Course Syllabus
CS375: Design & Analysis of Algorithms
Spring 2014

Instructor and Teaching Assistants

Instructor: Lei Yu
Office: Q5, Engineering Building
Lecture hours: A0: 1:10pm-2:10pm (MWF); B1: 2:20pm-3:20pm (MWF)
Office hours: 3:30pm – 4:30pm, MW, or by appointment
Email: lyu@cs.binghamton.edu
Webpage: http://www.cs.binghamton.edu/~lyu/index.html

Teaching assistant: Yi Zhang
Office: TBD, Engineering Building
Office hours: 12:00pm – 1:00pm, TTH, or by appointment
Lab section: A50, room: SW325, time: 1:15pm-2:40pm, TH
Email: "Yi Zhang" <yzhang25@binghamton.edu>

Teaching assistant: Yunli Tang
Office: TBD, Engineering Building
Office hours: 11:00am – 12:00pm, MW, or by appointment
Lab section: B51, room: AA G023, time: 2:50pm-4.15pm, TH
Email: "Yunli Tang" <ytang4@binghamton.edu>

Course Description

Analysis of common algorithms for processing strings, trees, graphs and networks. Comparison of sorting and searching algorithms. Algorithm design strategies: divide and conquer, dynamic, greedy, back tracking, branch and bound. Introduction to NP-completeness. Required activity includes student presentations. Prerequisites: CS 240 and MATH 314.

Prerequisites

CS 240: Data Structures and Algorithms
Math 314: Discrete Mathematics

CS375 assumes students have mastered:

1. A high level language such as Java, C or C++.
2. Basic data structures such as arrays, linked lists, trees, heaps, and graphs.
3. Basic mathematical concepts such as:
    a. Logarithms and exponents
    b. Arithmetic and geometric series
    c. Combinations and permutations
    d. Limits and derivatives
    e. Proof techniques such as induction, direct proof, proof by contradiction, etc.

Textbook

- T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein. *Introduction to Algorithms*. 3rd edition, The MIT Press, 2009.

Reference Books

- R. Neapolitan and K. Naimipour. *Foundations of Algorithms*. 4th edition, Jones and Bartlett Publishers, 2011.
- Kleinberg and Tardos. *Algorithm Design*. Addison Wesley, 2005.
- A. V. Aho, J. E. Hopcroft, and J. D. Ullman. *Data Structures and Algorithms*. Addison-Wesley, 1984.
- S. Baase. *Computer Algorithms- Introduction to Design and Analysis*. 3rd edition, Addison-Wesley, 2000.
- G. Brassard and P. Bratley. *Fundamentals of Algorithmics*. Prentice Hall, 1996.
- E. Horowitz, S. Sahni and B. Rajasekaran. *Computer Algorithms C++*. Computer Science Press, 1996.
- R. Neapolitan and K. Naimipour. *Foundations of Algorithms Using C++ Pseudo Code*. 3rd edition, Jones and Bartlett Publishers, 2004.
- R. Neapolitan and K. Naimipour. *Foundations of Algorithms using Java Pseudo Code*. 3rd edition, Jones and Bartlett Publishers, 2004.
- R. Sedgewick. *Algorithms in Java, Part 1-4*. 3rd edition Addison Wesley, 2003.
- R. Sedgewick. *Algorithms in Java, Part 5*. 3rd edition Addison Wesley, 2004.
- R. Sedgewick. *Algorithms in C++, Part 1-4*. 3rd edition Addison Wesley, 2002.
- R. Sedgewick. *Algorithms in C++, Part 5*. 3rd edition Addison Wesley, 2004.
- R. Sedgewick and P. Flajolet. *An Introduction to the Analysis of Algorithms*. Addison Wesley, 1996.
- S. S. Skiena. *The Algorithm Design Manual*. Springer-Verlag 1998.

Main Topics

No.	Topic	Cormen Textbook Chapters
1	Time complexity, insertion sort and merge sort, asymptotic growth functions	Chapters 1, 2, 3
2	Solving recurrences, divide and conquer algorithms	Chapters 4
3	Heaps, heapsort, binary search trees	Chapters 6, 12
4	Sorts	Chapters 7, 8
5	Backtracking	
6	Hash functions	Chapter 11
7	Dynamic Programming	Chapter 15
8	Greedy method	Chapter 16
9	Amortized Analysis	Chapter 17
10	Graphs	Chapter 22
11	NP-Completeness	Chapter 34

Grades

1. Grading policy

- Two exams — 40% (Midterm 15%, Final 25%)
- 7-8 short quizzes — 10% (2% each, after dropping 2-3 lowest grades)
- Two programming Assignments — 20% (10% each)
- Four theoretical assignments — 20% (5% each)
- Project and presentation — 10%

Please note that no makeup quiz or exam will be given.

