September 5, 2017

yarowsky@jhu.edu

Johns Hopkins University

Department of Computer Science

Prof. David Yarowsky

(601) 315/415/615

DATABASES
Instructor: Prof. David Yarowsky
TAs: Eric Calder, Hackerman 324 GMalone 239
410-516-5372
yarowsky@jhu.edu, calder6@jhu.edu

Office Hours: Instructor - Tuesday/Thursday after class and by appointment.
TA - TBA, special review sections, and by appointment.

Classroom: Hackerman B17

Meeting Time: Tu, Th: 3:00-4:15 PM

410.315/415/615 - DATABASES

Other Potentially Useful Textbooks:
- Other Potentially Useful Textbooks:
- Other Potentially Useful Textbooks:
- Other Potentially Useful Textbooks:
- Other Potentially Useful Textbooks:
Course Requirements

- Class Participation: 5%
- Homeworks (4): 25%
- Midterm: 20%
- Final Exam: 25%
- Final Project: 25%

Homeworks will include paper-and-pencil exercises and MySQL implementation exercises. The Final exam will be cumulative.
Lateness Policy

One homework assignment may be handed in up to 5 days late without penalty.

Final projects handed in late will receive a penalty of 10% for every day late.

No other late homeworks will be accepted.

Day late: •
Academic honesty is required in all work you submit to be graded. You must solve all homework and programming assignments entirely on your own (Homeworks 1/2/4), unless group work is specified in writing. If you use fragments of source code from sources other than your text (such as on-line resources), you must put a reference to that effect in your homework submission. Students who cheat falsifying program output or results is prohibited. Please see your professor if there are any questions about what is permissible. Students who cheat will suffer a serious course grade penalty in addition to being reported to university officials. You must abide by JHU’s Ethics Code, available at http://jhunix.hcf.jhu.edu/~ethicsbd.
Grades and upperclass students are encouraged to enroll.

- Nevertheless, 601.415/615 should be manageable by advanced undergraduates.
- Exams will differ somewhat and will be graded on a different scale.
- For 601.315, homeworks in 601.415/615 will include additional component(s) not required in the final project will include additional problems.
- They will differ primarily in terms of assignments and grading.
- 601.315/415/615 will be share common lectures.

601.315 vs. 601.415/615
Databases (315/415/615, Fall) and Database Systems (316/416/616, Spring) are complementary courses and make a natural course sequence.

315/415/615 focuses on:
- Design and use a database
- Formal database models, theory and foundations
- How to design and use a database
- Object-oriented and XML-based data models and future directions
- Database programming languages, especially SQL and PL/SQL
- Database internals and systems, including query and join processing
- Indexing, the organization, estimation and optimization

In contrast, 316/416/616 will focus on:
- Implement a database for a novel task (including practical execution)
- The final project will be application-focused (e.g., how to design an
  including data mining and natural language interfaces)

315/415/615 focuses on:
- How to design and use a database
- Formal database models, theory and foundations
- How to design and use a database
- Object-oriented and XML-based data models and future directions
- Database programming languages, especially SQL and PL/SQL
- Database internals and systems, including query and join processing
- Indexing, the organization, estimation and optimization

601.315/415/615 vs. 601.316/416/616:
- Databases (315/415/615, Fall) and Database Systems (316/416/616, Spring) are complementary courses and make a natural course sequence.

(see below)
The course project(s) will focus on database system internals and their development.

- Database architectures, streaming, and partitioning.
Can I take 316/416/616 as a stand-alone course without 315/415/615?

Yes, 316/416/616 does not have 315/415/615 as a formal prerequisite.

Graduate students who have prior database employment experience or have taken a prior course in database systems are normally expected to begin directly with 416.

Anyone with a research focus in the databases area should certainly begin directly with 416.

However, either through prior employment or via a prior course, you should have some database experience before taking 316/416/616, or you have some database experience before taking 315/415/615 as a formal prerequisite.
Can I take 315/316 or 415/416 or 616/616 as a 2-course sequence?

