CS 535 — Analysis of Algorithms
Syllabus version 1.1
Gruia Calinescu
Department of Computer Science
Illinois Institute of Technology
Spring 2019

1 Course Summary
Welcome to CS 535 Analysis of Algorithms. This is a theoretical computer science graduate course. We will learn to design more efficient algorithms by being able to rigorously analyze their time and space requirements. Some material overlaps with CS 430 (Introduction to Algorithms), but our treatment will be more abstract and mathematically correct.

2 Textbook
The required textbook is Introduction to Algorithms (third edition) by Cormen, Leiserson, Rivest, and Stein, MIT Press, 2009. ISBN-10: 0262033844 — ISBN-13: 978-0262033848 It should be available at the bookstore, and the library has a copy on reserve (available only inside the building). Other editions of the book will be fine for learning, but there may be differences in the assigned exercises and problems.


3 Prerequisites
CS 430 is listed as prerequisite. In particular, the following topics are assumed and can appear on homeworks and exams:

1. Mathematical Background (Appendices A, B, C.1, D)

2. Pseudocode, Notations, and Mergesort (Chapters 1 and 2)

3. Growth of Functions (Chapter 3) and Recurrence including the master method (Chapter 4 except 4.6)
4. Heap, Heapsort (Chapter 6) and Quicksort (Chapter 7 except 7.4)
5. Elementary Data Structures (stacks, queues, linked lists, trees - Chapter 10)
6. Binary Search Trees (Chapter 12) and some balanced version of search trees
7. Greedy Algorithms (Chapter 16)
8. Dynamic Programming (Chapter 15)
9. Graph Algorithms: BFS, DFS (Chapter 22 except 22.5), Minimum Spanning Trees (Chapter 23), Shortest Paths: Dijkstra’s algorithm (Subchapter 24.3) and the Floyd-Warshall algorithm (Subchapter 25.2)

Familiarity (or a desire and ability to learn) mathematical proofs is also necessary.

4 Getting Help

Office hours are Tuesday 3:00 - 4:00 (including immediately after the class) and Thursday 2:00-3:00 in room SB 228D, or by appointment. For an appointment send e-mail to calinescu@iit.edu. You can also call me at 312-567-5273. Please spend a little time trying to understand yourself a homework problem before asking for help.

The handouts (including this syllabus and homeworks) will be available at http://www.cs.iit.edu/~cs535. Partial homework solutions will be posted on blackboard.

The TA for this class is Xiaolang Wang. Xiaolang (xwang122@hawk.iit.edu) has office hours Monday and Wednesday 1-2, in SB 004 (basement of Stuart Building), phone 312-567-5149.

Students are expected to check email every week day of the semester. Clarifications on assignments or other important announcements might be sent by email - and will also be posted on the web page.

5 Grading

The grading allocation is given below.

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homework</td>
<td>45%</td>
</tr>
<tr>
<td>Midterm</td>
<td>18%</td>
</tr>
<tr>
<td>Final exam</td>
<td>32%</td>
</tr>
<tr>
<td>Class participation</td>
<td>5%</td>
</tr>
</tbody>
</table>

The midterm will be on March 14, in class. The final exam will be held as scheduled by the Registrar’s office, during the week May 6-11. Makeup exams will be only for emergency situations.

Six homeworks will be assigned. You can discuss the problems with each other, but must write the solutions individually. Seek help from me or the TA if you are having difficulties with the homework. Except for extraordinary circumstances, homeworks will be accepted at latest one week late. The penalty for late submission is 10% for one lecture late and 20% for one week late.
The midterms and the final exam are closed books, closed notes, no electronic devices - although I will likely provide some of my handouts to you. At least one week in advance, I will hand in and post an additional list of candidate problems for each exam. The midterm may contain variants (same solution ideas) of the problems from the first three homeworks and this additional list. The final will follow the same pattern (with homeworks 4-6, and another additional list of candidate problems). The student in the PhD section will get one extra assignment, to replace class participation.

The final grades will be assigned by comparison with the students who took this class in six previous semesters. There were in total 88 A’s, 164 B’s, 14 C’s, and one E. As a guideline, about 78% will be needed for an A, and 58% for a B (but I might modify these thresholds to ensure fairness).

Standard departmental policy regarding academic (dis)honesty applies. This includes http://www.iit.edu/student_affairs/handbook/information_and_regulations/code_of_academic_honesty.shtml

In particular, homework solutions copied from the Internet are not allowed. If I have evidence that the work submitted is not your own, I will assign a score of zero on that particular assignment at the first occurrence. It may also be reported to academichonesty@iit.edu. The second occurrence will be reported and an appropriate sanction will be applied after consultation with the office of the Associate Provost for Academic Affairs. Whenever in doubt, ask first if some action is allowed or not. Moreover, the students must submit the signed College of Science Academic Integrity Pledge together with the first homework.

For the students in the TV and Internet sections, the weight of the homeworks will increase to 50% and there will be no class participation grade. Class participation is a subjective quantity, based mainly on how I feel you helped with comments and questions during the class. Just showing up gives you half of the points, but talking while I talk can decrease your score to zero.

6 Topics to be covered

1. Lower Bounds for sorting (Subchapters 8.1, 8.2), Medians (Subchapter 9.3)
2. Splay Trees (not in the textbook)
3. Advanced Data Structures (Chapters 17, 19, 21)
4. Graph Algorithms (the parts of chapters 22, 23, 24, 25, 26 not listed as prerequisite): Strongly Connected Components, Shortest Paths with negative weights, Network Flows
5. NP-Completeness (Chapter 34) and (if time permits) Approximation Algorithms (Chapter 35)
6. possibly Dynamic Programming (Chapter 15, even it is a prerequisite)
7. If time permits: Pattern Matching (Chapter 32, KMP – 32.4) and/or Geometric Algorithms (Chapter 33)