Department of Electrical and Computer Engineering

Faculty of Engineering and Architecture

American University of Beirut

Course Information

**Course title:** Database Systems

**Course number:** EECE 433

**Instructor: Dr. Hazem Hajj**

* **Class I**:

Sessions: 20303 and 20304;

**Class Times**: T-TR 11:00 – 12:15 PM; Bechtel 403

**LAB Times:** 2:00- 4:45 PM; Tuesdays

* **Class II**:

Sessions: 23167 and 23168;

**Class Times**: M-W-F 8:00 – 8:50 AM; Bechtel 537

**LAB Times:** 2:00- 4:45 PM; Wednesdays

* **Office Hours**:
	+ Monday-Friday: 9:00-10:00 AM, or by appointment
	+ M-W-F: 10:00-11:00 AM, or by appointment

**Lab and/or Project:**

* Lab Instructor: **Mr. Zaher Kanafani**
* Regular assignments on weekly basis starting from second week.
* Lab and Project Questions should be directly addressed to Lab instructor.
* Lab Assignments should be handed to Lab Instructor directly.
* For the semester project, students will be grouped into groups of 3. Exception only if number of students in class is not multiple of 3, or students drop.
* Lab assignments and HW will be individual.
* LAB will include lab lectures, Q&A on lab assignments, and grading of lab assignments.

**Catalog description:**

This course covers the nature and purposes of database systems and an introduction to data modeling: entity relationship model, relational model with relational algebra, relational calculus and SQL, integrity constraints, file organization and index files, and normalization.

**Credit hours:** 3 credits

**Required or elective:**

Required for CCE / ECE students

**Prerequisites:**

By course: EECE330

By topic: Data Structures and Algorithms

**Textbook(s) and/or required materials**

Textbook: Database Management System 3rd edition, by Ramakrishnan and Gehrke.

Additional material will be placed on reserve in the library and posted on Moodle.

**References:**

None

**Course Objectives**

|  |  |
| --- | --- |
| *The objectives of this course are to give students:* | *Correlates to program objectives* |
| Teach students how to develop SQL queries | 1,2,3 and 4 |
| Teach students how to design database schemas | 1,2,3 and 4 |
| Teach students how to develop applications that integrate with databases | 1,2,3 and 4 |
| Teach students storage mechanisms for databases | 1,2,3 and 4 |
| Introduce students to advanced topics in databases such as distributed databases, data warehousing, and decision support systems | 1,2,3 and 4 |

**Course Topics**

|  |  |  |
| --- | --- | --- |
| *No.* | *Subjects covered* | *50 min. lectures* |
| 1 | Overview of Database Systems (ch. 1) | 2 |
| 2 | Introduction to Database Design (ch. 2) | 3 |
| 3 | Relational Model: Integrity constraints, Logical DB design, and views (ch. 3) | 5 |
| 4 | Relational Algebra and Calculus (ch. 4) | 3 |
| 5 | SQL: Queries (ch. 5) | 5 |
| 6 | Database Application Development (ch. 6) | 4 |
| 7 | Internet Applications (ch. 7) | 5 |
| 8 | Overview of storage and indexing (ch.8) | 3 |
| 9 | Schema Refinement and Normal Forms (ch. 19) | 5 |
| 10 | Overview of Security and Authorization challenges (ch. 21) | 1 |
| 11 | Intro to Parallel and Distributed Databases (ch. 22) | 2 |
| 12 | Intro to Data Warehousing and Decision Support Systems (ch. 25) | 2 |

|  |  |  |
| --- | --- | --- |
| *No.* | *Subjects covered* | *75 min. lectures* |
| 1 | Overview of Database Systems (ch. 1) | 1 |
| 2 | Introduction to Database Design (ch. 2) | 3 |
| 3 | Relational Model: Integrity constraints, Logical DB design, and views (ch. 3) | 3 |
| 4 | Relational Algebra and Calculus (ch. 4) | 2 |
| 5 | SQL: Queries (ch. 5) | 4 |
| 6 | Database Application Development (ch. 6) | 3 |
| 7 | Internet Applications (ch. 7) | 3 |
| 8 | Overview of storage and indexing (ch.8) | 2 |
| 9 | Schema Refinement and Normal Forms (ch. 19) | 3 |
| 10 | Overview of Security and Authorization challenges (ch. 21) | 1 |
| 11 | Intro to Parallel and Distributed Databases (ch. 22) | 1 |
| 12 | Intro to Data Warehousing and Decision Support Systems (ch. 25) | 2 |

