CSE530 Algorithms & Complexity
Lecture 0: Course Information

Antoine Vigneron
antoine@unist.ac.kr

Ulsan National Institute of Science and Technology

February 23, 2018

Introduction

- This course gives basic introduction to algorithms and complexity. The topics covered are: review of asymptotic notations, elementary data structures and graph algorithms, dynamic programming, maximum flow, linear programming, Turing machine formalism, the classes **P** and **NP**, **NP**-completeness and reduction, and probabilistic algorithms

- Instructor (1st half):
  - Antoine Vigneron (앙톤 비뉴롱)
  - `antoine@unist.ac.kr`

- Instructor (2nd half):
  - Aaram Yun (윤아람)
  - `aaramyun@unist.ac.kr`

Instructor

- Antoine Vigneron (앙톤 비뉴롱)
  - Vigneron (French): Winemaker
- Short CV:
  - MSc École Polytechnique (Paris), PhD HKUST (Hong Kong)
  - Previously worked at NUS (Singapore), INRA (France), KAUST (Saudi Arabia)
- Research:
  - Geometric Algorithms Lab EB2 710-2.
  - Research area: Algorithms. More precisely, geometric algorithms.
  - Mostly theoretical.

Course Structure

- First half of the semester on **Algorithms**
  - Taught by Antoine Vigneron
  - Design and analysis of worst-case efficient algorithms
- Second half on **Computational Complexity**
  - Taught by Aaram Yun
  - Focus: Classifying problems as hard or tractable

1st Half: Algorithms

Schedule

Week 1:	Review of algorithms analysis
Week 2:	Dynamic programming
Week 3:	Review of graph algorithms and data structures
Week 4–5:	Maximum flow
Week 6–7:	Linear programming

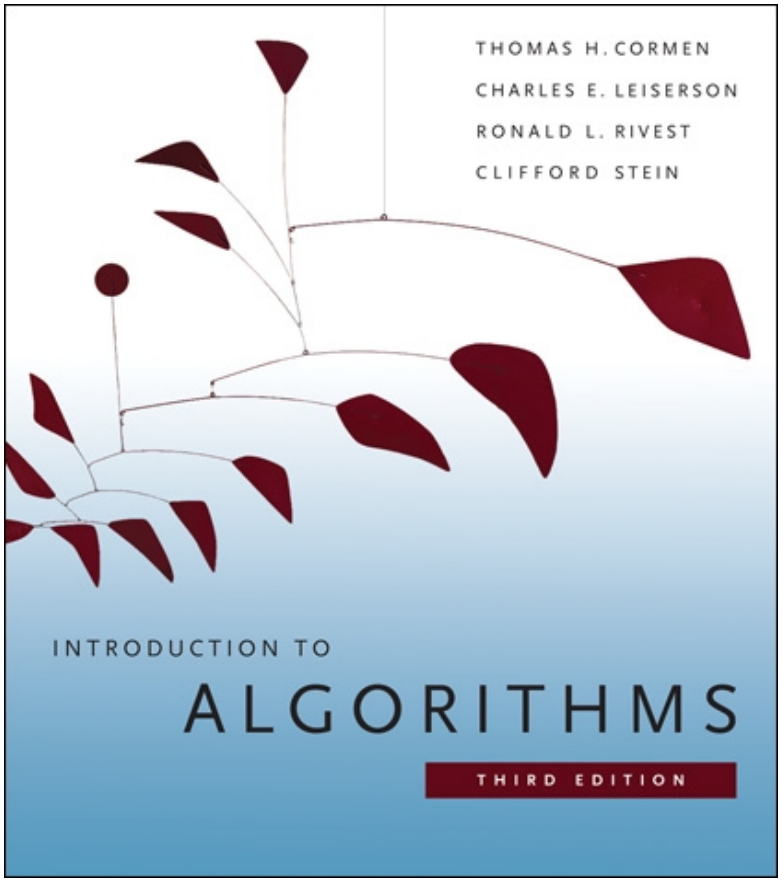
- The topics from Week 1 and 3 are usually covered in undergraduate courses. I will review them quickly as they are important, and necessary to understand the rest of the course.
- Dynamic programming (Week 2) is also often covered in undergraduate courses, but I will present algorithms that you probably don't know.
- Maximum flow and linear programming (Week 4–7) are typical topics for a postgraduate algorithms course.

1st Half: Algorithms

- Reason for choosing these topics: Dynamic programming, maximum flow, and linear programming are the most common techniques that give polynomial-time algorithms for non-trivial problems.
- So it is important for a researcher in computer science to master these techniques: It often allows you to find an efficient algorithm for computational problems that arise in your research work, when such an algorithm exists.
- The second half of the course (on complexity) addresses the related question: Is there an efficient algorithm for solving this problem? Or is it known to be hard?

Reference Book (1st half)

- Introduction to Algorithms by Cormen, Leiserson, Rivest and Stein.
- Available online from the UNIST library website.



- Very long textbook.
- Most of it will not be covered.
- But I will follows a few chapters from this book.

Grading

- 10% attendance
- 20% homework
- 35% midterm
- 35% final exam
- There will be two assignments during the first half of the semester
- I will also give (non-graded) exercises with solutions
- Closed-book exams, pen and paper

Grading: Attendance

- Reminder: In UNIST, attendance is checked. More than 8 absences yield an F grade.
- You will also get an attendance grade in this course that depends on how many times you are absend or late.
- Please try to use the electronic attendance system.
- If you fail to check attendance, tell me at the end of the lecture and I will register you by hand.
- If you do not attend, I need easily verifiable evidence. Example: a Medical certificate (with the date), a conference badge.
- Example of non accepted evidence: Course notes, or the fact that your assignment was returned on that day.
- I will take a picture of the classroom in each lecture in order to determine whether you were present in case of a complaint.

Academic Integrity

- Regarding assignments:
  - You should not *look* at a solution from another student.
  - You should not *show* your solution to another student.
  - But you can discuss the general approach to solve a problem with another student. In this case, mention it in your assignment.
  - Similarly, you should not *copy* your solution from the internet, or another source.
  - You can look at websites on topics related to the assignment, but you are not allowed to copy your answers *in whole or in part*. If you used a website, mention it in your assignment.
- Regarding exams/quiz:
  - You should not *look* at a solution from another student.
  - You should not *show* your solution to another student.
  - Exams are *closed book*. You should not use any document during the exam such as slides, class notes, books, or any written material other than the exam paper.