## *Syllabus*: CSCIX370, Database Management – Spring 2018

**Instructor name:** I. Budak Arpinar

**Instructor accessibility**: *office location*: Hardman 225, *office hours:* Tue: 3:15-4:15, Wed: 3:20-4:20 (starting from Jan 10th), *e-mail:* [budak@cs.uga.edu](mailto:budak@cs.uga.edu), *web:* [www.cs.uga.edu/~budak](http://www.cs.uga.edu/~budak), *TA:* *TBD*.

**Lines of communication:** All inquiries regarding the course can be sent to instructor or TA. Please use a subject line prefix “X370:” in your e-mails (e.g. “X370: Project 1 question”). There will also be a Piazza discussion group. This will be used for staff to communicate with students. It is also for students to post messages with technical or logistical questions regarding the course that are expected to be of interest to most students, and for staff members to post their replies.

**Course description:** CSCIX370 provides the student with a comprehensive introduction to the design of databases and the use of database management systems for applications. The course presents two perspectives on database management: (1) internal workings of a database management system, and (2) application development using a database server.

Primary vehicle for evaluation will be exams and team-oriented projects.

**Course objectives or learning outcomes**: Throughout the course we will cover the relational model and, SQL - the standard language for creating, querying, and modifying relational and object-relational databases. From a more theoretical perspective, we will cover relational algebra, relational design principles based on functional dependencies and normal forms, and the entity-relationship approach to database design. A variety of other issues important to database designers and programmers will be covered, including indexes, transactions, and integrity constraints. Various indexing techniques and their advantages and disadvantages at certain situations will be discussed. Basics of query processing and optimization will be presented as time permits. At different stages in the course, we will address several practical topics such as using MySQL, programming in SQL, JDBC and other database tools.

**Prerequisites, co-requisites**: The programming projects will use Linux/Windows/Mac and MySQL database management system, and the Java programming language. We will assume students are already proficient in Java.

**Required course text:** Fundamentals of Database Systems, 7th Edition, Ramez Elmasri, and Shamkant B. Navathe, 2016.

**Topical outline:**

1. Part 1: Introduction to Databases   
   Chapter 1: Databases and Database Users   
   Chapter 2: Database Systems Concepts and Architecture
2. Part 2: Conceptual Data Modeling and Database Design   
   Chapter 3: Data Modeling Using the Entity Relationship (ER) Model   
   Chapter 4: The Enhanced Entity Relationship (EER) Model   
   Chapter 9: Relational Database Design by ER- and EER-to-Relational Mapping
3. Part 3: The Relational Data Model and SQL   
   Chapter 5: The Relational Data Model and Relational Database Constraints   
   Chapter 6: Basic SQL   
   Chapter 7: More SQL: Complex Queries, Triggers, Views, and Schema Modification   
   Chapter 8: The Relational Algebra
4. Part 4: Database Programming Techniques

Summary of Chapters 10 & 11

1. Part 6: Database Design Theory and Normalization   
   Chapter 14: Basics of Functional Dependencies and Normalization for Relational Databases   
   Chapter 15: Relational Database Design Algorithms and Further Dependencies
2. Part 7: File Structures, Hashing, Indexing, and Physical Database Design   
   Chapter 16: Disc Storage, Basic File Structures, Hashing, and Modern Storage Architectures   
   Chapter 17: Indexing Structures (B+trees and hashing) for Files and Physical Database Design
3. Part 8: Query Processing and Optimization   
   Chapter 18: Strategies for Query Processing   
   Chapter 19: Query Optimization
4. Part 9: Transaction Processing, Concurrency Control, and Recovery

Summary of Chapters 20, 21 & 22

**Grading policy:** The components of the final grade will be distributed as follows (subject to slight adjustments, if necessary):

* 15% MT Exam
* 25% Final
* 36% Small projects (4)
* 20% Term project
* 4% Homework (4) – from textbook (do NOT c/p from web)
* Bonus/Penalty: Attendance (-3… +3)
* Bonus: On-line course evaluation (+1)
* Other (small) bonus opportunities in exams and projects

# **The following table will be used for letter grading. A X.5 or greater score will be rounded up to X+1; otherwise it will be rounded down to X (e.g., 92.50 will be rounded up to 93; 92.49 will be rounded down to 92).**

|  |  |  |
| --- | --- | --- |
| A | 94 | above |
| A- | 90 | 93 |
| B+ | 87 | 89 |
| B | 84 | 86 |
| B- | 80 | 83 |
| C+ | 77 | 79 |
| C | 74 | 76 |
| C- | 70 | 73 |
| D | 60 | 69 |
| F | below | 59 |

# ****Projects** (group):** Lectures in all topics are complemented by a significant term project for developing a database application, and a set of smaller projects. The term project will require a proposal for a comprehensive database application and must be approved before implementation.

Each project group will include 4 members (self-composed). Each project will contain detailed submission instructions as well as some test data (if relevant) and grading criteria. A project will require a peer-review from individual team members.

# ****Late Policy:**** Programming work will be submitted electronically through ELC. Programming work submitted after the deadline, but less than 24 hours late will be accepted but penalized 10%, programming work submitted more than 24 hours but less than 48 hours late will be penalized 20%, and programming work submitted more than 48 hours but less than 72 hours late will be penalized 30%. No programming work will be accepted more than 72 hours late. This late policy will be strictly enforced.

# ****Re-grade Policy:**** Please send an email to TA stating that you believe that you deserve a project re-grade, and explaining precisely why. Please include your name, and the number/title of the project in question.

**Attendance policy:** The instructor will check attendance at random days (approximately 10 times per semester). All unexcused absences will be counted. Students with good attendance records will be rewarded with 1, 2 or 3 bonus points added to their overall score. Students with bad attendance records will be penalized with 1, 2 or 3 points deduction from their overall scores based on number of absences.

**Policy for make-up of examinations:** No make-up exam will be given in cases of undocumented absences. A make-up exam will be possible if a documented health or family emergency is present.

**Honor Code and Academic Honesty Policy:** As a University of Georgia student, you have agreed to abide by the University’s academic honesty policy, “A Culture of Honesty,” and the Student Honor Code. All academic work must meet the standards described in “A Culture of Honesty” found at: https://ovpi.uga.edu/academic-honesty/academic-honesty-policy. Lack of knowledge of the academic honesty policy is not a reasonable explanation for a violation. Questions related to course assignments and the academic honesty policy should be directed to the instructor.

# ****Programming Work:**** Each team is expected to submit their original work. On many occasions, it is useful to ask others (the instructor, the TA, or other students) for hints or debugging help, or to talk generally about programming strategies. Such activity is both acceptable and encouraged, but you must indicate any assistance (human or otherwise - except Piazza discussions/TA/instructor help) that you received. In any event, you are responsible for coding, understanding, and being able to explain on your own all project work that you submit. We will pursue aggressively all suspected cases of violations, and they will be handled through official university channels. If you have any questions about this policy or about the degree to which we will pursue violations, please discuss your concerns with the course staff immediately.

# Electronic device use policy (in the classroom): You are allowed use your laptops, tablets and phones (please silence them) if you are involved in an academic work related to this course. These academic activities include coding, reading course related content, note taking etc. Forbidden activities include gaming, reading news, chat etc. A permissible exception is checking e-mails.

**Disability and health:** Students with a disability or health-related issue who need a class accommodation should make an appointment to speak with the instructor as soon as possible.

**Syllabus changes:** The course syllabus is a general plan for the course; deviations announced to the class by the instructor may be necessary.