**Course Learning Outcomes**

|  |  |
| --- | --- |
| *At the end of the course, students:* | *Correlates to program outcomes\** |
| *H* | *M* | *L* |
| 1. Are able to describe the importance and use of databases
 | n | c | i,j |
| 1. Are able to explain the structure of a database
 | n | c | i,j |
| 1. Are able to develop ER diagrams as part of the DB design
 | e, n | c | i,j,k |
| 1. Are able to describe the different features of an ER Model
 | n | c | i,j |
| 1. Are able to understand the relational model in database
 | n | c | i,j,k |
| 1. Are able to understand key constraints and integrity constraints
 | n | c | i,j,k |
| 1. Are able to perform a logical DB design from an ER model
 | e | c | i,j,k |
| 1. Are able to understand the relationship between relational algebra and DB query
 | a,e,n | c | i,j |
| 1. Are able to describe different SQL query constructs (e.g. SELECT, GROUP BY, Nested queries...)
 | e | c | i,j |
| 1. Are able to develop basic programs in SQL
 | e | c | i,j,k |
| 1. Are able to implement simple query constraints in SQL
 | e | c | i,j,k |
| 1. Know how to establish a connection with a database using a high level programming language (e.g. C++)
 | e | c | i,j,k |
| 1. Know how to develop simple internet application with a database connection
 | e | c | i,j,k |
| 1. Understand how indexing can help with faster access to a database
 | e,n | c | i,j |

\* *H: High correlation, M: Medium correlation, L: Low correlation*

**Class/laboratory schedule**

1. Two 75-minute lectures per week; or three 50 mins lectures per week.
2. One 3-hours lab session per week
3. Use of computer lab is needed for working on the assignments.

**Resources of the course**

Textbook, database software (e.g. SQL and Oracle), compiler for high level language (e.g. for C++), IDE environment (e.g JavaBeans and Visual Studio), Moodle, and other resources as needed by lab and project.

**Computer usage**

Programming in high level language and SQL.

**Evaluation methods**

1. Participation (5%)
2. Lab Assignments/Drop Quizzes (5%)
3. Project (15%) Course project will be presented end of semester.
4. Quizzes (45%):
* Quiz I: Thursday March 22, 2012 - 5:00 – 7:00 pm
	+ Note that I will be in conference March 19-23.
	+ Problem sessions will be held during that week.
* Quiz II: Thursday April 26, 2012 - 5:00 – 7:00 pm
1. Final Exam (30%)

Note my potential absences for external conferences:

 Feb 27-March 1 – Will make up the two missed classes.

 March 19-23 – Problem sessions and quiz I will be held during my absence.

Class disruptions will negatively affect the grade, and students may be subject to disciplinary action.

Class and lab attendance are required. Students must be on time! Missing class will negatively affect grades.

Note that outstanding performance on course project is an indication of consistent effort, and may help boost the overall grade further.

Honor system and high ethics must be followed in all assignments and course engagements.

**Lab Expectations:**

* Lab attendance is mandatory. The instructor will explain or answer questions related to Database lab topics in conjunction with the assignments. LAB will generally be conducted as follows:
	+ ½ hr: Informal Lab lecture and Q&As
	+ 1 hr: Answering questions about the lab assignments
	+ 1hr: Grading lab assignments (on due dates)
* Lab assignments will be due on a weekly or bi-weekly depending on the complexity of the assignment.
* You have Programming Team project to be submitted at the end of the semester. You have to group yourselves into groups of three for this purpose. You all need to have a common base of knowledge. Expertise can vary. The names have to be submitted to the lab instructor during the first week.
* Suggested programming languages to use are C# or Java. It is advisable to seek lab instructor alignment on your choice. The language will be used to design windows and Web applications connected to databases (e.g. Microsoft SQL Server with C# and mysql for Java). For Java and C#, you will need to get familiar with at least one of the two IDE environments: Microsoft Visual Studio 2005 or Netbeans 6.5. **Each student** has to be familiar with one of these programming languages and its corresponding IDE.
* Each lab assignment has to be presented to the instructor and submitted to moodle. Every assignment has a due date. Late assignments will be accepted but penalized. Normally 10% per day.
* Lab assignment grading: The priority is to meet the requirements, but you should always look to exceed expectations.
* Project grading: The priority is to implement your project but you should also reflect good understanding of state-of-the art (ie need to conduct literature review).