CS4234
Optimiz(s)ation Algorithms

S1 AY23/24

Welcome!

https://www.wikihow.com/Address-a-Professor

Before Tue, 13 June 2023, 3pm

Title	First Name	Last (Sur) Name
Mr ⊗ (I spent 5 years for my PhD!)	Steven	Halim
Dr √		
Associate Professor ⊗ (not yet)		
Professor ⊗ (not yet)		
Prof √ (general version, "ok-ish")		

After Tue, 13 June 2023, 3pm

Title	First Name	Last (Sur) Name
Mr ⊗ (I spent 5 years for my PhD!)	Steven	Halim
Dr √ (but why just use this?)		
Associate Professor √ (but too long)		
Professor √ (my *next* rank actually)		
Prof √ (general version, best now)		

Summary: The best version is **Prof Halim**

How to be my full disciple (*typical* route)

CS2040S in Y1/Y2 Sem1

Too late if you do this on Y3/Y4
You will avoid me if you do Sem2 version

CS3230 in Y1/Y2/Y3 Sem2

Too late if you do this on Y4 Sem2
You will avoid me if you do Sem1 version

CS4234 in Y2/Y3/Y4 Sem1

Most students cannot take CS4234 in Y1
There is no CS4234 in Sem2

CS3233 in Y2/Y3/Y4 Sem2

Most students cannot take CS3233 in Y1
There is no CS3233 in Sem1

CS4234 Overview (1)

Optimization Algorithms ('z'/American and 's'/British)

- https://www.comp.nus.edu.sg/~stevenha/cs4234.html

Materials updated from 8 AYs ago, has branched out quite far:

- https://www.comp.nus.edu.sg/~gilbert/CS4234/

Lecturer: **Associate Professor** Steven Halim

- Office: COM2-03-37 (unless WFH)
- ~~Office Phone: not used~~, use email/Discord
- Email: dcssh@nus.edu.sg
- Discord: "prof_halim"

CS4234 Overview (2)

- Read my IOI stuff (IOI 2021 and IOI 2022)
  or my ICPC stuff (ICPC HCMC 2022)
  or my achievements (NDA18, ATEA list)
- Office: COM2-03-37
  - (usually in every Mon-Fri, 10am-5pm)
- Office Phone: [not used, just email me]
- Email: dcssh@nus.edu.sg
  - 'prof_halim' at Discord
  - On Instagram, ~~Facebook, Telegram~~
- I have 1 wife (Grace) and
  3 'J' kids (**J**ane, **J**oshua, **J**emimah)
- Wish for CS4234 S1 AY23/24:
  I need to scale up, preparing for 60pax, gets 84…

CS4234 Overview (3)

Tutor: Me + Zheng Han (A+ last AY), Jun Jie (A- last AY)

- Slots: All F2F, at COM3-01-21 (corner of COM3, The Terrace level)
  - T1: Mon, 11am-12noon (with Zheng Han)
  - T2: Mon, 2-3pm (with Zheng Han)
  - T3: Mon, 5-6pm (with Jin Jie)
- If sick (MC), etc
  - Show TA a valid proof (MC), you will be excused from tutorial, no penalty,
  - But read the reference answers carefully,
  - Ask others in Discord (@staff) if you have further questions

Optimization Algorithms

Optimization:

Find the minimum/maximum:

Discrete: a collection of items

Combinatorial: a collection of items generated by counting, combining, and enumerating

Examples:

Integers

Graphs and Trees

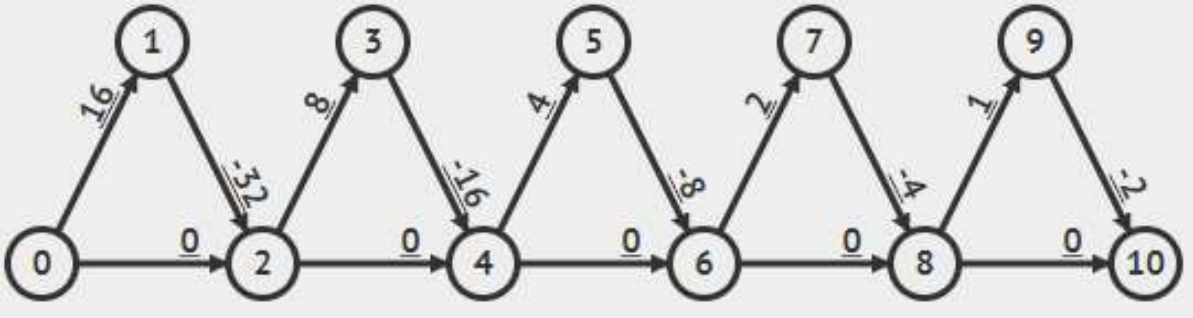
Similar structures…

Search Space Size != Difficulty

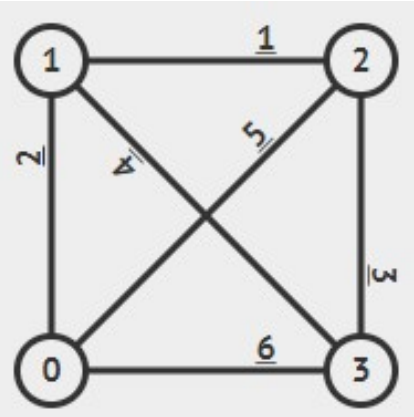
- Searching
- Shortest Paths

A = [5, 4, 2, 7, 3, 6, 1], O(N) to find-min, how?
Sorted-A = [1, 2, 3, 4, 5, 6, 7], O(1), how?

  - 2^5=32 different (how to count?) (acyclic) simple paths from 0→10
    - With a correct algorithm, which is _____, we can solve this in O(V+E)



- Minimum Spanning Tree
  - $4^{4-2} = 4^2 = 16$ different (how to count?) spanning trees
    - With a correct algorithm, which is _____, we can solve this in O(E log V)



Combinatorial Optimization

Find the "best" item in a large set of items:

Problem	Set of items	Size	Difficulty
Searching	List of integers	Linear	Easy
Shortest Path	All paths in a graph	Exponential	Easy
Minimum Spanning Tree	All spanning trees	Exponential	Easy
Steiner Tree	All steiner trees	Exponential	Hard
Travelling Salesman	All possible tours	Exponential	Hard
Matching	All possible matchings	Exponential	Easy
Bipartite Vertex Cover	All possible covers	Exponential	Easy
Vertex Cover	All possible covers	Exponential	Hard
Maximum Clique	All possible subsets	Exponential	Hard

Combinatorial Optimization

Find the "best" item in a large set of items:

Problem	Difficulty
Maintain student records	Easy
Exam timetable scheduling	Hard
Program halting problem	Impossible
Data compression	Easy
VLSI chip layout	Hard
Job assignment problem	Easy
Computer deadlock problem	Easy
Finding patterns in a database	Easy

Combinatorial Optimization

Operations Research:

How to make better decisions (e.g., maximize profit)

- Project planning / critical path analysis
- Facility location: where to open stores / plants
- Floorplanning: layout of factory or computer chips
- Supply chain management
- Berth assignment problem (BAP): port management
- Assignment problems (e.g., weapon target assignment)
- Routing / transportation problems: buses, subways, trucking.
- Airline ticket pricing

Optimization Algorithms

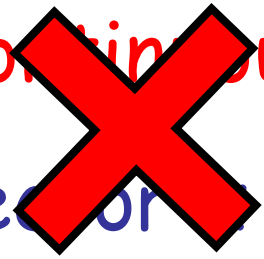
Optimization:

Find the minimum/maximum:

Discrete: a collection of items

Combinatorial: a collection of items generated by counting, combining, and enumerating

Continuous: given a function f(x), find the vector x that maximizes f(x)

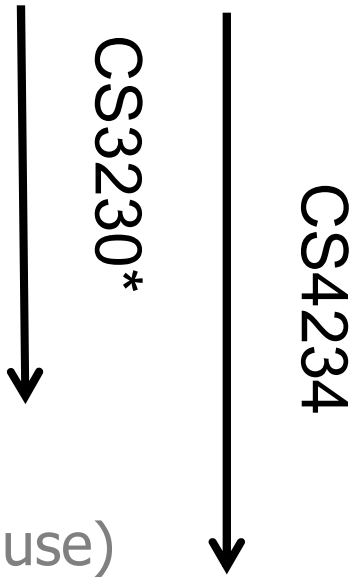


Calculus stuff, not the focus of CS4234

Optimization Algorithms

Goals: Algorithmic

1. Design (problem solving)
2. Analysis (rigorous, deep understanding)
3. Implementation (able to put it to use)



*Steven has just co-taught CS3230 S2 AY 2022/23 with Dr Diptarka Chakraborty and has turned CS3230 to have a bit more implementation side too :O…
But if you took CS3230 at other (earlier) semesters, you probably encountered CS3230 with a much more theoretical side than practical side

"If you need your software to run twice as fast, hire better programmers.

But if you need your software to run more than twice as fast, use a better **algorithm.**"

-- *Software Lead at Microsoft*

CS4234 : Optimization Algorithms

Now the eight iteration under Prof Halim

- First offered in S1 AY16/17
- Poor in S1 AY20/21 (4.1 ☹)
  - Online IOI 2020… very busy that semester
- Personal **Best** recently (S1 AY21/22+S1 AY22/23)
- Average rating of **4.504/5** for the past 7 offerings
- *Subset* of CS5234 (which has also evolved)
  - CS5234: Combinatorial and Graph Algorithms
  - More general: all sorts of optimization (not only graphs)
  - More specific: just optimization

CS4234 : Students

Target students:

- Advanced (3rd or 4th year) undergraduates
  - There are (a few) master/exchange students
    - And (a few) ex CS3233 students
- (Very) interested in algorithms
- Interested in tools for solving hard problems

Prerequisites:

- CS3230 (Analysis of Algorithms) → CS2040(S) → CS1231(S)
- Mathematical fundamentals (currently MA1101R, but good performance in CS1231(S) is needed too)

Pre-requisites (1)

You must already know these (use https://visualgo.net to review):

- Data Structures (with Analysis)
  - Linked Lists, Stacks, Queues,
  - (Binary) Heaps and Priority Queues,
  - Binary Search Trees, Balanced BSTs
- Algorithm Design Paradigms (with Analysis)
  - Standard Sorting and Searching algorithms,
  - Complete Search, Divide-and-Conquer, Greedy Algorithms, Dynamic Programming,
  - Graph traversal algorithms: DFS, BFS,
  - MST Algorithms and Shortest Path Algorithms,

Pre-requisites (2)

You must already know these (use https://visualgo.net to review):

- Analysis of Algorithms
  - Expertise with Big-O, Ω, Θ notations
  - Summation of series, Master Theorem
  - Competent with Algorithmic Analysis:

    Quicksort, Heapsort, Divide-and-Conquer algorithms

    DFS, BFS, MST & Shortest Paths algorithms

PS1 – Prerequisites (1)

- https://nus.kattis.com/courses/CS4234/CS4234_S1_AY2324/assignments/sq7y7j/standings, (4% !!!)
  - So that I can quickly be alerted on any "Prerequisites" issue involving some of you
  - This is an **elective** level 4 algorithm course after all…
- The easiest problem in the set is problem _____
  - I expect **everyone** to AC this by the end of Wk2, eventually
  - It involves a simple _____ algorithm that has been covered in CS3230
  - A prerequisite for the early topics of CS4234
- The second easiest problem in the set is problem _____
  - I initially thought it was a ____ problem but the underlying graph is _____
  - It involves _____ algorithm again that has been covered in CS2040/CS3230 (now seen as an ____ problem)
- The first 2 tasks should be doable
  - Please get AC(cepted) for at least these 2 out of 4 tasks
  - Strongly reconsider taking CS4234 this sem if you struggled badly just to get these 2 ACs

PS1 – Prerequisites (2)

- The next 2 tasks are sufficiently more challenging
  - Do not feel bad if they are still not AC after deadline, there will be partial marks!!
  - But still of CS2040/C/S and/or CS3230 level
- They involving spanning trees (frequently used in CS4234):
  - SSSP (produces a spanning tree), which is problem _____
    - Use this algorithm _____ or possibly using _____ algorithm
  - MST (also produces a spanning tree), which is problem _____
    - Use this algorithm _____
  - Graph Traversal (yeah, also produces a spanning tree), which is problem _____
    - Use this algorithm _____
  - Details not discussed today
    - But maybe during next Week 02 depending on the situation
- Task E+F+G+H are not discussed openly
  - They are for "extra challenges" for those who found CS4234 too easy
    - But interested students should try problem F (recursionrandfun) to "replace" B

