# CS 173: Discrete Structures

Eric Shaffer
Office Hour: Wed. 12-1, 2215 SC
shaffer1@illinois.edu

# Announcements

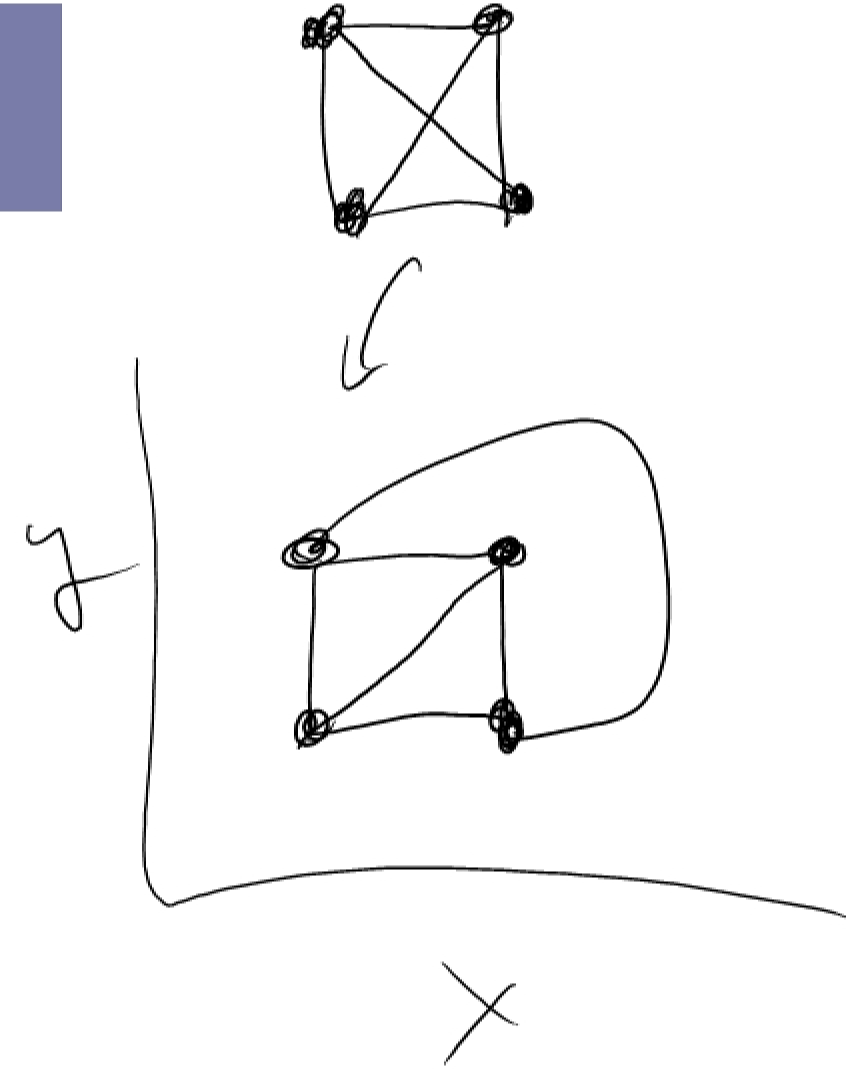
- HW 11 is due Wednesday…
- Tomorrow
  - TAs will have office hours at least from 2:30-4:30
  - Check newsgroup -- there may be more
- Final Exam
  - 1:30-4:30pm, Tuesday May 12th, in 1 LH-THEAT
  - Conflict exam 8-11am, Wed. May 13th, in 1404 Siebel

# Agenda

- Today
  - Non-planar graphs (section 9.7)
  - Applications of graphs (don't worry -- no testing on this)
  - Unsolicited advice (and no testing on this either)

# Non-planar graphs

- A graph is called ***planar:***
  - if it can be drawn in a 2D plane with no edges crossing
  - edge crossing is an intersection of the lines representing edges
- Some graphs are ***non-planar:***
  - any 2D drawing of a non-planar graph will have edge crossings
- An ***embedding*** of a graph in space
  - Assigns coordinates to its vertices
- Remember, a graph is not inherently spatial
  - It's just an abstract object that specifies connectivity

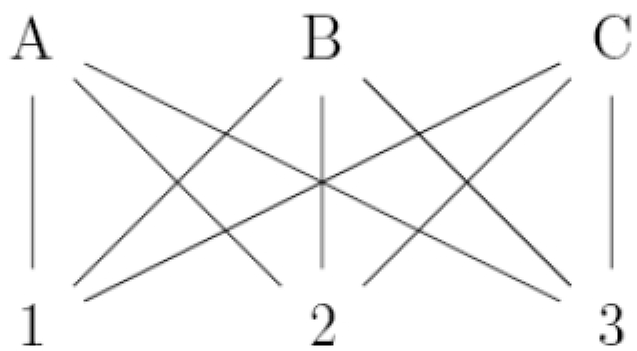


# Which graphs are non-planar?

How can we prove a graph is planar? Draw it w/no edge crossings

How can we prove a graph s non-planar?

One example: $K_{3,3}$

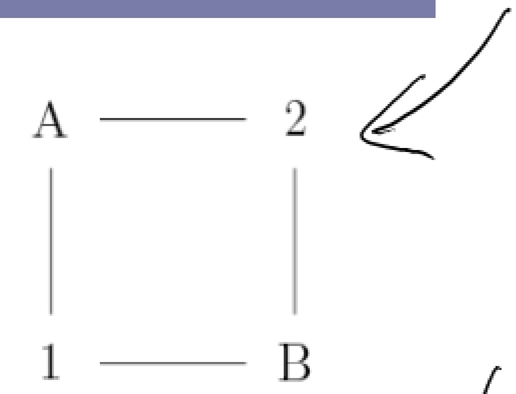


# Which graphs are non-planar?

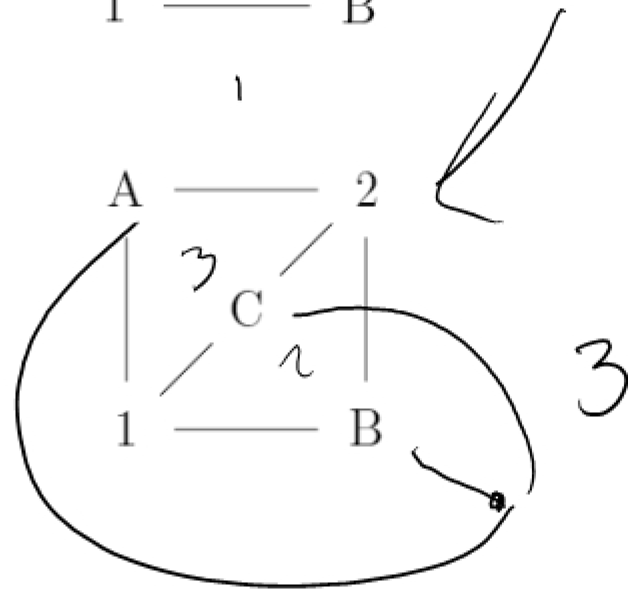
$K_{3,3}$ is non-planar

Proof:

The vertices A,B,1,2 form cycle




C must be either inside or outside,
assume it is inside




Vertex 3 must lie in one of three regions
It also must connect to A,B, and C

But no region has A,B, and C on the boundary

An edge must cross another edge

# Which graphs are non-planar?

$e$ = # edges
$v$ = # vertices
$f$ = # regions

We can also use Euler's formula to prove some useful facts

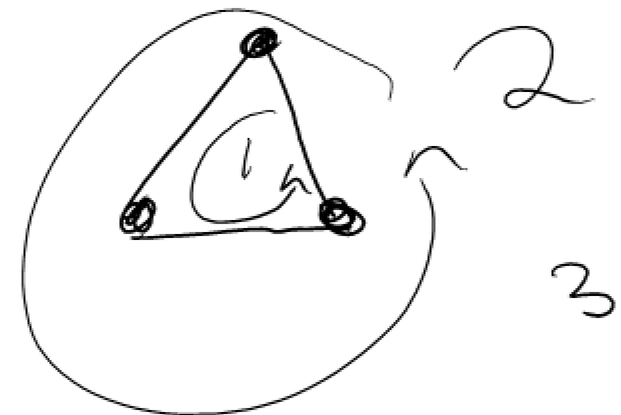
Let G be a connected, simple, planar graph

**Corollary 1: When $f \geq 3$, $e \leq 3v-6$**

Proof: Each "face" is bounded by **at least** 3 edges

Each edge is on 2 boundaries SO $2e \geq 3f$

We know $e-v+2=f$ so we can get $e-v+2 \leq (2/3)e$



$$2e \geq 3f$$

$$\frac{2}{3}e \geq f$$

$$e - v + 2 = f$$

$$e - v + 2 \leq \frac{2}{3}e$$

**Corollary 2: If G is a connected, simple, planar graph then G has a vertex of degree not exceeding 5**

Proof: Let G be a graph with 3 or more vertices (cases for less than 3 easy)

If every vertex has degree of 6 or more then $2e \geq 6v$

BUT this would contradict $2(e) \leq 2(3v-6)$

SO some vertex must have degree 5 or less

contradiction

# Which graphs are non-planar?

$K_5$ is non-planar

Proof:

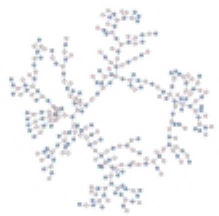
We have v=5 and e=10

So, $e \leq 3v-6$ does NOT hold for this graph

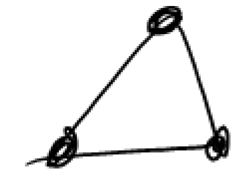
The graph is clearly simple and connected, so it must be non-planar

$$10 \not\leq 3(5)-6$$

$$10 \not\leq 9$$



# Which graphs are non-planar?

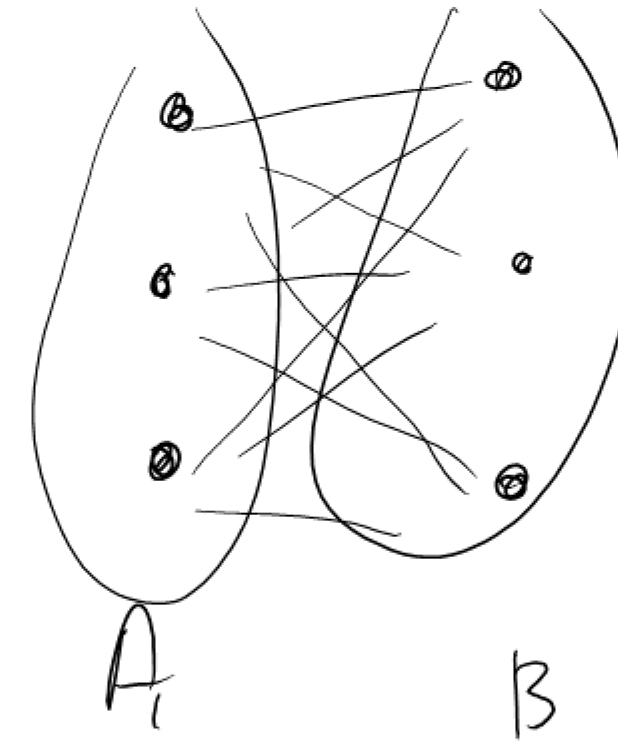


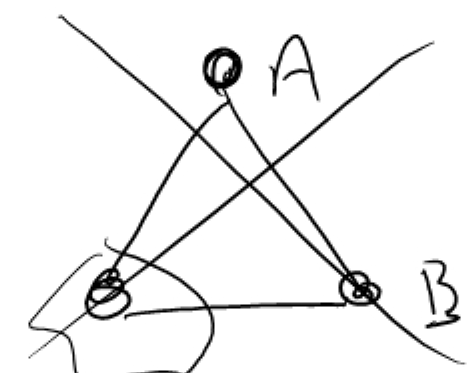
If graph G is connected, simple and planar with no circuits of length 3
Then $e \leq 2v-4$

Use this to prove $K_{3,3}$ is non-planar

$9 \not\leq 2(6)-4$

$9 \not\leq 8$

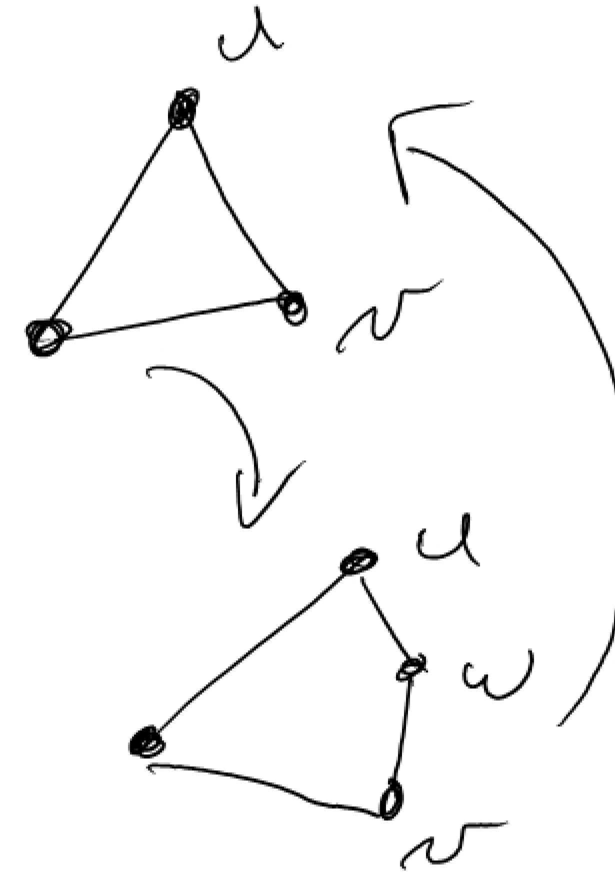


# Homeomorphism

- An ***elementary subdivision***
  - Removes an edge (u,v)
  - Adds vertex w and edges (u,w) and (w,v)
  - Like splitting an edge
  - Preserves planarity

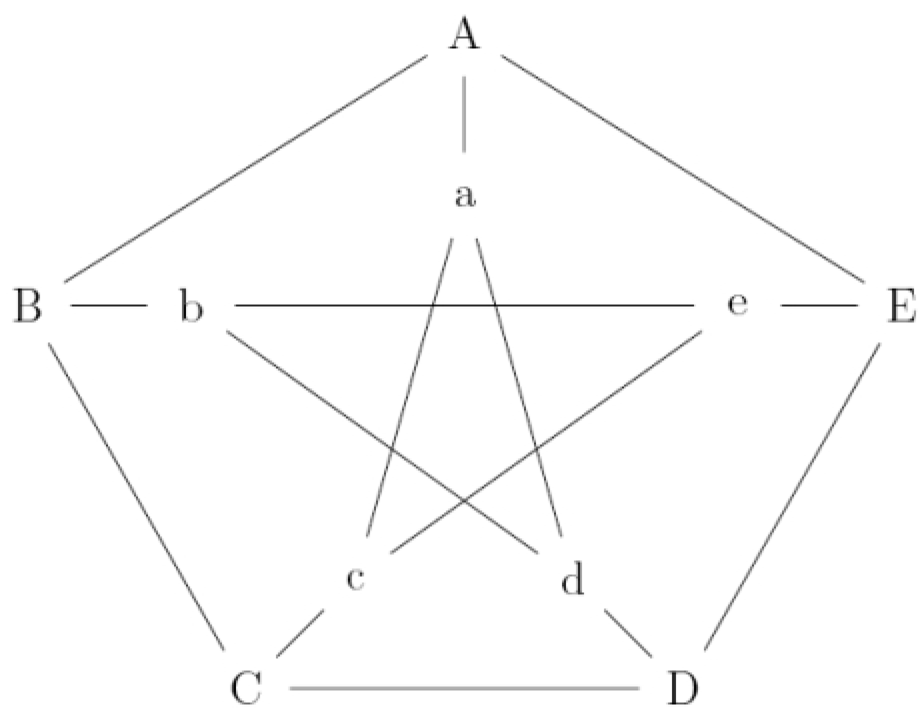
- Two graphs are ***homeomorphic*** if
  - The can be obtained from the same graph by a sequence of elementary subdivisions



# Kuratowski's Theorem

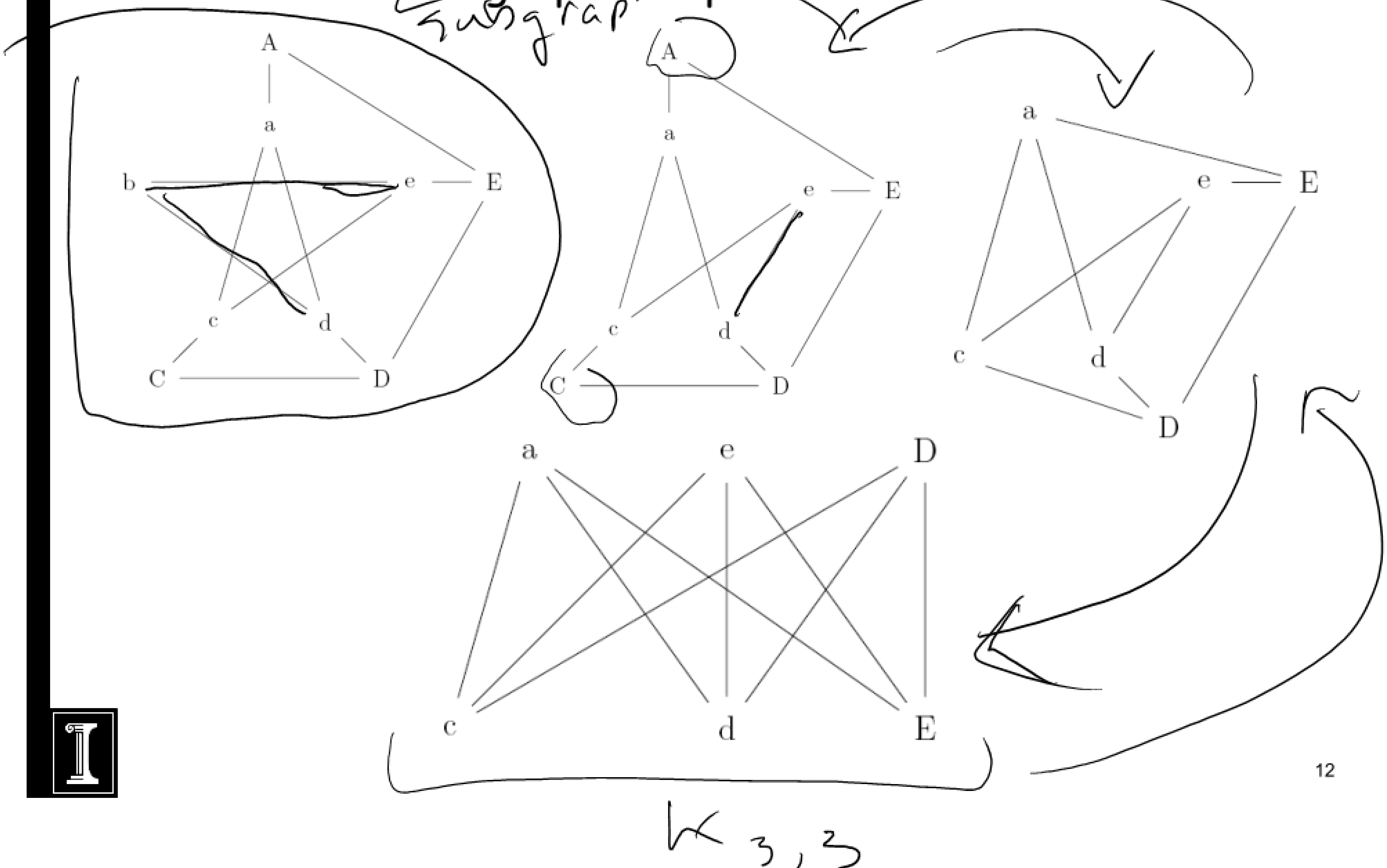
A graph is non-planar if and only if it contains a subgraph homeomorphic to $K_5$ or $K_{3,3}$

Is the Petersen graph non-planar?



