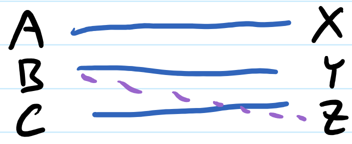
Stable Matching

Problem (informal): Given n machinists and n welders, find a good way to match them.

Machinist	Preferences	Welder	Preferences
A	X, Y, Z	X	A, B, C
B	X, Z, Y	Y	A, C, B
C	Z, X, Y	Z	A, B, C

(B, Z) is an instability wrt
(A, X)
(B, Y)
(C, Z)



matching $\left\{ (A, X), (B, Z) \right.$ perfect matching $\left. (C, Y) \right\}$

Matching: set of ordered pairs from $M \times W$ s.t. for each $m \in M$ there is at most 1 $w \in W$ s.t. (m, w) is in the matching and for each $w \in W$ there is at most 1 $m \in M$ s.t. (m, w) is in the matching

Perfect matching: a matching with each m, w matched with <u>exactly</u> one w', m'

Instability with respect to perfect matching S is a pair $(m, w') \notin S$ s.t. m prefers w' to whoever m is matched with in S and w' prefers m to whoever w' is matched with in S

Stable Matching: a perfect matching with no instabilities

Gayle-Shapely

```
FreeM <- M
FreeW <- W
Invitations <- {}
Tentative <- {}

While there is an m in FreeM s.t. there is a w s.t. (m,w) not in Invitations
    choose such an m
    let w be m's highest ranked s.t. (m,w) not in Invitations

    add (m,w) to Invitations

    if w in FreeW then
        remove w from FreeW
        remove m from FreeM
        add (m,w) to Tentative
    else
        find m' s.t. (m', w) in Tentative
        if w prefers m to m'
            remove m from FreeM
            add m' to FreeM
            remove (m', w) from Tentative
            add(m, w) to Tentative
return Tentative
```

$(m,w) \in$ Invitations means m invited w to form a team

$(m,w) \in$ Tentative means m is paired with w

$|\text{Invitations}| = \#$ iterations

$|\text{Invitations}| \le n^2$

so $\#$ iterations $\le n^2$

Invitations	FreeM	FreeW	Tentative
	A, B, C	X, Y, Z	
(C, Z)	A, B	X, Y	~~(C, Z)~~
(A, X)	B	Y	(A, X)
(B, X)			
(B, Z)	C	Y	(B, Z)
(C, X)			
(C, Y)			(C, Y)

stable matching

Machinist	Preferences
A	X, Y, Z
B	X, Z, Y
C	Z, X, Y

Welder	Preferences
X	A, B, C
Y	A, C, B
Z	A, B, C

Does this always terminate?

Does this return a stable matching?

What is the running time?

Asymptotic Running Time

$T(n)$ is $O(f(n))$ means $\exists$ (there exists) $n_0 \ge 0, c > 0$ s.t. $\forall n \ge n_0$, $T(n) \le c \cdot f(n)$

$\Omega(f(n))$ $\exists n_0 \ge 0, c > 0$ s.t. $\forall n \ge n_0$, $T(n) \ge c \cdot f(n)$ $\quad \cancel{3n^2 + 10n} \le n^2$

For $n \ge 1$, $3n^2 + 10n \le 13n^2$
so $3n^2 + 10n$ is $O(n^2)$

$\Theta(f(n))$ means $T(n)$ is $O(f(n))$
and $T(n)$ is $\Omega(f(n))$

Suppose Algs A, B (bubblesort, heapsort) both solve problem P (sorting) and that worst case for A is $\Theta(f(n))$ $\quad \Theta(n^2)$
and worst case for B is $\Theta(g(n))$ $\quad \Theta(n \log n)$

and $f(n)$ is $\Omega(g(n))$ but not $\Theta(g(n))$

then for sufficiently large n, there are inputs of size $\ge n$ s.t. A is slower than B

If f is $O(g)$ and g is $O(h)$ then f is $O(h)$ (transitive)
f is $\Omega(g)$ and g is $\Omega(h)$ then f is $\Omega(h)$
f is $\Theta(g)$ and g is $\Theta(h)$ then f is $\Theta(h)$

f is $O(h)$ and g is $O(h)$ then $f+g$ is $O(h)$
n is $O(n^3)$ $\quad 3n^2+14$ is $O(n^3)$ $\quad 3n^2+n+14$ is $O(n^3)$

f is $O(g)$ then $f+g$ is $O(g)$ (and $f+g$ is $\Omega(g)$)
$f \le g$
$f+g \le 2g$
and $f+g$ is $\Theta(g)$

If f is a polynomials of degree d (with positive coefficients on n^d term) then

f is $\Theta(n^d)$

$10n^4 + 14n^3 + 14926n^2 - 10n + 47$ is $\Theta(n^4)$

For every $\varepsilon, d \ge 0$ $\quad n^d$ is $O(n^{d+\varepsilon})$ $\quad (\log_2 n)^{10}$ is $O(\sqrt{n})$

For every $b > 1$, $d > 0$, $\quad \log_b n$ is is $O(n^d)$ $\quad \log_{100} n$ is $O(\sqrt{n})$

For every $d > 0$, $r > 1$, $\quad n^d$ is $O(r^n)$ $\quad n^{10000}$ is $O(1.001^n)$

For $\varepsilon \ge 0$, $r > 1$ $\quad r^n$ is $O((r+\varepsilon)^n)$

if $\lim_{n\to\infty} \frac{f(n)}{g(n)}$ exists and $= 0$ $\quad f(n)$ is $O(g(n))$
∞ $\quad f(n)$ is $\Omega(g(n))$
pos const $\quad f(n)$ is $\Theta(g(n))$

Suppose f is $O(g)$ and g is $O(h)$ [want f is $O(h)$]
Then, by def of O, $\exists n_1 \ge 0, c_1 > 0$ s.t. $\forall n \ge n_1$, $f(n) \le c_1 \cdot g(n)$
and $\exists n_2 \ge 0, c_2 > 0$ s.t. $\forall n \ge n_2$, $g(n) \le c_2 \cdot h(n)$
So $\underbrace{\forall n \ge n_1 \text{ and } n \ge n_2}_{n \ge \max(n_1, n_2)}$, $f(n) \le c_1 \cdot c_2 \cdot h(n)$

$\therefore$ $\exists n_0 \ge 0$ ($n_0 = \max(n_1,n_2)$) and $c > 0$ ($c = c_1 \cdot c_2$) s.t. $\forall n \ge n_0$, $f(n) \le c \cdot h(n)$
$\therefore$ f is $O(h)$