CS4234 Important Dates

- Four PS deadlines on Week 02, 04, Recess, 09

  Must register NOW, latest by the end of next Week 02!
  https://nus.kattis.com/courses/CS4234/CS4234_S1_AY2324/register

  Secret registration key in the welcome email!
- Mini project deadline on Week 12
- Mid-term Test @ COM3-MPH, F2F, Week 08

  Thursday, 12 October 2023, using all lecture time
- VisuAlgo Online Quiz @ COM3-MPH, F2F, Week 09

  Thursday, 19 October 2023, using a bit of lecture time
- Final Assessment @ ??, F2F

  Thursday, 30 November 2023, 9-11 AM

CS4234 Grading

- Grading (last 7 years → this 8th iteration)

20%→16%	Problem Sets
15%→10%	Mini Project
9%	VisuAlgo Online Quiz
5%	Tutorial Attendance + Participation
25%	Mid-term Test
35%	Final Assessment

- Problem Sets
  - 4 sets (approximately every 2-3 weeks, 5→4% each)
  - Algorithm design, analysis, with implementation
    - Using **just one** Online Judge : nus.kattis

CS4234 Grading

- Mini-Project
  - Small project, four→five (84/5 = 17 groups, **TBC**)
  - Idea: To put together some of the different ideas we have used in the class
  - Time scale: Last 3 weeks of the semester
  - 15→10%
- Tutorial Attendance & Participation
  - 5%, bell curve system, i.e.,
    more active in F2F class = more points

CS4234 Problem Sets

- Already Released

  PS1-Prerequisites:

  Released on Week 00 (Wed, 02 Aug 2023, 08.00am) but not due until the end of Week 02 (Sun, 27 Aug 2023, 11.59pm), so that you can use this information to retain/drop CS4234

  As of Mon of Week 01, out of 84 students:
  - 13 have cleared all ABCD+EFGH (too easy? :O)
  - 29 have cleared ABCD
  - 33 have cleared AB (considered safe)
  - 59 students have registered
  - **Last 25** are still missing (haven't even register)
  - Steven will check again on Thu of Week 02…

CS4234 Problem Sets

- Problem Set Grading (per task)

  Simple scheme (graded with help of Online Judge):

  1% : Perfect (coding style is ignored)

  0.7% : Excellent, one minor error

  0.4% : We (staff) find two errors

  0.1% : Too many mistakes, we cannot fix

  0% : No submission (or caught by plagiarism checker)

CS4234 Problem Sets

- What to submit (online judge version):

  C++ (17), Python (3), or Java (11) code that solves the optimization problem(s) --- I am not familiar with other languages :O

  Especially if not AC at the end: Write additional comments inside source code (only the last submitted code will be *randomly* read) – optional if AC:

  1. Summary of the coded algorithm in English
  2. Write a "proof of correctness" (hopefully your non AC is just due to coding bug) and "performance analysis" (hopefully your TLE is just due to constant factor weakness)

CS4234 Problem Sets

- Policy on plagiarism:

  Do your work yourself:

  Your submission should be *unique*, unlike anything else submitted, on the web, etc

  You can discuss with other students (e.g., Discord), but...:

  1. Discuss only general approach and techniques
  2. Just take general notes during such discussions
  3. Code the solution on your own
  4. List all collaborators (including ChatGPT/eq) in your submission

  ~~Do not ask for solutions on the web:~~ (you CAN use ChatGPT/eq)

  Use web to learn techniques and to review material from class; ~~students who ask about CS4234 graded assignments in StackOverflow, Quora, Generative AI will be punished~~

CS4234 Problem Sets

- Policy on plagiarism:

  Penalized severely:

  First offense: Minimum of one letter grade lost on final grade for class as I will remove your 4% of that PS

