Objectives of Lecture 1

Introduction

• Get a rough initial idea about the content of the course:
  – Lectures
  – Resources
  – Activities

• Mind refresher for Database Systems (CMPUT 291)
  (Students who are taking this course need to have knowledge about databases and expertise in structured programming, i.e., CMPUT 291 or equivalent is a course requirement)

Class and Office Hours

Classes for Section B2:
Mondays, Wednesdays and Fridays 13:00 to 13:50
CSC B10

Classes for Section B1:
Mondays, Wednesdays and Fridays 11:00 to 11:50
CSC B2 (Taught by Dr. Sander)

Office Hours:
Tuesdays and Wednesdays from 15:00 to 16:00
Office: ATH 352
By appointment:
E-mail: zaiane@cs.ualberta.ca Tel: 492 2860
Check appointment page on my web page.

Labs and TAs

Labs (CSC 219):

H01: Wednesday 14:00 to 16:50
H02: Thursday 08:00 to 10:50
H03: Thursday 11:00 to 13:50
H04: Thursday 14:00 to 16:50
H05: Friday 08:00 to 10:50

TAs:
Zhibin An (zhibin@cs.ualberta.ca)
Corrine Cheng (corrine@cs.ualberta.ca)
Chi-Hoon Lee (chihoon@cs.ualberta.ca)
Marianne Morris (marianne@cs.ualberta.ca)

Other Expert Assistants:
Alexandru Coman (acoman@cs.ualberta.ca)
Daniel Mallett (mallett@cs.ualberta.ca)
Resources

Course home page:
http://www.cs.ualberta.ca/~zaiane/courses/cmput391/
Contains links to course notes, detailed course calendar and other resources

Newsgroup
news://news.srv.ualberta.ca/ualberta.courses.cmput.391

Textbook:

Other recommended textbooks:

On-line Resources

http://www.cs.ualberta.ca/~zaiane/courses/cmput391/

• CMPUT 391 web page
• Course slides
• Web links
• Glossary
• Activity descriptions
• U-Chat
• Frequently asked questions
• Announcements

There will be no handouts distributed in class.

(Tentative, subject to changes)

Course Schedule

There are 14 weeks from January 5th to April 7th

• Lectures: cover the basic material for the course.
• Tutorials: complement the course and will be given during some lab hours. They contain information that is necessary to do the project.
• Assignments and Project: will be given later in the semester. You should work on them during lab hours (when there are no tutorials or lab exercises).
  – Implementation assignments will also be demonstrated during lab hours in the week following the assignment deadline.
  – The project demos will be demo’ed at the end of the semester.
• There are additional 5 lab exercises that will be marked by the TA.

Midterm (February 23rd)
Final Exam (April 21st for section B2  (April 14th for section B1)
Project Demos (last week of the semester)
**Course Calendar**

- **Introduction** Jan 05
- **Database Design & Normalization** Jan 07-09-12-14-16-19
- **Query Processing and Optimisation** Jan 21-23-26-28-30-Feb 02
- **Data Warehousing and OLAP** Feb 04-6
- **Transactions / ACID** Feb 09-11-13
- **Reading Week** Feb 23
- **Midterm** Feb 25-27-Mar 01-03-05
- **Querying XML** Mar 08-10
- **Information Retrieval** Mar 12
- **Data Mining** Mar 15-17-19
- **O-ODB & Spatial Data Management** Mar 22-24-26-29
- **Parallel and Distributed Databases** Mar 31- April 02-05
- **Project Demos** Mar 29 to Apr 02

---

**Lab Tutorials**

- Normalization
- Installation and use of Tomcat
- Java servlets
- Connectivity with databases (JDBC)
- Java Server Pages (JSP)
- Triggers with ORACLE
- Database Security
- Querying XML repositories
- Locking Isolation Levels with ORACLE

---

**Evaluation and Grading**

Your final grade will depend on the entire profile of the grades in your lecture section (bell-curve distribution) and a particular composite score does not guarantee a particular final grade. However, your composite score will be computed using the following weights:

- **Assignments** 20% (5 assignments, 4% each)
- **Lab Exercises** 5% (5 exercises, 1% each)
- **Mid-Term Examination** 20% (Feb 23rd)
- **Project** 25% (demo at end of semester)
- **Final Exam** 30% (April 21st)

- You have to pass the final exam in order to pass the course
- A+ will be given only for outstanding achievement.

---

**More About Evaluation**

**Re-examination.**
None, except as per regulation.

**Collaboration.**
Do Collaborate on assignments; do not merely copy.
Do not exchange machine-readable code (programs)

Plagiarism, cheating, misrepresentation of facts and participation in such offences are viewed as serious academic offences by the University and by the Campus Law Review Committee (CLRC) of General Faculties Council. Sanctions for such offences range from a reprimand to suspension or expulsion from the University.

