Syllabus
CMSC 362: Theory of databases

Spring 2015

Time:	TR 11:00
Room:	Ruffner 354
Website:	`http://cs.longwood.edu/courses/cmsc362/`

A course covering the theory and practice of modern database design and implementation. Topics include relational and hierarchical database design, database query languages, update consistency, and distributed databases.

Professor:	Don Blaheta
Office:	Ruffner 337
Phone:	x2191
Email:	`blahetadp@blahedo.org` [1]
Office hours:	Mondays 3–4pm; Tuesdays 2–3:30pm; Wednesdays 4–5:30pm

Overview

Data management is an old problem, once solvable only by large bureaucracies and lots of filing cabinets; the advent of digital computers in the mid-20th century made it possible to store more and more information on ever-smaller storage media with increasing levels of automation. More importantly, computers let us quickly perform complex searches on that data and to correlate information in ways never before possible.

This course is broadly divided into two halves. First, we will build on your practical foundation in databases from CMSC 262 so that you'll be able to design and build a small but substantial web-oriented database application. Then, in the second half, we'll explore some of the more theoretical aspects of database analysis and integrate this with the application you'll be building.

[1] Or `blahetadp@longwood.edu` if you'd rather, but I prefer the off-campus one and check it more frequently.

Objectives

At the end of this course, the successful student will be able to:

1. understand, write, and process expressions in the relational algebra;
2. create, query, and manage tables in multiple SQL-based DB systems;
3. design E-R diagrams appropriate to a domain and translate them into implementation;
4. assess database performance and explain issues that will affect performance as a system scales; and
5. explain the relative merits of different kinds of DBMS under various circumstances.

Textbook and resources

The textbook is Jeffrey D. Ullman and Jennifer Widom, *A first course in database systems*, 3e (ISBN 978-0-13-600637-4). It was substantially revised both in order and content from the previous edition, so I don't recommend using the older version.

All our implementations will be done on the departmental Linux systems. If you do not have an active account on those systems, contact me ASAP to get that set up.

Content

Grading scale

I tend to grade hard on individual assignments, but compensate for this in the final grades. The grading scale will be approximately as follows:

A−	[85, 90)	A	[90, 100]	[2]	
B−	[70, 85)	B	[75, 80)	B+	[80, 85)
C−	[55, 60)	C	[60, 65)	C+	[65, 70)
D−	[40, 45)	D	[45, 50)	D+	[50, 55)

While there will be no "curve" in the statistical sense, I may slightly adjust the scale at the end of the term if it turns out some of the assignments were too difficult.

[2] Alas, no A+. Pity.

Graded assignments

I figure that I have about 8–10 hours of your time every week, including class time as well as reading assignments, homework, and the project. If you find you're spending more time than this, please do come discuss it with me, and we'll see what we can work out. The work you do for this course will be evaluated as follows:

Preparation and participation. I expect you to read (and think about!) at least a few pages before nearly every class. You need not perfectly understand the reading—that's what class is for—but you should be able to identify confusing parts and be ready to ask questions about them. I will sometimes give reading quizzes to confirm that you're doing this. Note that attendance is a prerequisite: if you aren't there, you may or may not be prepared but you definitely aren't participating. Preparation and participation are collectively worth 4% of the grade. — 4%

Homework. Most weeks, usually on Thursday, I will assign a short written homework at the end of class to be due at the beginning of the next. Each homework will proceed in two rounds: in response to your first handin, I'll give feedback (but no grade); after you have revised it, I'll assign a grade. Each problem will get 5, 3, or 0 points. The homeworks are group work: you can work with anyone in the class (or on your own if you prefer), and mark the names of the whole group at the top of a single handin. These will make up 10% of the grade. — 10% group

Exams. There will be two take-home exams, one in mid-February and one in mid-April. There will be no additional final exam (just the term paper and course project). Each exam is worth 15% of the grade. — 30% non-collaborative

Project. At the end of February, you will be assigned to a group and begin the course project, implementing a real database system start-to-finish. The final duedate for this will be during our final exam period (where you will demo it to the class and visitors), and there will be several checkpoints and milestones along the way. I'll give you more details as the time gets closer. The project will be worth 25% of the grade. — 25% group

