CS 361 – Database Systems
Syllabus – Spring 2016

INSTRUCTOR: Tom Naps

OFFICE: Halsey 214, phone 424-1388

EMAIL: naps@uwosh.edu

OFFICE HOURS: MWF 9:30 - 10:30, Thursday 9:15 - 11:15

REFERENCES:

- Daily class handouts available by 4:00am MWF on D2L. Get a copy of them before class, organize them, take notes on and about them. They comprise a high-level, but not detailed, outline of what we will cover during class. Get a copy of them before class, organize them, take notes on and about them. Handouts that are not liberally saturated with your own explanatory notes will likely prove useless when you need them most.
- Daily class handouts available by 10:00pm the day before class on D2L.
- Review problems posted on D2L at the end of each class. Complete them by 4:00am on the day of the following class meeting and be ready to discuss the answers you have submitted (e.g., review problems posted Wednesday are due by 4:00am Friday).
- Optional but good references (all three have provided material for my lecture slides, review problems, exam problems):
  - *Database Management Systems – the "Cow Book"*, 3rd edition, Raghu Ramakrishnan and Johannes Gehrke
  - *Database Systems: The Complete Book (2nd Edition)*, Hector Garcia-Molina, Jeffrey D. Ullman and Jennifer Widom, Prentice Hall, 2008
  - Jennifer Widom's *free* Database MOOC at Stanford – *https://class.stanford.edu/courses/DB/2014/SelfPaced/about*. Some of the material there matches pretty well what we will cover under topics 1-6, 9, 13 below

Topic Coverage

1. Overview of database systems
2. Introduction to database design
3. The relational model
4. Relational algebra and calculus
5. Schema refinement and normal forms
6. SQL
7. Database application development
8. Storage and indexing
9. Transaction management
10. Concurrency control
11. Crash recovery
12. Query evaluation and optimization
13. Non-relational ("NOSQL") data models – XML, JSON, MongoDB

Learning Outcomes

Given our coverage of these topics, you will be expected to ...

- Fundamental concepts of database design
  - model customer requirements of a relational database with an Entity-Relational diagram (E-R diagram)
  - transform an E-R diagram to a database schema
  - write a query to a relational database in SQL
  - formulate a query to a relational database from the basic operators in relational algebra
  - design a database to provide the necessary information for an organization while minimizing redundancy and null entries.
- SQL
  - design and create a database using the Data Definition Language of a Database Management System
  - write queries to a relational database in an interactive mode
- Design of "good" relations, Schema Refinement, Concept of normalization and other theoretical issues
  - formulate the integrity constraints in the form of functional dependencies
  - eliminate extraneous attributes in a functional dependency
  - eliminate redundant functional dependencies
  - develop a cover from a set of functional dependencies
  - evaluate a proposed relational schema and determine whether it is in Third-Normal-Form (3NF) or Boyce-Codd-Normal-Form (BCNF)
  - implement a normalization program that checks whether a proposed relational schema is in 3NF or BCNF
  - decompose proposed relational schemas that are not in 3NF or BCNF into 3NF or BCNF
- Basic file organization and various file structure methods
  - determine the access time of records based on the file organization and file structure
  - specify the type of stable and non-stable storage in the design of a database management system.
  - analyze the requirements and select the design of an index (Hash, B+ tree)

  - organize data on disk to minimize disk accesses for various queries.

- Algorithms and implementation of large database systems
  - analyze the need of a database operator (scan, equality search, range search, insert, delete etc) and determine an appropriate and/or efficient algorithm (external sorting, hash, B+, clustered vs. unclustered, various join algorithms) in its implementation.
    Implement a Relational Operation (Example: Join)

- Transaction processing, concurrency issues, and recovery
  - identify and prevent deadlocks in concurrent database accesses
  - be able to describe the recovery process of databases
  - design the data structure and program of a database recovery mechanism
  - provide accurate, consistent, and efficient transactions within the context of concurrency issues and the possibility of various kinds of failures, such as a system crash.

Course Grading Policies

Your grade for the course will be based on the following weighted factors:

Factor	**Weight**
Class participation and preparation	15%
Individual Homework Assignments	22.5%
"Real-world" project (Done individually or by team of two in four phases – conceptual design, relational schema design and refinement, SQL database definition and implementation of essential queries, application development and testing)	22.5%
Exams (four – dates to be announced):	40%

To get the 40% contribution to your grade from the four exams, I will use the formula:

$$E = 0.05 \times E_{worst} + 0.15 \times E_{best} + 0.10 \times E_{other1} + 0.10 \times E_{other2}$$

where E_{worst} is your worst exam score and E_{best} is your best exam score. At the end of the term, your work in all of these areas will contribute to a numerical grade for the course based on a 100-point scale. Grade cutoff levels on this final scale are:

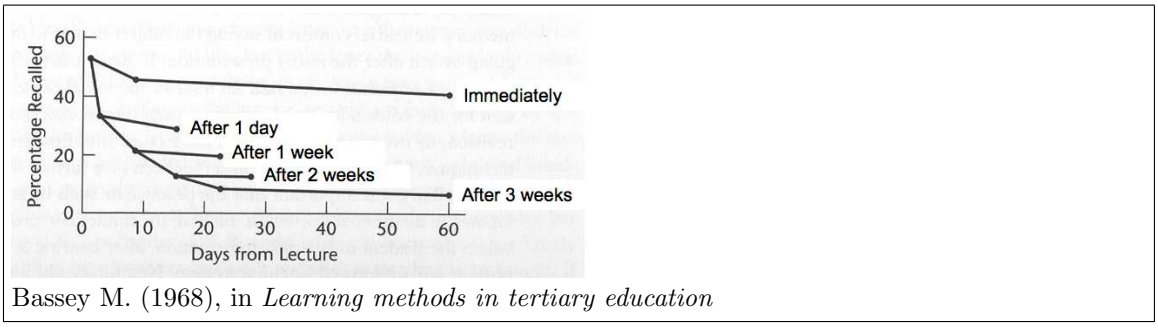
A $\geq$ 92	B $\geq$ 82	C $\geq$ 72	D $\geq$ 62
A- $\geq$ 90	B- $\geq$ 80	C- $\geq$ 70	D- $\geq$ 60
B+ $\geq$ 88	C+ $\geq$ 78	D+ $\geq$ 68	F $<$ 60

FAQ

Do I have to come to class? You are expected to arrive prepared to ALL the course sessions. Furthermore you are expected to participate in the classroom discussions and activities to the best of your abilities. This includes being ready to defend your answer to the review problems from the previous class (more on that later). It is difficult to envision a student missing and/or arriving unprepared to a number of the class sessions and still succeeding in the course.

How much time will this course take? Figure about three hours outside of class for each hour in class. That heuristic makes being a full-time student pretty much equivalent to holding a full-time job, so this is really good preparation for the real world that awaits you after graduation.

How can I best prepare for the exams? We've known what the following graph illustrates since 1968:



Bassey M. (1968), in *Learning methods in tertiary education*

More interesting evidence ...

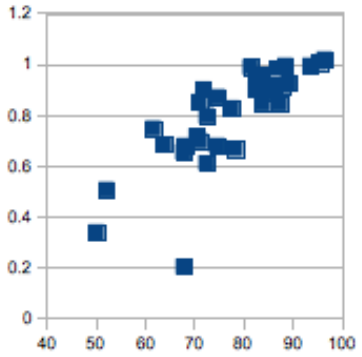
"Research has demonstrated that after a lecture, students recall 62% of the information. However, only 45% is recalled by students after 3-4 days and in 8 weeks only 24% of the information is recalled. If a quiz or exam was administered after the lecture, recall was doubled at the 8-week period. It is interesting that many faculty members appear to ignore the potential impact which quizzes and tests can have upon learning." – Bonwell C.C., Eison J.A.: Active Learning: Creating Excitement in the Classroom. Washington, DC: George Washington University, 1991.

Consequently at the end of most of our class periods you will find on D2L a small set of review problems covering what we discussed in class. The time to work on these review problems is immediately after the material is covered in class. You are also encouraged to discuss review problems with your classmates in a spirit of mutual help toward better understanding of how to solve them.

We will always discuss the review problems at the beginning of the class following their distribution. Your solutions to the review problems are due no later than 4:00am on the Monday, Wednesday, or Friday following the day of their distribution. (e.g., Review problems posted Wednesday are due by 4:00am Friday.) These solutions should be submitted using the link to a blank "quiz" form for the review problems that you will find on D2L.

If you have participated in class the day the review problem was distributed, have made a good faith effort to work on the review problem, and are "stuck" on it, I will be more than happy to help you with it if you come my office anytime within three days after you have received the review problem in class. After those three days (not counting weekends), *because you have made the choice to not learn effectively*, you are on your own in terms of grappling with these review problems.

Although the review problems only count 15% of your grade, the following correlation from a previous course between review-problem-percentage (on a 0 to 1.0 vertical scale) and overall percent in the course (on a horizontal scale of 0 to 100) is indicative of their true importance.



What if I'm late in submitting an assignment or one phase of the project for evaluation? Each assignment will carry with it a due date. If you are late in submitting it for evaluation, it will be accepted but will be penalized at the rate of 10% of point value the first day late, *an additional* 20% the second, *an additional* 30% the third ...

Is there any way I can carelessly lose points in the course? Yes ...

- Be late in submitting your work for evaluation on assignments.
- Don't participate in and prepare for the class.

What is this class participation/preparation stuff? How does it add up to 15% of my grade? ...

- Be sure to get those review problems done and submitted on time
- Exhibit your knowledge when called on to explain your answer to a review problem.
- Exhibit your knowledge when called on to respond to other questions in class

Is there any way I can get some bonus points? ...

- Do an outstanding job when called on to explain your answer to a review problem.

Can I get an extension on work that is due on a specified date? Only if you're ill enough to provide a signed note from the attending physician or have other reasons serious enough that the Dean of Students Office is willing to provide a written note justifying the extension.

If I miss a test, can I make it up? If you are unable to take a scheduled exam, it may be possible to take a make-up exam provided that you do BOTH of the following, which are then subject to my approval:

- Make arrangements prior to the scheduled exam (for last minute emergencies, telephone me at 424-1388 or leave a message at the computer science office, 424-2068). No after-the-fact notifications will be accepted ... *AND*

- Have a written medical excuse signed by the attending physician OR have a note of justification from the Dean of Students Office.

Only one make-up exam will be given. It will be a rigorous comprehensive exam given at an arranged time during the last week of the semester.

Can I consult with others on the individual assignments? Problems on the individual assignments will come in three flavors – (1) purely written answers, (2) SQL scripts, (3) Java programs exemplifying algorithms that lie at the core of a DBMS. Your work on the first two types of problems must be done without consulting any other students. For those problems in the third category that involve developing an algorithmic Java program, here is my policy: It is acceptable to consult another student for help in debugging such a program that you have authored yourself and that is not producing the result you expected. You should not consult another student before reaching that debugging stage.