# Kuratowski's Theorem

Is the Petersen graph non-planar? YES

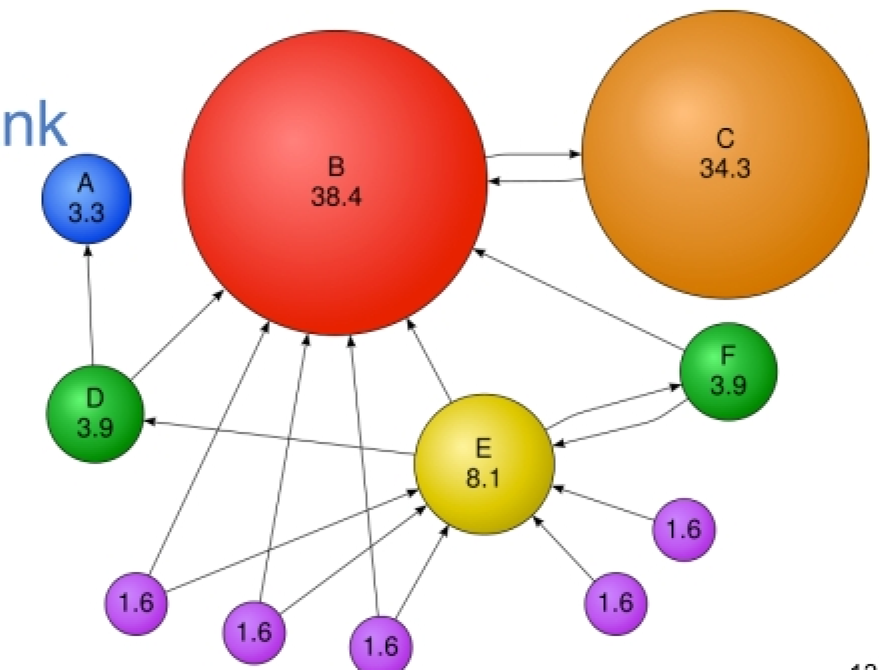


# Google -- The PageRank Algorithm

- U.S. Patent 6,285,999 (original paper 1998)
- PageRank measures the relative importance of a webpage
- Google search engine is much more than this
    - …and the ranking algorithm surely is altered by now
- Algorithm:
    1. Each page gets an initial rank
    2. Alter each rank using:

$$PR(u) = \sum_{v \in B_u} \frac{PR(v)}{L(v)}$$

    3. Repeat….




u = web page
v = web page
PR(v) = rank of page v
L(v) = # links out of v

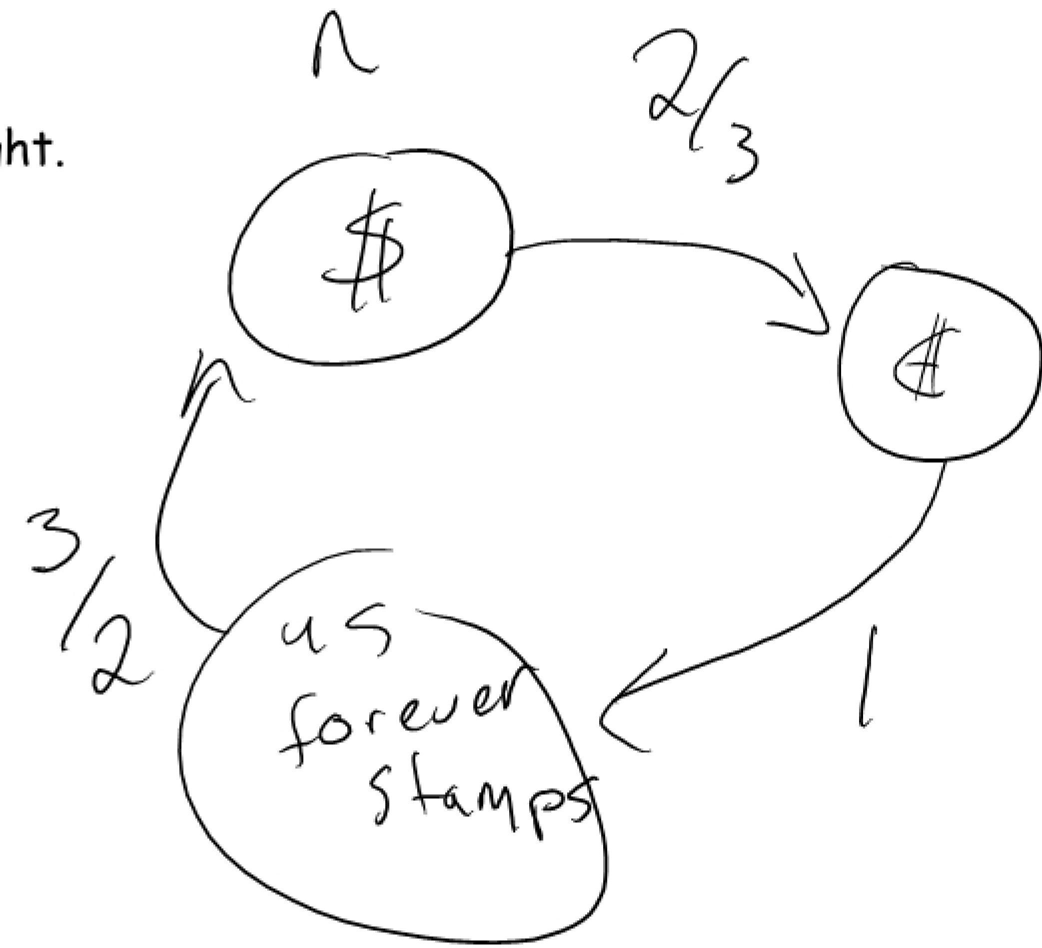
# How else can I get rich?

