CS 312, Sections 001 & 002, Winter 2009:

Algorithm Analysis

Instructor: Eric Ringger
Web: http://faculty.cs.byu.edu/~ringger/
Email: ringger AT cs DOT byu DOT edu
Phone: 801-422-7615
Office: 3368 TMCB
Office Hours: M 3-4pm; W 4-5pm; and by appointment

Lecture:
Section 001: MWF 10:00-10:50am, 3718 HBLL
Section 002: MWF 11:00-11:50am, 3718 HBLL

TAs: Paul Felt, Aaron Stewart, David Wilcox
Email: byu312ta AT gmail DOT com
Office Hours: M 2-3, 4-5; T 10-11, 3-4; W 1-2; Th 10-11, 3-4; F 1-2; and by appointment
Location: CS 312 TA Cubicle in TMCB Basement

Web-site: http://faculty.cs.byu.edu/~ringger/CS312/

Announcements: See the BYU Blackboard page for this course. Please check for announcements regularly.

Last updated: *2009 January 12*

Description and Objectives

Welcome to CS 312! This course will provide an introduction to the analysis of algorithms from both a theoretical and a practical perspective. Algorithms solve problems, and we will explore a wide variety of problems, some relatively abstract and some down-to-earth. Application domains include cryptography, geometry, 3-D graphics, image processing, bioinformatics, logistical planning, and route planning. As such, CS 312 provides an excellent mathematical foundation and will serve as a gateway to problem solving in other domains.

We will explore several families of algorithms, with the ability to solve problems of increasing complexity:

- Divide and Conquer
- Graph Algorithms
- Greedy Algorithms
- Dynamic Programming
- Real-valued Linear Programming
- Heuristic Search (Branch & Bound and A*)
- Probabilistic (Randomized)

The theoretical analysis of algorithms can answer questions that are difficult or impossible to answer by other means. However, one's trust in theoretical results should be tempered by a firm understanding of their limits. The most important strength and limitation of theoretical analysis is the generality one can achieve. This generality is not always useful when attempting to solve a small set of problems with a well-defined performance goal.

After completing CS 312, the student will have learned to successfully apply the following theoretical analysis techniques:

- Asymptotic analysis
- Probability theory in service of average case analysis
- Recurrence relations for the analysis of recursive algorithms

The practice and theory of algorithms covered in this class will serve you well as you move on to more ambitious subjects in Computer Science, and it will serve you well in your careers. This course is also required by many of the department's 400-level courses.

As a practical objective of the course, CS 312 students will use Visual Studio, the C# programming language, and the .NET framework. This gives CS 312 students solid exposure to a widely used development environment. We find that many CS graduates working in industry use Microsoft tools in their professions, and their experience in CS 312 will serve them well. That said, for several of the projects, we provide pre-constructed scaffolding to save you time and to allow you to focus on implementing the core algorithms and data structures for the project. Consequently, you will be insulated from some of the .NET framework. If you would prefer to implement from scratch, you may do so, but you must recognize that this will require more time on your part.

Web site

Class web site: http://faculty.cs.byu.edu/~ringger/CS312/
Class forum: Blackboard
Online grade book: Blackboard

The class web site contains all of the pointers you might need, including a link to the detailed, day-to-day schedule. Blackboard allows us to post announcements and pointers as they come up. Note that we also use Blackboard to share the keys for the homework assignments and for grades. The forum on Blackboard can be used by you to discuss questions about the assignments.

Please actively check Blackboard for announcements and pointers.

Textbook

The textbook for the course is *Algorithms* by Dasgupta, Papadimitriou, and Vazirani. We also use supplemental handouts, which are available via links from the course schedule. We hope you'll find these resources useful. If not, let us know how we can improve them.

Development Tools

Visual Studio is installed on the Windows machines in department labs, and you can get a free copy for your own machines if you are a BYU CS student through the MSDN Academic Alliance here:

Primary BYU MSDNAA page: http://csalliance.cs.byu.edu/
Information from Microsoft: http://msdn.microsoft.com/academic/

Furthermore, if you do not wish to go through MSDNAA, you can download Visual C# Express for free from Microsoft without any MSDNAA credentials. To the best of my knowledge, it will suffice for all that we do in this class.

http://www.microsoft.com/express/vcsharp/

You must use Visual Studio to develop in C# on the Windows platform to complete the projects.

