanisms