CSE 421: Introduction to Algorithms

Stable Matching

Yin-Tat Lee

Quiz

Best paper on FOCS 2018

Approximating Edit Distance Within Constant Factor in Truly Sub-Quadratic Time*

Diptarka Chakraborty†1, Debarati Das‡2, Elazar Goldenberg§3, Michal Koucký¶4, and Michael Saks‖5

1,2,4Computer Science Institute of Charles University, Malostranské náměstí 25, 118 00 Praha 1, Czech Republic
3The Academic College Of Tel Aviv-Yaffo, School of Computer Science, Tel Aviv-Yaffo, Israel
5Department of Mathematics, Rutgers University, Piscataway, NJ, USA

July 13, 2018

See arXiv:1804.04178

Input: two n-bit strings s_1 and s_2.

- $s_1 = AGGCTACC$
- $s_2 = CAGGCTAC$

Output: minimum number of insertions/deletions to transform s_1 into s_2.

Algorithm: ????

Even if the objective is precisely defined, we are often not ready to start coding right away!

After this quarter, you should able to solve it by $O(n^2)$ time.

Open problem: Is $O(n^{1.999})$ possible? (Probably not due to Strong ETH)

Message: Don't be discouraged if it takes 30 years to solve a problem.

However, limit your time of each homework to just 10 years. There are 8 homework.

This Course

How to solve problems using computers, aka, algorithms.

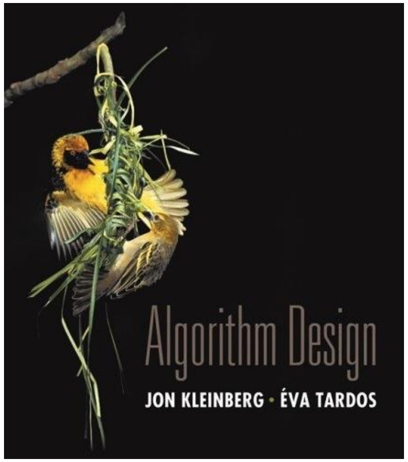
Goal:

- Learn some techniques to design algorithms
- How to analyze the runtime
- Understand some problems are difficult

Grading Scheme

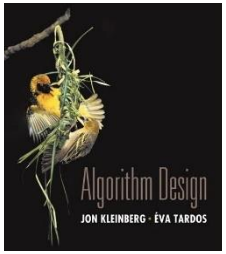
- Homework ~ 50%
  Weekly homework due on Wed before the class
- Midterm ~ 15-20%
- Final ~ 30-35%

HW 1 is already out! Due next Wed before class!



Course textbook

Where to get help?



- Ask questions in the class!
- Read the textbook!
- Piazza: Online discussion forum.

Last year we used Canvas.
Students said Piazza is better.

- Office hours:
- Myself: M 2:30-3:30, Tu 4:30-5:30 in CSE 562

TA	Office hours (from Oct 01 - Dec 07)	Room
Dongkai Xu	Mon 10:30am-11:30am	CSE 007
Xin Yang	Mon 04:30pm - 05:30pm	CSE 007
Mathew Luo	Tue 11:30am-12:30pm	CSE 007
Swati Padmanabhan	Tue 12:30pm-01:30pm	CSE 007
Guanghao Ye	Tue 03:00pm-04:00pm	CSE 007
Faye Yu	Wed 04:30pm- 05:30pm	CSE 007
Zongyuan Chen	Thu 01:30pm-02:30pm	CSE 007
Leiyi Zhang	Fri 12:30pm-01:30pm	CSE 007

CSE 007
(starting from next Monday)

CSE 421: Introduction to Algorithms

Autumn, 2018

Yin Tat Lee

> **MWF 1:30-2:20**, THO 101
> Office hours M 2:30-3:30, Tu 4:30-5:30 in CSE 562

Links:
- Assignment: Canvas
- Discussion: Piazza
- Mailing List: cse421a_au18 [archives]
- Email TA and instructor: cse421staff18au@cs.washington.edu
- Textbook: Algorithm Design by Jon Kleinberg and Eva Tardos
  We will cover chapters 1-8.
- Reference: Introduction to Algorithms by CLRS

Grading Scheme (Roughly):
- Homework (50%)
- Midterm (20%)
- Final (30%)

Website: cs.washington.edu/421

Yin-Tat's Research

Study theoretical computer science.