Late assignment submission may be accepted, but late penalty (20% per day) will apply.
No submission late for more than 2 days will be accepted.

2. Grading disputes

In this course, we commonly give partial credit to partially correct answers. Should you dispute a partial credit, please be aware that we will not re-grade a single question in a homework, quiz or exam. ALL partial credits of the work will be re-examined. The new grade may be higher, lower, or stay the same. This new grade will not be changed.

3. Missing grades

Your grades will be posted on Blackboard. Please check your status on blackboard periodically and make sure that there are no missing grades or errors. A missing grade at the end of the semester will indicate that the work has NOT been done.

4. Extra credit

Homework assignments may contain extra credit questions that may add up to 10% to your overall grade (enabling an overall grade higher that 100%). These problems are usually more difficult to solve. Students striving for a grade of A are encouraged to work on these problems.

5. Quizzes

Some of the quizzes are pop up quizzes, while others are pre-announced with a scheduled date.

6. Exams

Students MUST take all exams on the scheduled date and time. There will NOT be makeup exams.

7. Academic honesty expectations

Please review the academic honesty document and make sure that you understand it! The link is: http://www2.binghamton.edu/watson/advising/pdfs/honesty-policy.pdf

Cheating and copying will NOT be tolerated.

- Each programming or theoretical assignment should start with the following statement:

  "I have done this assignment completely on my own. I have not copied it, nor have I given my solution to anyone else. I understand that if I am involved in plagiarism I will have to sign an official form that I have cheated and that this form will be stored in my student's records. I also understand that I will receive a grade of **0 for all programming and theoretical assignments** so that my maximum grade will be 60%."

- Each exam and quiz will have a first page with the following statement:

  "I understand that if I am caught copying, or talking during the exam/quiz I will have to sign an official form that I have cheated and that this form will be stored in my student's records. I also understand that I will receive a grade of 0 for the exam/quiz"

Your homework assignment, quiz or exam will not be graded unless the statement above is followed by your signature.

8. Reading Assignments and review questions

You will be given a reading assignment for each unit. It may also include material that will not be covered in class but will be included in homework assignments. For some units you will also be provided with some "review questions".

Reading the text in this course is time consuming. To understand the material you may need to read the material more than once. After you read the material try to answer the review questions.

9. Homework assignments

- Much of what one learns in this course comes from solving problems.
- Start working on the homework early.
- Most questions require both knowledge of the material and problem solving ability.
- If you don't know how to solve a problem, don't give up.
- Make sure that you understand the questions.
- See if you can solve a problem for some simple cases, and then try to find a general solution.
- Try again on the next day.

All solutions of theoretical assignments must be typed (no handwritten solutions) and submitted in hard copy.

Advance electronic submissionto the TA is acceptable if the student is expected to miss the class on the due date.

Please make an effort to make your programs easy to understand and grade. Grading all assignments in this course is very time consuming! All programming assignments should have:

1. For the program (in a README file):
   - Step-by-step instructions on how to compile and run the program.
   - The data structures used by the program and how they are implemented.
   - Some analysis or discussion of its computation time.
   - The classes used and their interaction.

2. For each class in your program
   - An explanation of the purpose of the class, and the methods it includes.
3. For each method (function)
   - A description of the purpose of each function and an explanation of how it works.
   - A description of the purpose, and the assumptions made about each parameter of a function.
   - A comment for every variable declaration.

Program grades

- You will get a grade of at most 10% if your code does not compile.
- If the program compiles but it has runtime errors or bugs, your grade will be based on the severity of the bugs.
- Please note that the TA or instructor will not look at or debug your code before submission.

- You may find it helpful to use tools such as gdb, valgrind, and IDE debuggers. These will help you figure out why your program does not work properly and save you a lot of time. Google is a good standby when you get errors you have not seen before.

10. Programming Language

Programs must be written in Java, C++, or C. Make sure a program compiles on harvey.cc.binghamton.edu and runs correctly.

Class and Labs Attendance

Attendance is required. Some of the quizzes are pop quizzes. If you regularly miss classess or labs, your overall quiz score will be negatively affected.

Computers and Other Electronic Devices

You are not allowed to use your laptop/notebook/tablet computers during class except during a project presentation. Cell phones must be turned off or in vibrate alert mode during class.

Office Hours

We encourage you to take advantage of the office hours.