## Arbitrage!

- Model currencies as vertices in a graph
- Each edge has a weight
- Moving an amount along an edge multiplies it by the edge weight.

- What should happen if you move money along along a cycle?

what if that 1.57

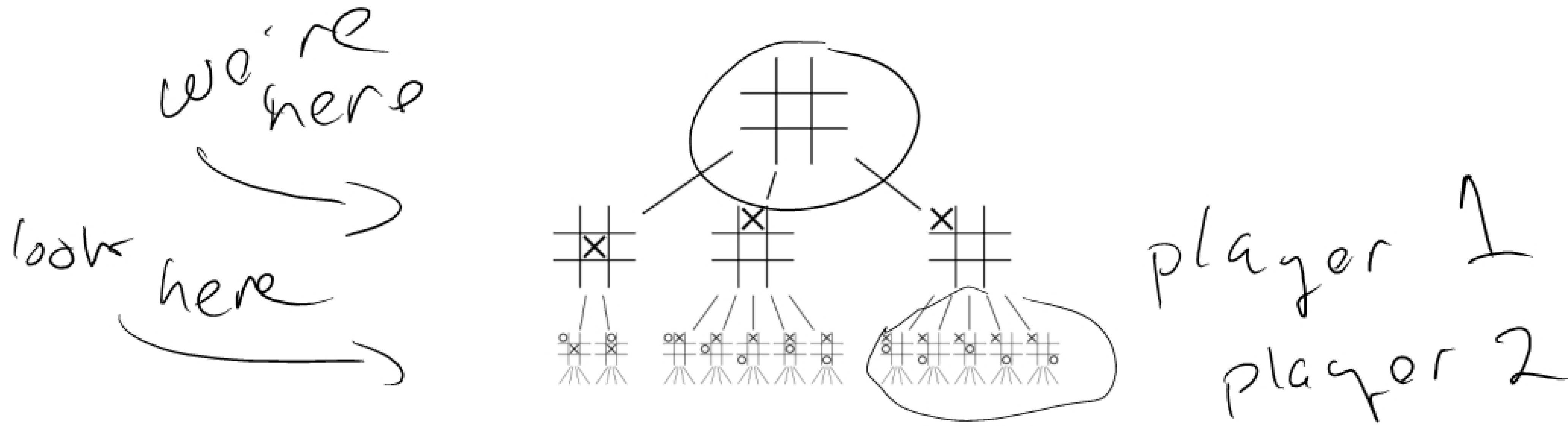



Arbitrageurs exploit market inefficiencies

# Game Trees

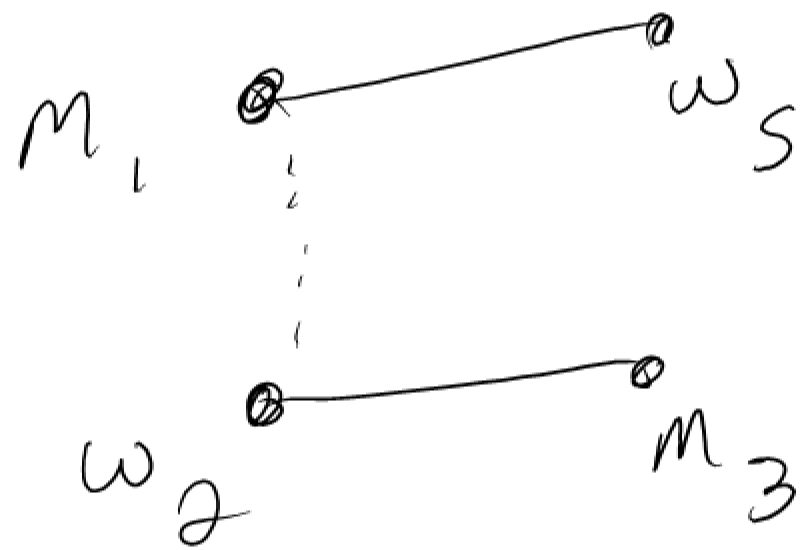
So you want to write a game playing AI for the iphone (or whatever)?