Writing. This being a "writing-intensive" class, there will be three major writing assignments building toward a term paper; see attached rubric for more information about grading. These papers will be worth 31% of the final grade. Students must earn a grade of C− or better in the course in order to apply it toward their writing intensive course requirement. — 31% collaborative

Calendar (tentative)

Wk	T	R
	January	
1	**13** — Introduction, overview	**15** Ch. 1, §§2.1–2.3 Overview & review of basic SQL **Paper 1 out**
2	**20** * §2.4 Relational algebra	**22** §2.5 Constraints
3	**27** §§4.1–4.4 Review of entity-relationship modeling Design principles	**29** §§4.5–4.6 Mapping ER models to relational designs **Paper 1 due**
	February	
4	**3** §6.1 SQL specifics: Projection, selection, strings, patterns, dates, `NULL`, `UNKNOWN`, order	**5** §§6.2, 6.3.6–6.3.8 Products and joins in SQL
5	**10** §§5.1–5.2 Bags Extended relational algebra	**12** §6.4 Full-relation operations in SQL: `DISTINCT`, grouping, aggregation, `HAVING` **Exam 1 out**
6	**17** — Project overview **Exam 1 due**	**19** TBA Web front-end programming
7	**24** §§9.1–9.3.1, 9.5 Front-end cont'd 3-tier architecture Using SQL in other code	**26** — Project work day **Paper 2 out**

SPRING BREAK

* **20 January**: Deadline to add/drop classes (5pm)

Wk	T	R
	March	
8 *	**10** §§3.1–3.2 Functional dependencies	**12** §3.3 Decomposition Boyce-Codd Normal Form
9	**17** §3.5 Third Normal Form	**19** §§3.6–3.7 Multi-value dependencies and 4NF **Paper 2 due**
10	**24** §§7.1–7.3 Constraints	**26** §§7.4–7.5 Assertions and triggers
		April
11	**31** §§6.3.1–6.3.5 Subqueries	**2** §§8.1–8.2, 8.5 Views **Paper 3 out**
12	**7** §§8.3–8.4 Indexes	**9** §§6.5–6.6 Updates and transactions **Exam 2 out**
13	**14** — Project work day **Exam 2 due**	**16** §§11.1–11.3 Semi-structured data XML and DTDs
14	**21** §§4.7–4.8 UML	**23** §§5.3–5.4 Datalog **Paper 3 due**

Wednesday 29th
Project demo day
3:00–5:30pm

* **9 March**: Deadline to withdraw from a class (5pm)

Policies

Support

Don't hesitate to talk to me if you're having trouble with any part of the class. I can help—it's my job! I'm in my office a lot (not just during posted office hours). Feel free to come in and ask questions (or just to talk). If you can't catch me in my office, email is probably your best bet.

Honor code policy

Above all, I ask and expect that you will conduct yourself with honesty and integrity—and not to ignore the other ten points of the Honor Code, either. Take pride in what you are capable of, and have the humility to give credit where it is due.

The two main forms of academic dishonesty are "cheating" and "plagiarism". "Cheating" is getting help from someplace you shouldn't, and "plagiarism" is presenting someone else's idea as if it's your own. If you ever find yourself inclined towards either of these, know that there are always other, better options. Persevere! See my website[3] for some discussion and examples of how to steer clear of these problems, and feel free to come talk to me if you need help finding some of those other options (even if it's for another course).

Cheating or plagiarism (on any assignment) will normally receive a *minimum* penalty of a lowered *course* grade, ranging up to an F in the course. Cases will also be turned in to the Honor Board. But: I believe in your potential, and I hope that you will, or will grow to, observe this policy not simply to evade punishment but positively as a matter of character.

Accommodations

If you have any special need that I can accommodate, I'm happy to do so; come speak to me early in the term so we can set things up. If you have a documented disability, you should also contact Longwood's Office of Disability Resources (Graham Hall, x2391) to discuss some of the support the college can offer you. All such conversations are confidential.