Yes.

The instructors will work to make this a natural 2-course sequence. There will be modest overlap of material (10%) but taught via different perspectives and emphases, and will serve as a good refresher. If you have not taken a prior course in databases and are interested in both the theory/applications and systems sides of the field, then this sequence makes a lot of sense and is encouraged.

However, if you have already had a prior course in databases, or intend to continue in database systems research, then you are strongly encouraged to take 316/416/616 and then another advanced follow-on course in databases and/or storage systems, then take 415/515/615. Then you will have covered all of the database material that is covered in 315/316 or 415/416, but from different perspectives. Otherwise, the content will be taught as separate courses.
Can I take 315/416 as a sequence?

Yes, 416 does not require 415 as a prerequisite, but you should have done well in 315 and be prepared to do some background catching to meet the expectations of the 416 instructor.
Can I take 415/316 as a sequence?

Yes, if you are an undergraduate and would like to continue focusing on database systems and database systems internals but a less difficult level, then this sequence could make sense.
Students will be able to select final projects of interest to them from a fairly diverse set of options. Details will be provided in class.

For most projects, students will be required to populate and test their implemented database design with substantial quantities of real world data extracted from the world wide web or other online sources. Implememted database design will be due in early November, including a full database implemetation system specification and design.

A project proposal will be due in early November, including a detailed.

Students may work in teams of 1 or 2 people.

Final Projects
Sample Final Project Domains (previous years)

- Stock market news and price correlations (data mining)
- Internet proxy server database
- Human genome databases
- Bibliographic database for medical robotics
- Astronomical and pharmaceutical databases for research support
- Representations of acoustics data for speech recognition
- Fantasy hockey league
- Connecticut volunteer emergency rescue organization
- JHU Fencing club and Anime club
- Olympic sports data
- Movie industry data (directors, producers, actors, films, etc.)
- World geography and population data (from CIA World Fact Book)
- Used car information (by model and year, from Edmunds)
( SELECT Countryname FROM Quake WHERE Year = 2013 )

WHERE magnitude IN ( SELECT MAX ( magnitude )
FROM Quake
WHERE magnitude IN
SELECT Countryname
FROM Quake

Which country had the most powerful earthquake in 1994?
Which country had the most powerful earthquake in 1996?

Which earthquake had magnitude greater than 7 on the same continent?

List the years in which there are at least two earthquakes of magnitude greater than 7 on the same continent.

What was the average magnitude of 1996 earthquakes in Asia?
What was the average magnitude of the most powerful earthquakes in China?

Which country had the greatest number of earthquakes in 1996?

Natural Language Interfaces to an earthquake database

Sample Final Project Domains (continued)
SEGMENT 1 - Survey of Data Models

- Object-Oriented models
  - Relational query languages: SQL, QBE (Query-by-Example)
  - Formal representations: Relational algebra and calculus
- Relational model
  - Entity-Relationship model (formal conceptual framework)
- Network and Hierarchical models (of historical interest)
SEGMENT 2 - DATABASE DESIGN AND IMPLEMENTATION

Formal Analysis:
- Integrity constraints
- Domain constraints
- Triggers

Practical Database Implementation:
- Normalization
- Functional dependencies
- PL/SQL and stored procedures
- Embedded SQL (in a host language like C or Perl)
- MySQL (a detailed exploration)

- MySQL (a detailed exploration)
SEGMENT 3 - Database System Internals

- Distributed databases
- Parallel databases
- Database system architectures
- Database security
- Recovery systems
- Transaction processing
- Query optimization
- Query processing
SEGMENT 4 - Emerging Technologies and Applications

Decisionsupport systems
Datamining
Datawarehousing
Naturallanguage interfaces
Spatial, geometric and geographic databases
xml-based datamodels
DNA and Human Genome databases
Very large text databases and information retrieval
Multimedia Databases (image, sound, video, etc.)
The impact of the WWW on database technology (and v.v.)

601.466 - Information Retrieval and Web Agents