Among others, I gave the (theoretically) fastest algorithms for
- Linear Programs
- Network flow

Two topics that we will discuss in this course.

Some problems in this course are still a hot area of research!

Stable Matching Problem

Given n men and n women, find a "stable matching".

- We know the preference of all people.

	favorite		least favorite
	1st	2nd	3rd
Xavier	Amy	Brenda	Claire
Yuri	Brenda	Amy	Claire
Zoran	Amy	Brenda	Claire

Men's Preference Profile

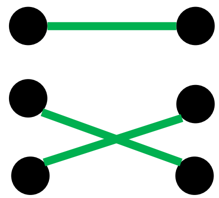
	favorite		least favorite
	1st	2nd	3rd
Amy	Yuri	Xavier	Zoran
Brenda	Xavier	Yuri	Zoran
Claire	Xavier	Yuri	Zoran

Women's Preference Profile

Stable Matching

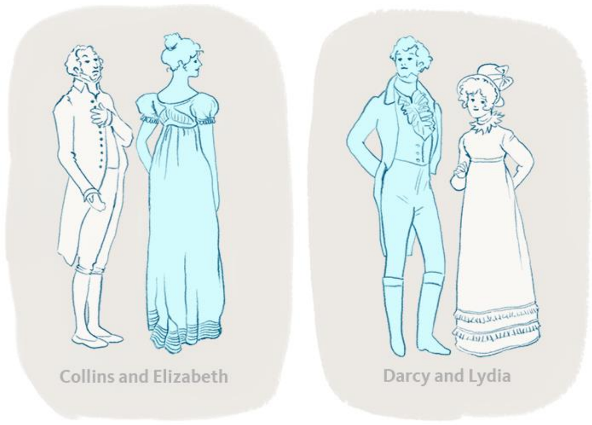
Perfect matching:

- Each man gets exactly one woman.
- Each woman gets exactly one man.



Stability: no incentive to exchange

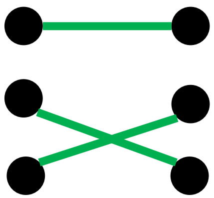
- an unmatched pair **m-w** is unstable
- if man **m** and woman **w** prefer each other to current partners.



An unstable match:
ELIZABETH AND DARCY LIKE EACH OTHER BETTER THAN THEIR PARTNERS

Stable Matching

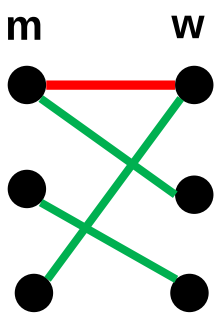
Perfect matching:



- Each man gets exactly one woman.
- Each woman gets exactly one man.

Stability: no incentive to exchange

- an unmatched pair **m-w** is unstable
- if man **m** and woman **w** prefer each other to current partners.



Stable matching: perfect matching with no unstable pairs.

Stable matching problem: Given the preference lists of n men and n women, find a stable matching if one exists.

Example

Question. Is assignment X-C, Y-B, Z-A stable?

	favorite ↓ 1st	2nd	least favorite ↓ 3rd
Xavier	Amy	Brenda	Claire
Yuri	Brenda	Amy	Claire
Zoran	Amy	Brenda	Claire

Men's Preference Profile

	favorite ↓ 1st	2nd	least favorite ↓ 3rd
Amy	Yuri	Xavier	Zoran
Brenda	Xavier	Yuri	Zoran
Claire	Xavier	Yuri	Zoran

Women's Preference Profile

Example

Question. Is assignment X-C, Y-B, Z-A stable?

