

# 5112 Algorithms for Applications

Prof. Alex Conway



Join  
the  
Slack

Course  
Website



# Course Details

## Intro

## Algorithms

## Stable Matching (first real topic)

# Course Details



4-5 programming assignments  
no final, no midterm, no project

Website

Grades: 90% assignments, 10% participation

You may work in small groups, but each person must turn in their own writeups.

AI: Allowed, but tell me how you used it. If it solves the problem or generates output, include the prompts.

# Office Hours / Slack

My office hours:

Tue after class in Bloomberg 362 (or by appointment)

TA office hours:

On Slack, responses within 24 hours.

In person: announce on Slack

Slack



Intro

Algorithms

# Stable Matching

Candidates

Alex  
Jennifer  
Raj

Universities

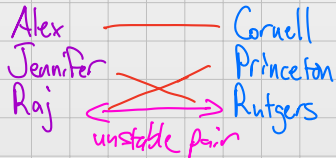
Cornell  
Princeton  
Rutgers

# Stable Matching

## Preferences

### Candidates

### Universities



Both Raj and Rutgers  
prefer each other to  
their current match

Alex

C

R

P

Cornell

J

A

R

Jennifer

P

R

C

Princeton

A

J

R

Raj

R

C

P

Rutgers

R

A

J

# Stable Matching

## Candidates

Alex

Jennifer

Raj

## Universities

Cornell

Princeton

Rutgers

← unstable pair →

Both Raj and Rutgers prefer each other to their current match

## Preferences

Alex

C

R

P

Jennifer

P

R

C

Raj

R

C

P

Cornell

J

A

R

Princeton

A

J

R

Rutgers

R

A

J

A stable matching is a matching w/o any unstable pairs.



## Preferences

Alex

C

R

P

Jennifer

P

R

C

Raj

R

C

P

A - C 3

A - P 4

Cornell

J

A

R

Princeton

A

J

R

Rutgers

R

A

J

# Gale-Shapley Algorithm (1962)

while  $\exists$  an unmatched university  $U$ :  
     $C \leftarrow$  next candidate  
    if  $C$  prefers  $U$  to their current match:  
        match  $C$  to  $U$ .  
    move next candidate pointer

<u>Preferences</u>		
Alex	Jennifer	Raj
C	P	R
R	R	C
P	C	P
Cornell	Princeton	Rutgers
$\rightarrow$ J	$\rightarrow$ A	$\rightarrow$ R
A	J	A
R	R	J

# Gale-Shapley Algorithm (1962)

while  $\exists$  an unmatched university  $U$ :  
     $C \leftarrow$  next candidate  
    if  $C$  prefers  $U$  to their current match:  
        match  $C$  to  $U$ .  
    move next candidate pointer

Preferences

Alex  
C  
R  
P

Jennifer  
R  
C  
P

Raj  
R  
C  
P

Cornell  
J  
A  
R

Princeton  
A  
J  
R

Rutgers  
J  
R  
A

Alex ——— Cornell  
Jennifer ——— Princeton  
Raj ——— Rutgers

Can we run out of candidates to make offers to?

Invariant: Once a candidate has an offer, they will always have an offer.