Let's consider tic-tac-toe (game tree with 26,830 leaves)



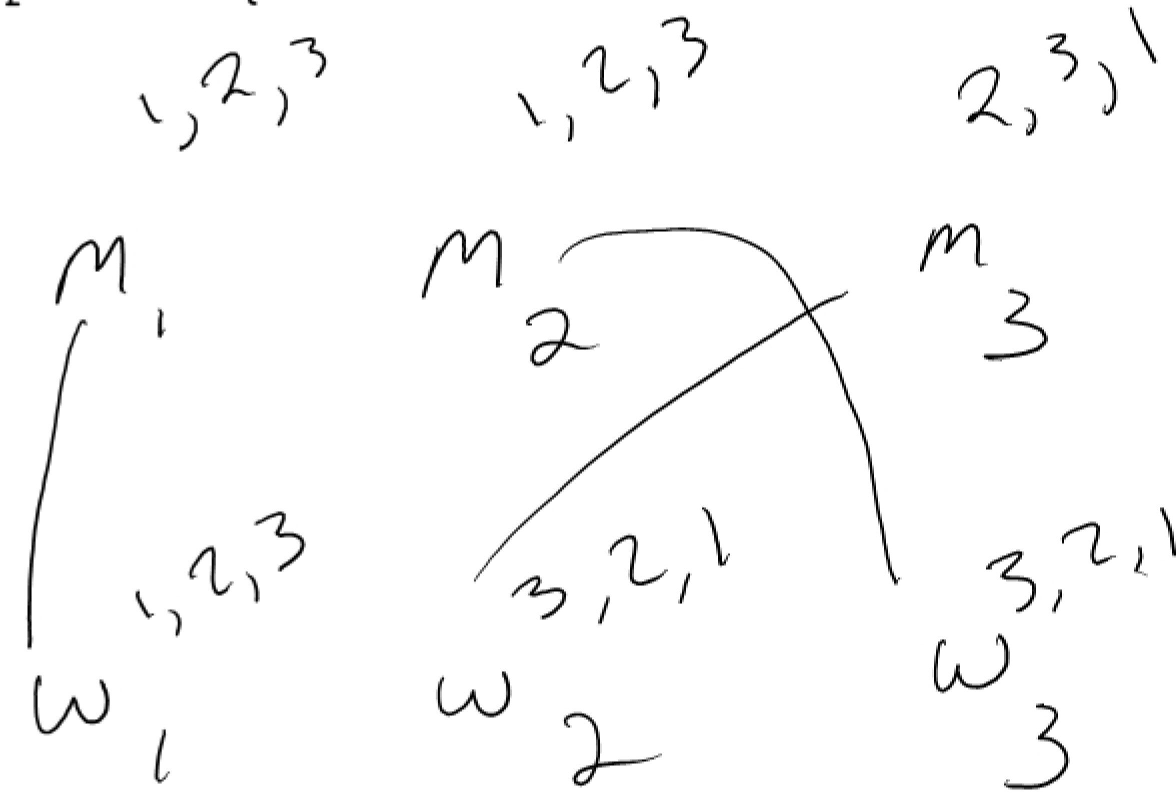
# "College admissions and stability of marriage"

- Paper by Gale and Shapley [1962]
- We have n men and n women
- Each person ranks their possible marriage partners
- A stable matching will a pair man and a woman such that
  - No two people prefer each other to their partners



# "College admissions and stability of marriage"

```
function stableMatching {
    Initialize all m ∈ M and w ∈ W to free
    while ∃ free man m who still has a woman w to propose to {
        w = m's highest ranked such woman
        if w is free then (m, w) become engaged
        else some pair (m', w) already exists
             if w prefers m to m' then
                (m, w) become engaged
                 m' becomes free
             else
                (m', w) remain engaged
    }}
```

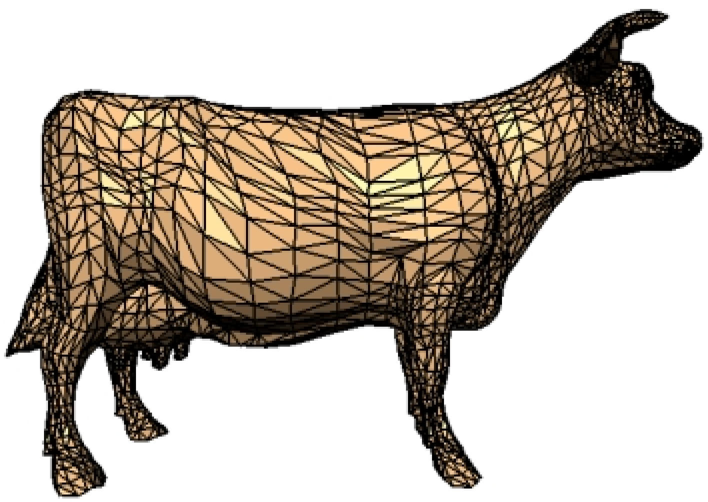


# "College admissions and stability of marriage"

- Properties of the algorithm
  1. Everyone gets matched
  2. Matching is stable
- If men do the asking:
  - Every man is paired with his highest ranked feasible partner.
  - Each female is paired with her lowest ranked feasible partner.

