

# Who should take this course?

Anyone who is either

- a computer science/engineering student
- interested in getting to know **the most fundamental area in CS**

or

- forced to take it because it's required.

Pre-req.

- Advanced Data Structure (CSE 250 or equivalent)
- Some formal calculus
- A course which requires formal proofs (Discrete Math)
- If you don't have them, talk to me.

# Course Material

## Required textbook

**Introduction to Algorithms (3e)**, by Cormen, Leiserson, Rivest, and Stein.

## Online material

<http://www.cse.buffalo.edu/jinhui/courses/cse531/Classnotes>

## Important Notes:

- The classnotes are fairly extensive. You might think that it is enough to learn course material by studying the notes only. However, more details are discussed in class, and these details are equally important.
- The classnotes are numbered as: note00, note01, etc.
  - When print the notes, please use the “multiple pages per sheet” function provided by Acrobat to reduce the number of pages

# Workload

- Heavy, start early!
- Approximately 30 pages of **dense** reading per week
- 5 -6 written homework assignments
- 1 midterm
- 1 final exam

# Grading Policy

- Class Attendance 5%
- Homework total: 25%
- Midterm: 20%
- Final: 50%

## Note:

- Assignments are due at the beginning of the lecture on the due date
- No late assignments will be accepted, except for medical reason (with doctor's note) or family emergency (with documentation).
- Any grading dispute must be **raised within 1 week after the homework/exam are returned**. If you don't pick them up on time, you forfeit your right to question the grading.

## Absolutely no tolerance on plagiarism/cheating

- Do assignments individually on your own. Do not discuss with classmates
- 0 on a particular assignment/exam for the first violation
- Fail the course on the second violation **plus** report to the department
- Consult the University Code of Conduct for details

# How to do well in this class?

- Ask questions in class
- Attend lectures and recitations
- Do homework/reading early
- At least skim through reading assignments before lectures
- Print out lecture notes before attending lectures

We, the TAs and I, are here to help. Don't hesitate to ask questions. Suggestions are always welcome.

# The material you should already know

- Part I Chapters 1 - 4
  - How to analyze growth rate of runtime functions
  - During the first two weeks, we will have a quick review on this, and some material not covered in CSE 250.
- Part II, Chapters 6 - 8
  - Sorting algorithms
  - Lower bound theorem on sorting problem
- Part III, Chapters 10 -13
  - Basic data structures
  - Chapter 13 is on **red-black trees**. If you know the the concept of **balanced binary search trees**, and know at least one such trees (example: AVL tree), it is fine.

# The material you should already know

- Part IV Algorithm design techniques
  - Divide and Conquer
  - Greedy
  - Dynamic programming
  - You should know what they are, and know a few examples of each
  - We will discuss these in more details