To help you get started, here are some useful articles at MSDN:

- "Getting Started with Visual C#": http://msdn2.microsoft.com/en-us/library/a72418yk.aspx
- "C# Programming Guide": http://msdn.microsoft.com/en-us/library/67ef8sbd.aspx
- "Migrating to Visual C#": http://msdn2.microsoft.com/en-us/library/ms228394.aspx

Out-of-class Learning Experiences

Interviews with students suggest that what and how students do outside of class is the strongest predictor for engagement and learning in a given class (Light, Richard J., *Making the Most of College: Students Speak their Minds*, Harvard University Press, 2001.) The same study suggests that working in groups was also well-correlated with successful engagement and learning in class. So you are welcome to work in groups outside of class. However, you should each write and turn in your own work. Furthermore, if you work with others, you must acknowledge that you did so by listing your collaborators on your work.

Projects: There will be seven projects throughout the semester, some of them more challenging than others. For each project, you solve a concrete problem by implementing an algorithm in such a way as to meet a conservative performance requirement. A reasonable implementation will sail through the performance requirement. You will prepare a typed report electronically according to the guidelines for document for each project. You will answer questions posed in the project guidelines and usually report the results of an empirical analysis of your algorithm.

Submission: All project reports should be prepared in PDF format and submitted via the course submission web-page: http://nlp.cs.byu.edu/cs312/submit.php

Improvement: Furthermore, during the semester you will identify an opportunity to make a significant improvement to one of your projects. During the course you may see a better way to solve a problem from an early project. Solving an earlier project with a better algorithm is the central requirement of the semester-end Improvement report. Improvement also entails implementing a better solution according to some metric, including running time, memory consumption, or applicability to more problems. Simply tweaking your algorithm to give better performance is not sufficient; there must be a substantial change in the algorithms or data structures used in your solution to the problem at hand. In addition to making your solution better (but not in place of improving your algorithm), you may also include a nice visualization tool for observing the behavior of the algorithm as it runs. We include several improvement suggestions with each project; you may consider any of those as options. Ideas will also be proposed from time to time in class. You are not restricted to the ideas accompanying each project guidelines document or mentioned in class.

Proposal: You must submit a (maximum) one page typed proposal for your improvement by the proposal deadline, and your proposal must be approved by the instructor prior to proceeding. Improvements not preceded by an approved proposal will not be graded.

Report: You will satisfy the improvement requirement by writing a (maximum) 3 page report describing your improvement, your objective for the improvement, and an empirical analysis demonstrating that you accomplished your objective.

Submission: the improvement proposal and report should be prepared in PDF format and submitted via the course submission web-page: http://nlp.cs.byu.edu/cs312/submit.php

Homework: Regular homework assignments are posted on the schedule and announced in class. We will have homework nearly every day. Think of the homework as a way to solidify concepts learned during the lecture and from the required reading. Often the homework will help you prepare to succeed on your projects as well. A reasonable attempt at a homework

assignment will earn at least half credit. "Reasonable" means that you show your work and justify your answers. Unless you are having fun, do not spend more than two hours on a non-project homework assignment. Homework assignments are due on paper at the beginning of class on the due date specified on the schedule. Late homework is not accepted, not even later during class.

As is the case with the projects, you are encouraged to work together on the homework, but you should each write up and turn in your own homework and indicate with whom you worked.

Exams: There will be one mid-term exam and a final exam. The mid-term exam will take place in the testing center with a three hour time limit. You may bring one page of hand-written (by you only) notes into each exam. Your notes must be submitted with the exam. The mid-term exam will be scheduled to allow for flexibility. You are encouraged to study together in preparation for the exams.

The final exam will be held in class according to the University's calendar (see the course schedule). You may also bring one page of hand-written (by you only) notes into this exam. As per University policy, no exceptions will be granted to final exam time or location. Plan your travel accordingly.

Course Policies:

Grade weights: Your overall grade will consist of the following components:

Learning Experience	Final Grade Weight
Project #1	2.5%
Project #2	8%
Project #3	8%
Project #4	8%
Project #5	8%
Project #6	2.5%
Project #7	8%
Improvement	10%
Homework	15%
Mid-Term Exam	15%
Final Exam	15%

Grade scale: The following table shows the guaranteed lowest final grade for any given percentage (i.e., if you earn 93.0% of the possible points above, you will get a final grade no lower than an A). I reserve the right to adjust final grades upward in your favor at the end of the semester in order to reflect your diligence in the course and to account for inadequacies in my effort to communicate concepts to you.