  Second offense: F for the class and referral to SoC disciplinary committee

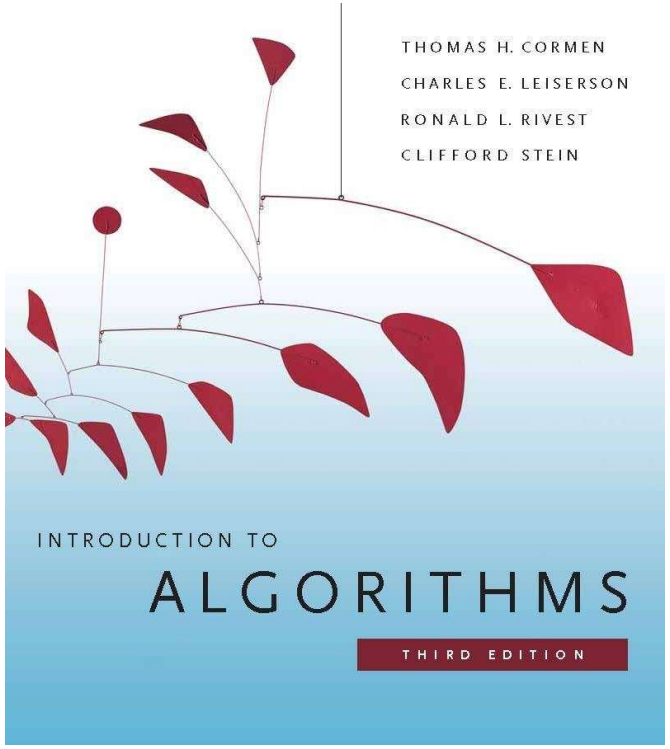
  Do not copy/compare solutions!

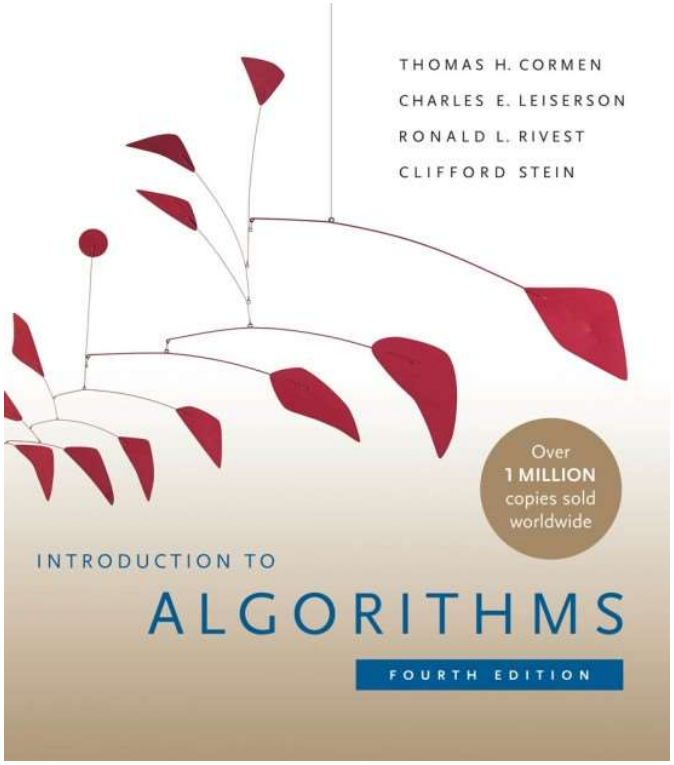
  PS: Do not store your coded solutions in public repository (e.g., GitHub, ideone.com, repl) with public setting ☹, Do not use ChatGPT/eq answer **verbatim** (someone else may use too)