# Processing 3D models

- Digital 3D models are like blueprints
- Often, a 3D surface is described by a triangulated mesh
- List of points in 3D space
- List of triangles specifying connectivity



v 0.3 1.1 0.2
v 0.4 0.4 0.5
v 1.4 0.8 1.2
v 0.9 0.5 0.7
v 1.0 0.1 1.1

f 2 4 1
f 2 5 4
f 3 1 4
f 4 5 3

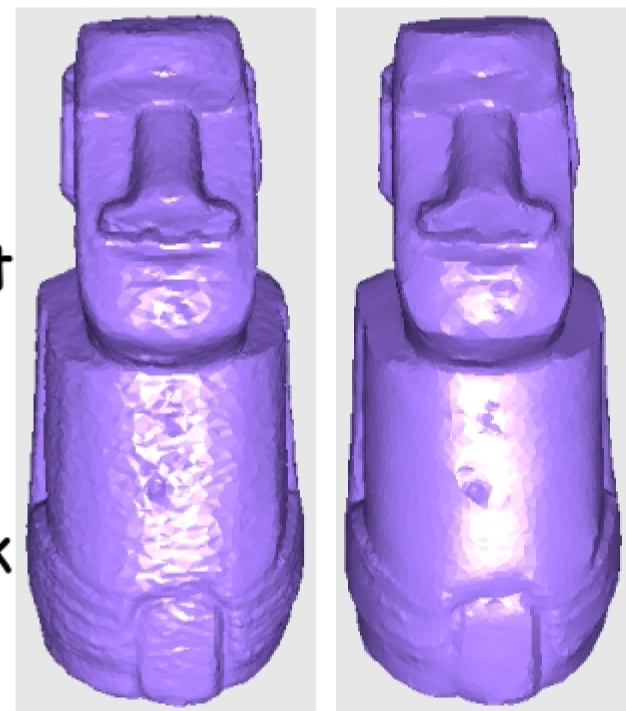
# Processing 3D models

- Meshes often need to be "preprocessed"
- Meshes acquired by 3D scanners need

  holes filled

  noise removed

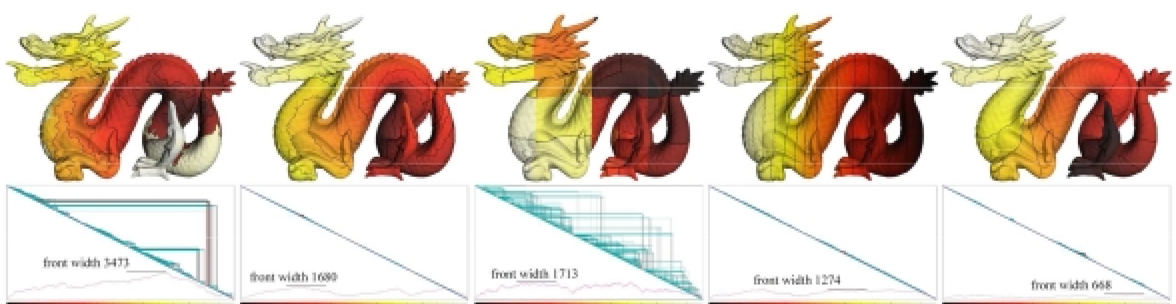
  coarsening
- For really big meshes indexed format is problematic
- If the mesh has to be read from disk lack of locality is makes things slow



# Graph Theory to the Rescue

- One can mix the vertex and triangle data into a single list
- Called a "streaming" format [Isenburg, Lindstrom 2005]
- The question how to order the data to avoid big jumps
- Solve the Minimum Linear Arrangement problem

```
f 2 4 1
f 2 5 4
f 3 1 4
v 0.3 1.1 0.2
v 0.4 0.4 0.5
f 4 5 3
v 1.4 0.8 1.2
v 0.9 0.5 0.7
v 1.0 0.1 1.1
```



# Unsolicited Advice

On the final exam

1. If a question is confusing, state what you are assuming it is asking for (e.g. number of ways to choose a playing card -- "with or without replacement").
2. I would answer what I know first,
   and answer the difficult questions later.
3. Know the terminology
4. Studying -- read the book
5. What do the instructors think is important?
   Look over the HWs, MTs, quizzes, and topic lists
   What was the topic we spent the most time on?
6. Write clear proofs -- use sentences, different letters for different variable, explain what your steps are accomplishing.

# Unsolicited Advice-- extended and remixed

1. Make sure you want to do what you think you want to do
   1. Get a summer internship if possible:

      https://www.informatics.uiuc.edu/display/jobopps/Internships
   2. If you want to be a programmer, spend time programming

2. If you are interested in graduate school and/or research
   1. Undergraduate Research Lab - CS 498LA

      https://agora.cs.uiuc.edu/display/cs498la/Home
   2. A number of professors hire undergrads:
      https://agora.cs.uiuc.edu/display/undergradProg/Undergraduate+Research+Opportunities