UNIVERSITY OF NEBRASKA AT OMAHA
COURSE SYLLABUS/DESCRIPTION

Department and Course Number	CSCI 4850
Course Title	Database Management Systems
Course Coordinator	Zhengxin Chen
Total Credits	3
Date of Last Revision	May 19, 2009

1.0 Course Description

1.1 Overview of content and purpose of the course (Catalog description).
The course provides students basic knowledge on database management systems (DBMS). Main topics include: basics of data abstraction and data modeling, DBMS architecture, conceptual design and ER modeling, relational data modeling, relational algebra, SQL, integrity constraints and security, relational database design, storage media and indexing techniques, basics of query processing and transaction processing, and related topics.

1.2 For whom course is intended.
The course is intended for senior and graduate students in computer science (CS) who are interested in database concepts. The course is also for senior or graduate students majored in other disciplines so long as they have the basic knowledge of data structures and algorithms.

1.3 Prerequisites of the course (Courses).
CSCI 3320 (Data structures and algorithm analysis) or equivalent.

1.4 Prerequisites of the course (Topics).

Binary search trees
B+ tree (basic concepts)
Graphs
Hashing
Object-oriented concepts
Basic operators in set theory
Sorting algorithms

1.5 Unusual circumstances of the course.
None.

2.0 Objectives

List of performance objectives stated in terms of the student educational outcomes.

2.1 Understand the need for databases management systems

2.2 Understand the need for data abstraction and data modeling at various levels (including a basic understanding of XML);

2.3 Understand the basic database system architecture, including the basics of distributed and parallel databases

2.4 Be able to conduct conceptual data modeling using the entity-relationship (ER) approach, and compare it with an object-oriented approach.
2.5 Be able to write queries in formal languages such as relational algebra;
2.6 Design simple databases using such commercial tools such as Oracle
2.7 Write and execute queries in a commercial language (such as SQL) and perform other basic data manipulation activities (such as update);
2.8 Have a good understanding on the basics of integrity constraints and relational database design theory;
2.9 Understand the basic storage media and basic indexing techniques;

3.0 Content and Organization

3.1 Introduction (2.5 hours)
3.1.1 Data abstraction and data models
3.1.2 Databases and database users
3.1.3 Database System Architecture
3.2 Entity relationship (ER) modeling (6.0 hours)
3.2.1 Entity sets and relationship sets
3.2.2 Attributes
3.2.3 ER diagrams
3.2.4 Roles
3.2.5 Weak entity sets
3.2.6 Design issues
3.2.7 Using UML for ER modeling
3.2.8 Mapping ERD to tables
3.3 Relational databases (10.0 hours)
3.3.1 The concept of relation
3.3.2 Relational algebra operators
3.3.3 Basics of tuple relational calculus and domain relational calculus
3.3.4 Modification of relational databases
3.3.5 Views in relational databases
3.3.6 Oracle and SQL
3.3.7 SQL DDL and DML
3.3.8 Overview of other relational languages and tools
3.4 Integrity and security (3.0 hours)
3.4.1 Importance of integrity constraints
3.4.2 Different forms of integrity constraints
3.4.3 Database security
3.4.4 Authorization, encryption and authentication
3.5 Relational database design (9.0 hours)
3.5.1 Need for normalization
3.5.2 Functional dependencies
3.5.3 Basics of normalization theory based on functional dependencies
3.5.4 Normal forms
3.5.5 Other forms of dependencies and advanced normal forms
3.6 Advanced data modeling overview (6.0 hours)
3.6.1 Concepts for object oriented databases

3.6.2 Object-Oriented Databases

3.6.3 Object-Relational Databases

3.6.4 XML

3.7 Data storage and querying (3.0 hours)

3.7.1 Storage and file structure

3.7.2 Indexing and hashing

3.7.3 Query processing

3.7.4 Query optimization

3.8 Transaction management overview (3.0 hours)

3.8.1 The concept of transaction

3.8.2 ACID properties

3.8.3 Database states

3.8.4 Concurrency Control

3.8.5 Recovery System

3.9 Database system architecture overview (3.0 hours)

3.9.1 Overview

3.9.2 Distributed Databases

3.9.3 Parallel Databases

3.10 Summary of related topics and recent progress (1.5 hours)

4.0 Teaching Methodology

4.1 Methods to be used.
The basic teaching method will be the instructor's lectures.

4.2 Student role in the course.
The students will attend lectures, participate classroom activities, complete assignments and term projects, and pass all exams.

4.3 Contact hours.
Three (3) hours a week.

5.0 Evaluation

5.1 Type of student projects that will be the basis for evaluating student performance, specifying distinction between undergraduate and graduate, if applicable. For Laboratory projects, specify the number of weeks spent on each project).
Students are evaluated based on their understanding of materials covered in the semester. Basis of evaluation includes understanding of basic concepts of database management systems, as well as problem solving skills in regard to basics of database design and programming, including database design using entity-relationship approach and normalization theory, database programming using a theoretical language such as relational algebra, database programming skill using SQL, B++ tree operations and extendable hashing, and related techniques.

5.2 Basis for determining the final grade (Course requirements and grading standards) specifying distinction between undergraduate and graduate, if applicable.
The weights of grade will be allocated for the categories of participation, homework, projects and exams.