**Plagiarism.**
Work submitted by a student that is the work of another student or any other person is considered plagiarism. Read Sections 26.1.4 and 26.1.5 of the University of Alberta calendar. Cases of plagiarism are immediately referred to the Dean of Science, who determines what course of action is appropriate.
Collaboration Policy

• Exams, Assignments and Lab Exercises are to be done individually.
• Even though you are allowed to form study groups and discuss assignments, each student must come up with his/her own solution by him/herself.
• Students may be asked at anytime to explain and/or justify their solutions and if they are clearly unable to do so then a zero mark may be assigned to the assignment in question and, if warranted, the case may be treated as a potential case of misconduct.

Plagiarism is a serious offence. It has been, and will continue to be, dealt with very seriously.

Assignments

- Exercises on Database Design and Normalization
- Algorithm Implementation on Database Design and Normalization
- Exercises on Query Processing and Optimization
- Exercises on Transaction Processing
- Combined Exercises

Lab Exercises

- Servlet
  - SELECT FROM WHERE
  - Servlet JDBC
  - Triggers
  - XMI
  - Locking: Isolation Levels

Course Project

- The objectives of the course project are to gain hands-on experience in design and implementation of Web-based information systems that use a database management system for storage and management of data.
Course Project

- Projects will be demonstrated in class at the end of the semester.

The idea is to build a web-based application from the ground up with technologies such as:
- ORACLE-8, Java, Servlets/JSP, JDBC, HTML forms, etc.

- The topic of the project is a management information system using Online Analytical Processing for a fictive “distributed” Electronics retail store.

Objectives for CMPUT 391

- To **understand** the fundamental concepts underlying database management systems:
  - database design methodology (normalization, etc.)
  - database management systems (query optimisation, concurrency, recovery, security, etc.)

- To **learn** about additional DB support for special data types such as XML documents and Spatial Data

- To **get acquainted** with data analysis issues such as data mining, data warehousing and information retrieval;

- To **gain** hands-on experience with database application systems and commercial database management systems.
  - developing an application system using ORACLE & web technology

What you studied in CMPUT 291

**ER Model**
**Relational Model**
**Relational algebra**
**Relational Calculus**
**SQL**
**Database Design/Normalization**
**Disk and File Structures**
**Indexing**
**Tree-structured indexes**
**Hashing**

The main objective for CMPUT 291 was:
Ensure that the student becomes a knowledgeable user of database management technology

- Understand how database management differs from file processing;
- Learn how to model data and build relational databases;
- Use query languages to access stored data.

You will be assumed to know this material. CMPUT 291 is a prerequisite for CMPUT 391

Basic Notions

**What is a database?**
- a large collection of data
- an integrated collection of data

**What is a database management system?**
- a software system that provides an efficient as well as convenient environment for accessing data in a database.

<table>
<thead>
<tr>
<th>Data: any information</th>
<th>Manufacturing</th>
<th>Product data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manufacturing</td>
<td>Student data</td>
<td>Courses</td>
</tr>
<tr>
<td>University</td>
<td>University</td>
<td>Student data, courses</td>
</tr>
<tr>
<td>Hospital</td>
<td>Hospital</td>
<td>Patient data, facilities</td>
</tr>
<tr>
<td>Bank</td>
<td>Bank</td>
<td>Account data</td>
</tr>
</tbody>
</table>

### Functionalities of DBMSs

- Specifying the database structure
  - data definition language
- Manipulation of the database
  - query processing and query optimisation
- Integrity enforcement
  - integrity constraints
- Concurrent control
  - multiple user environment
- Crash recovery
- Security and authorization

### Database Approach

**File management systems:**
- goal: efficiency
- problem: high speed ram vs. low speed disk access
- solution: complicated file structures

**Database management systems:**
- goal: efficiency as well as convenience
- problem: conflicts between efficiency and convenience
- solution: data independence supported by various database models
Database Management System

Users

- Naïve and casual database users
  - a person who knows nothing or not much about the database and accesses the data via forms and pre-built queries embedded in programs.

- Advanced users and application programmers
  - a person who knows about the structure of the database and about query languages, and can embed queries in programs.

- Expert users and application programmers
  - a person who knows how to write complex queries and advanced database programs based on knowledge of DBMS intricacies.

- Database administrator
  - a person who is responsible for the database design, scheme modification, user authorization, etc..

Database models:

- conceptual tools used to describe:
  - data
  - data relationships
  - data semantics
  - data constraints

Major database models

- E-R model: a logic foundation for conceptual database design
- network model: a set of records connected by links
- hierarchical model: a set of database trees
- relational model: a set of tables