[3] `http://cs.longwood.edu/~dblaheta/collab.html`

Attendance and late policy

Attendance is required, and assignments must be turned in on time. That said, if you have a good reason to miss class or hand something in late, I tend to be fairly liberal with extensions if you ask in advance. (Good reasons do include assignments due for other classes.) (And medical and family emergencies are exempted from the "in advance" part, of course. But get a note from a dean.)

Frequent absence will result in a lowered participation grade; habitual absence may in extreme cases result in a failing grade for the class. *Unexcused* late assignments will normally be given a zero.

Inclement weather policy

I don't plan to cancel class for weather unless the entire college shuts down. If you are commuting or are otherwise significantly affected by a weather event, use your own best judgement; and if you do miss class for this reason, contact me as soon as possible to make up missed work.

Early bird policy

Nobody's perfect, and on occasion an assignment gets written a little unclearly (or, once in a while, with an actual error in it). If you catch one and bring it to my attention early, so that I can issue a clarification or correction to the rest of the class, there'll be some extra credit in it for you.

Writing rubric

The department has a common rubric we use for writing assignments in writing-intensive courses, shown on the next page. A score on the scale 0–4 is assigned in each row, and then the row's multiplier is applied and all scores are added. Maximum 40 points; I scale the rubric score at ten points per grade level (so, 30 is a B, 23 is a C+, etc) before putting it in the gradebook. (Other professors use the same rubric but may scale it differently.)

Department rubric for writing assignments

Component	4 = Best	3	2	1	0 = Worst
Content Fundamentals (x2) Enough content Terminology Correctness	The student presents enough technical content to fully address the assignment, correctly using appropriate technical terms, and makes essentially no factual errors.	The student presents an adequate amount of material, deploys terminology reasonably well, and makes no significant errors of fact.	The student generally says true things.	The writing contains persistent misuse of terminology and/or significant errors of fact.	The writing contains a negligible amount of correct information.
Content Mastery (x3) Enough support Analysis Synthesis Originality Use of sources	Support is developed fully, from multiple angles; analysis is insightful; student demonstrates synthesis of ideas with clear and complete explanations; and writing is an appropriate mix of original work and (acknowledged) outside content that shows mastery of the material. (Each as appropriate to task)	The student has done more than recap or paraphrase other work, or make "obvious" statements, but the development is somewhat one-dimensional or uninspired.	The student has made significant progress toward supporting the purpose or thesis, but with insufficient analysis or explanation.	The student's arguments or support are handwavy and have little substance, are too minimal, or exhibit over-reliance on restatements of arguments from other sources.	There is no discernible support for the writing's purpose or thesis (if any).
Purpose / thesis (x1) Clear Strong Consistent	The purpose or thesis is on the assigned topic, strong, and clear from the introduction, and remains consistent through the conclusion.	The purpose or thesis is slightly off or not entirely clear, but the writing seems to be consistently on one path.	The purpose or thesis is unclear and/or seems to drift during the writing, but stays on a relevant topic.	The purpose or thesis is unclear or off-topic.	There is no discernible thesis or purpose to the writing.
Organization and flow (x2) Main points Transitions Sentence structure Fluidity	The student has structured the writing to make the arguments or main points are clear, and competently uses transitions and parallel and contrasting sentence structures to make the writing fluid and clear.	The student's writing is well-structured and mostly fluid but occasionally trips the reader with awkward transitions or sentence structures.	The student's writing has a reasonably clear overall structure.	The writing's structure is unclear, or it contains whole sections of choppy or awkward text that require extra effort from the reader to maintain focus.	The writing's structure and flow are so poor that the main points are not discernible.
Writing mechanics (x2) Grammar Spelling Punctuation Citations	The student uses correct grammar, makes essentially no errors of spelling or punctuation, and formats citations (if any) correctly.	The student makes no serious errors of grammar, spelling, punctuation, or citation, but there are a few minor problems.	The student's writing is basically coherent and understandable.	The student makes significant mechanical errors that create ambiguity or impede understanding.	The writing's low-level mechanics are so poor that the language is incoherent and difficult to understand.