Exams (2 midterms):	30%
Classroom participation/exercises/discussion	10%
Written and SQL programming assignments:	30%
Final exam:	30 %

5.3 Grading scale and criteria.

97%-100%	A+
94%-96%	A
90%-93%	A-
87%-89%	B+
84%-86%	B
80%-83%	B-
77%-79%	C+
74%-76%	C
70%-73%	C-
67%-69%	D+
64%-66%	D
60%-63%	D-
Below 60%	F

6.0 Resource Material

6.1 Textbooks and/or other required readings used in course.

6.1.1 A. Silbershatz, H. F. Kort and S. Sudarshan (4th ed.) *Database System Concepts,* 2001.

6.1.2 G. Garcia-Molina, J. D. Ullman and J. Widom, *Database systems: The complete Book,* Prentice Hall, 2001.

6.1.3 R. Ramakrishnan and J. Gehrke, *Database Management Systems* (3rd ed.), McGraw-Hill, 1997.

6.1.4 Elmasri, R. and S. B. Navathe, *Fundamentals of Database Systems*, (3rd.), Benjamin/Cummings, 2000

6.2 Other suggested reading materials, if any.
None.

6.3 Other sources of information.

6.3.1 IEEE Transactions on Knowledge and Data Engineering

6.3.2 Data and Knowledge Engineering

6.3.3 Proceedings ACM SIGMOD Conference

6.3.4 Proceedings IEEE Data Engineering (ICDE)

6.3.5 Proceedings Very Large Databases Conference (VLDB)

6.4 Current bibliography of resource for student's information.
None.

7.0 Computer Science Accreditation Board (CSAB) Category Content (class time in hours)

CSAB Category	*Core*	*Advanced*
Data structures	15.0	
Computer organization and architecture		15.0
Algorithms and software design		
Concepts of programming languages	15.0	

8.0 Oral and Written Communications

Every student is required to submit at least __0__ written reports (not including exams, tests, quizzes, or commented programs) to typically ___0_ pages and to make __0__ oral presentations of typically _0__ minutes duration. Include only material that is graded for grammar, spelling, style, and so forth, as well as for technical content, completeness, and accuracy.

9.0 Social and Ethical Issues

Please list the topics that address the social and ethical implications of computing covered in all course sections. Estimate the class time spent on each topic. In what ways are the students in this course graded on their understanding of these topics (e.g. test questions, essays, oral presentations, and so forth?).

No coverage.

10.0 Theoretical content

Please list the types of theoretical material covered, and estimate the time devoted to such coverage.

Functional dependency and relational database design: 6 contact hours.

11.0 Problem analysis

There will be analysis on quizzes, written/programming assignments and exams. Analysis may be given during the lecturing or in the written form to be distributed in the class. Students are asked to review the analysis materials, and are responsible for these materials.

12.0 Solution design

Students will gain database design experiences using entity-relationship approach and normalization theory through classroom quizzes, written assignments and exams.

CHANGE HISTORY

Date	***Change***	***By whom***	***Comments***
10/30/2003	Initial ABET version	Chen	
06/13/2003	Cleanup	Wileman	
01/27/2009	Inserted mapping table (course objectives to program outcomes	Chen	
05/19/09	Corrected some typos	Chundi	

UNIVERSITY OF NEBRASKA AT OMAHA
Mapping of CS Program Outcomes vs. course objectives

Department and Course Number	CSCI 4850
Course Title	Database Management Systems
Course Coordinator	Zhengxin Chen
Total Credits	3
Date of Last Revision	Jan. 27, 2009

S – Strong relationship
X – Contributing relationship

Course objective	**CS Program Outcomes**										
	(a) knowledge of discipline	**(b) analyze problem, define**	**(c) design and implement solution**	**(d) function on a team**	**(e) ethical issues**	**(f) communicate effectively**	**(g) analyze impact of computing**	**(h) continued professional development**	**(i) Current techniques and tools**	**(j) apply foundations**	**(k) apply design and development**
1. Understand the need for databases management systems						**X**		**X**			
2. Understand the need for data abstraction and data modeling at various levels (including a basic understanding of XML)		**S**					**X**				
3. Understand the basic database system architecture, including the basics of distributed and parallel databases		**X**	**S**								
4. Be able to conduct conceptual data modeling using the entity-relationship (ER) approach, and compare it with an object-oriented approach.		**S**									**X**
5. Be able to write queries in formal languages such as relational algebra										**S**	
6. Design simple databases using such commercial tools such as Oracle				**X**		**X**					
7. Write and execute queries in a commercial language (such as SQL) and perform other basic data manipulation activities (such as update)									**X**		**S**
8. Have a good understanding on the basics of integrity constraints and relational database design theory									**X**	**S**	
9. Understand the basic storage media and basic indexing techniques		**X**	**S**						**X**	**X**	

CS Program Outcomes (2008)

(a) An ability to apply knowledge of computing and mathematics appropriate to the discipline;
(b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution;
(c) An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs;
(d) An ability to function effectively on teams to accomplish a common goal;
(e) An understanding of professional, ethical, legal, security, and social issues and responsibilities
(f) An ability to communicate effectively with a range of audiences
(g) An ability to analyze the local and global impact of computing on individuals, organizations and society
(h) Recognition of the need for, and an ability to engage in, continuing professional development
(i) An ability to use current techniques, skills, and tools necessary for computing practices
(j) An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices
(k) An ability to apply design and development principles in the construction of software systems of varying complexity.