Textbooks

Introduction to Algorithms

- Cormen, Leiserson, Rivest, Stein



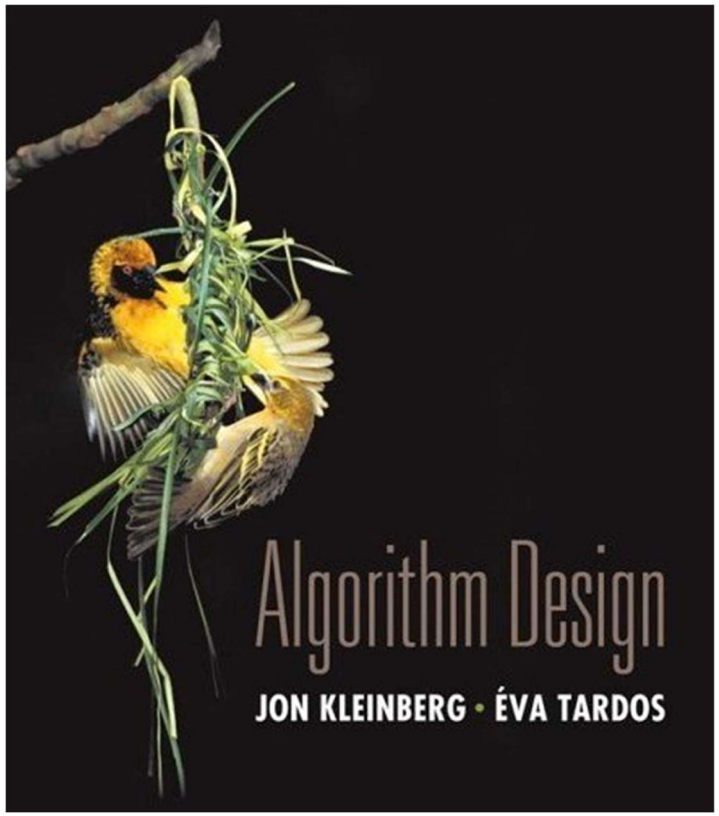


- Strongly Recommended (I have both versions)…

Textbooks

Algorithm Design

- Kleinberg and Tardos



- Recommended…

Textbooks

Competitive Programming 4 - ***esp Book 2***

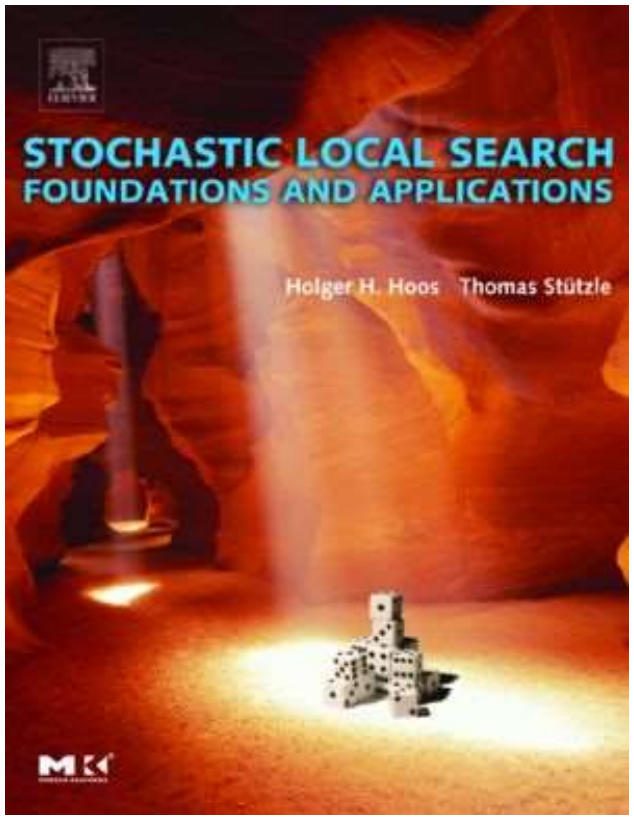
- Steven Halim, Felix Halim, and Suhendry Effendy

- Only a subset of these are relevant (especially Book 2 - Chapter 8+9)…
  - Also get Book 1 if you think your DS/BF/Greedy/DnC/DP/simple Graph concepts are still shaky
    - Book 1 has run out of stock, get the e-book version
- Book 2 is still available in NUS co-op, BUY NOW, clear the stock
  - Or get the e-book version, see, https://cpbook.net/details?cp=4

Textbooks

Stochastic Local Search: Foundations and Applications

- Holger H. Hoos & Thomas Stützle



- For the last part of this course (a hard book to get)

CS4234 Topics

- Topics (90% same as last 7 AYs)

  Introduction to Combinatorial Optimization

  Vertex Cover, Set Cover, Steiner Tree, TSP

  Linear Programming

  LPs, Relaxations, Rounding

  Flows and Matching (major expansion last AY during my external review)

  Maximum Flow (PR returns on Week 07),

  (Bipartite) Matching (now plus weighted)

  Local Search

  Gradient Descent, Meta-heuristics