Answer. No. Brenda and Xavier will hook up.

	1st (favorite)	2nd	3rd (least favorite)
Xavier	Amy	Brenda	Claire
Yuri	Brenda	Amy	Claire
Zoran	Amy	Brenda	Claire

Men's Preference Profile

	1st (favorite)	2nd	3rd (least favorite)
Amy	Yuri	Xavier	Zoran
Brenda	Xavier	Yuri	Zoran
Claire	Xavier	Yuri	Zoran

Women's Preference Profile

Example (cont'd)

Question: Is assignment X-A, Y-B, Z-C stable?

Answer: Yes. (X, Y are happy. No one want Z.)

favorite → 1st; least favorite → 3rd

	1st	2nd	3rd
Xavier	Amy	Brenda	Claire
Yuri	Brenda	Amy	Claire
Zoran	Amy	Brenda	Claire

Men's Preference Profile

	1st	2nd	3rd
Amy	Yuri	Xavier	Zoran
Brenda	Xavier	Yuri	Zoran
Claire	Xavier	Yuri	Zoran

Women's Preference Profile

Existence of Stable Matchings

Question. Do stable matchings always exist?
Answer. Yes, but not obvious.

Stable roommate problem:

2n people; each person ranks others from **1** to **2n-1**.
Assign roommate pairs so that no unstable pairs.

	1st	2nd	3rd
Adam	B	C	D
Bob	C	A	D
Chris	A	B	D
David	A	B	C

A-B, C-D $\Rightarrow$ B-C unstable
A-C, B-D $\Rightarrow$ A-B unstable
A-D, B-C $\Rightarrow$ A-C unstable

Stable matchings do not always exist for stable roommate problem.

Propose-And-Reject Algorithm [Gale-Shapley'62]

```
Initialize each person to be free.
while (some man is free and hasn't proposed to every woman) {
    Choose such a man m
    w = 1st woman on m's list to whom m has not yet proposed
    if (w is free)
        assign m and w to be engaged
    else if (w prefers m to her fiancé m')
        assign m and w to be engaged, and m' to be free
    else
        w rejects m
}
```

Switch the pdf for an example.

Tips: When stuck, try to list out properties of the algorithms.

Main Properties of the algorithm

Observation 1: Men propose to women in decreasing order of preference.

Observation 2: Once a woman is matched, she never becomes unmatched; she only "trades up."

What do we need to prove?

- The algorithm ends.
  How many iterations it takes?

- The output is correct.
  It find a perfect matching that is stable.

Proof of Correctness: Termination

Each step, a man proposed to a new woman.

There are $n \times n = n^2$ possible man-to-woman proposals.

Therefore, it takes at most n^2 iterations.

	1st	2nd	3rd	4th	5th
Victor	A	B	C	D	E
Walter	B	C	D	A	E
Xavier	C	D	A	B	E
Yuri	D	A	B	C	E
Zoran	A	B	C	D	E

	1st	2nd	3rd	4th	5th
Amy	W	X	Y	Z	V
Brenda	X	Y	Z	V	W
Claire	Y	Z	V	W	X
Diane	Z	V	W	X	Y
Erika	V	W	X	Y	Z

n(n-1) + 1 proposals required

Proof of Correctness: Perfection

Claim. All men and women get matched.

Proof. (by contradiction)

Suppose, for sake of contradiction, that Zoran is not matched upon termination of algorithm.

Then some woman, say Amy, is not matched upon termination.

(Observation 2: once women matched, they never becoming unmatched.) Amy was never proposed to.

But, Zoran proposes to everyone, since he ends up unmatched.

■

Proof of Correctness: Stability

Claim. No unstable pairs.

Proof. (by contradiction)

Suppose **A**-**Z** is an unstable pair: each prefers each other to the partner in Gale-Shapley matching.

Case 1: **Z** never proposed to **A**.

$\Rightarrow$ **Z** prefers his GS partner to **A**.