Letter Grade	Minimum Percent
A	93.0
A-	90.0
B+	87.0
B	83.0
B-	80.0
C+	77.0
C	73.0
C-	70.0
D+	67.0
D	63.0
D-	60.0
E	0.0

The University policies on I (Incomplete) grades will be strictly followed.

Early work: There is no special early policy for homework assignments (an early homework assignment counts as much as an on-time one). For the projects, turning in a project report early (at least by 5pm on the date of the previous lecture) will result in an automatic bonus of 1% (absolute) toward the course grade. See the schedule for early submission deadlines (in blue). I encourage early submissions to help keep the course moving forward smoothly.

If part of a project is considered extra credit, then the required part of the project can be handed in early and separately from the extra credit part, thereby making it possible to get the early bonus on the required part. The extra credit part of the project would still be due by the normal due date & time, with late policies applying. There may be no extra credit available in a given semester.

Late work: Homework assignments are not accepted late, as they are meant to help you prepare for the lectures. You either did it or you didn't.

The project reports are due in PDF format by 5pm on the specified due date. For project reports, each student has a budget of 5 free late days (non-holiday weekdays) which can be spent on any project, except that no more than 2 may be used on any single project. This should allow for some flexibility in the schedule, given a student's other class constraints and unexpected life events. Once the free late day budget for an assignment is spent, each additional late day results in a penalty of 10% per day, until that assignment is graded and handed back by the instructor or TA (at which point no credit is possible). Each late day ends at 5pm.

Weekends and University holidays will not be counted as late days.

All work must be submitted by the last day of instruction for the semester. The University does not permit me to accept any work thereafter.

In this class, our early and late policies should allow you enough flexibility for most special circumstances during the semester. Additionally, ***special circumstances can be accommodated by prior arrangement***. If you anticipate you need some kind of accommodation to handle a

special circumstance for which our early and late policies will not suffice (e.g., for homework), talk to your instructor as soon as possible. In all but the most dire circumstances, further special circumstances will not be accommodated after the fact. For example, if you are planning to get married during the semester, arrange with your instructor in advance to make up missed work, extend deadlines etc. If you got married the third week of classes and missed a project, but did not make prior arrangements, then no accommodations will be made. If you were in a car accident and at the Emergency Room and could not make prior arrangements, then of course we can make accommodations. Needless to say I hope the latter problem doesn't happen.

Cooperation: As noted, you are encouraged to work together on your homework and project reports. If you work with someone else, simply indicate the person(s) by name on your work. You each need to write your own code and report. Students are welcome (and encouraged!) to discuss the general topics of the class, including details of specific algorithms or methods appearing in the lectures or readings. Use of the Blackboard forum for this purpose is strongly encouraged. Your improvement should be your own work, although you may certainly discuss possibilities with others.

Please consult the sections on Projects and Homework for additional policy on cooperation.

Honor: The Honor Code includes a statement of standards regarding academic honesty. Academic honesty includes writing your own programs, properly citing sources in reports and doing your own work on exams. Examples of academic dishonesty include sharing code for projects with other students, turning in someone else's writing as your own report and cheating on an exam. The first violation of academic honesty standards will result in your course grade being lowered 1 grade level, and you will be required to either redo the work or receive a 0 on the assignment. The second violation will result in failing the class. All violations of academic honesty are documented and reported to the Honor Code office.

Harassment: Harassment of any kind is inappropriate at BYU. Specifically, BYU's policy against sexual harassment extends not only to employees of the university but to students as well. If you encounter sexual harassment, gender-based discrimination, or other inappropriate behavior, please talk to your professor, contact the Equal Employment Office at 422-5895 or 367-5689, or contact the Honor Code Office at 422-2847.

Disabilities: BYU is committed to providing reasonable accommodation to qualified persons with disabilities. If you have any disability that may adversely affect your success in this course, please contact the University Accessibility Center at 422-2767. Services deemed appropriate will be coordinated with the student and instructor by that office.