$\Rightarrow$ **A**-**Z** is stable.

men propose in decreasing order of preference

Case 2: **Z** proposed to **A**.

$\Rightarrow$ **A** rejected **Z** (right away or later)

$\Rightarrow$ **A** prefers her GS partner to **Z**.

$\Rightarrow$ **A**-**Z** is stable.

women only trade up

In either case **A**-**Z** is stable, a contradiction. ■

Summary

- Stable matching problem: Given **n** men and **n** women, and their preferences, find a stable matching.

- Gale-Shapley algorithm: Guarantees to find a stable matching for any problem instance.

- Q: How to implement GS algorithm efficiently?

- Q: If there are multiple stable matchings, which one does GS find?

Why this problem is important?

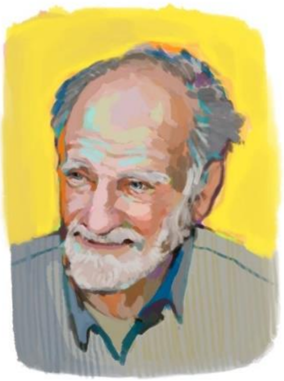
In 1962, Gale and Shapley published the paper
“College Admissions and the Stability of Marriage”
To
“The American Mathematical Monthly”

COLLEGE ADMISSIONS AND THE STABILITY OF MARRIAGE

D. GALE* AND L. S. SHAPLEY, Brown University and the RAND Corporation

1. Introduction. The problem with which we shall be concerned relates to the following typical situation: A college is considering a set of n applicants of which it can admit a quota of only q. Having evaluated their qualifications, the admissions office must decide which ones to admit. The procedure of offering admission only to the q best-qualified applicants will not generally be satisfactory, for it cannot be assumed that all who are offered admission will accept. Accordingly, in order for a college to receive q acceptances, it will generally have to offer to admit more than q applicants. The problem of determining how many and which ones to admit requires some rather involved guesswork. It may not be known (a) whether a given applicant has also applied elsewhere; if this is known it may not be known (b) how he ranks the colleges to which he has applied; even if this is known it will not be known (c) which of the other colleges will offer to admit him. A result of all this uncertainty is that colleges can expect only that the entering class will come reasonably close in numbers to the desired quota, and be reasonably close to the attainable optimum in quality.

David Gale (1921-2008)
PROFESSOR, UC BERKELEY



Lloyd Shapley
PROFESSOR EMERITUS, UCLA

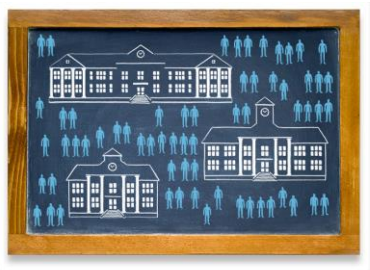
Why this problem is important?

Alvin Roth
PROFESSOR, STANFORD

Alvin Roth modified the Gale-Shapley algorithm and apply it to

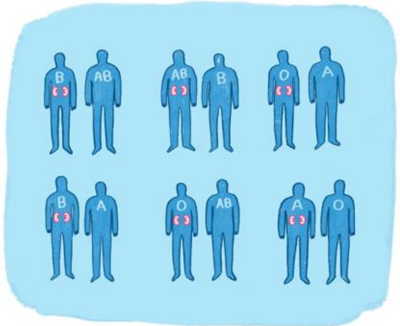
- National Residency Match Program (NRMP), a system that assigns new doctors to hospitals around the country. (90s)

- Public high school assignment process (00s)



- Helping transplant patients find a match (2004)

(Saved >1,000 people every year!)



Blood types: A, B, AB and O

Reference: https://medium.com/@UofCalifornia/how-a-matchmaking-algorithm-saved-lives-2a65ac448698

Why this problem is important?

Some of the problems in this course may seem obscure or pointless.

But their abstraction allows for variety of applications.

Shapley and Roth got the Nobel Prize (Economic) in 2012.
(David Gale passed away in 2008.)