Lecture 11

CSE 331

Please have a face mask on

Masking requirement

UB requires all students, employees and visitors – regardless of their vaccination status – to wear face coverings while inside campus buildings.

https://www.buffalo.edu/coronavirus/health-and-safety/health-safety-guidelines.html

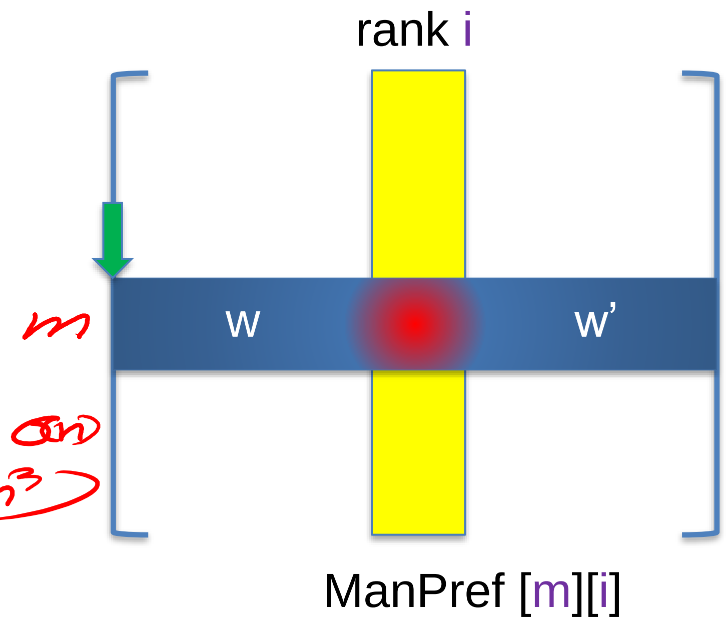
Answering Q4

Q4: Is w' > w in L_m?

A4: Scan ManPref[m] and figure out the ranks of i' and i ~~or w~~ of w' and w; check if i' < i.

Overall GS runtime: $O(n^3)$

$O(n^2) \cdot O(n) = O(n^3)$



(4) How do we decide if m prefers w' to w?

Q4 — Q3 :

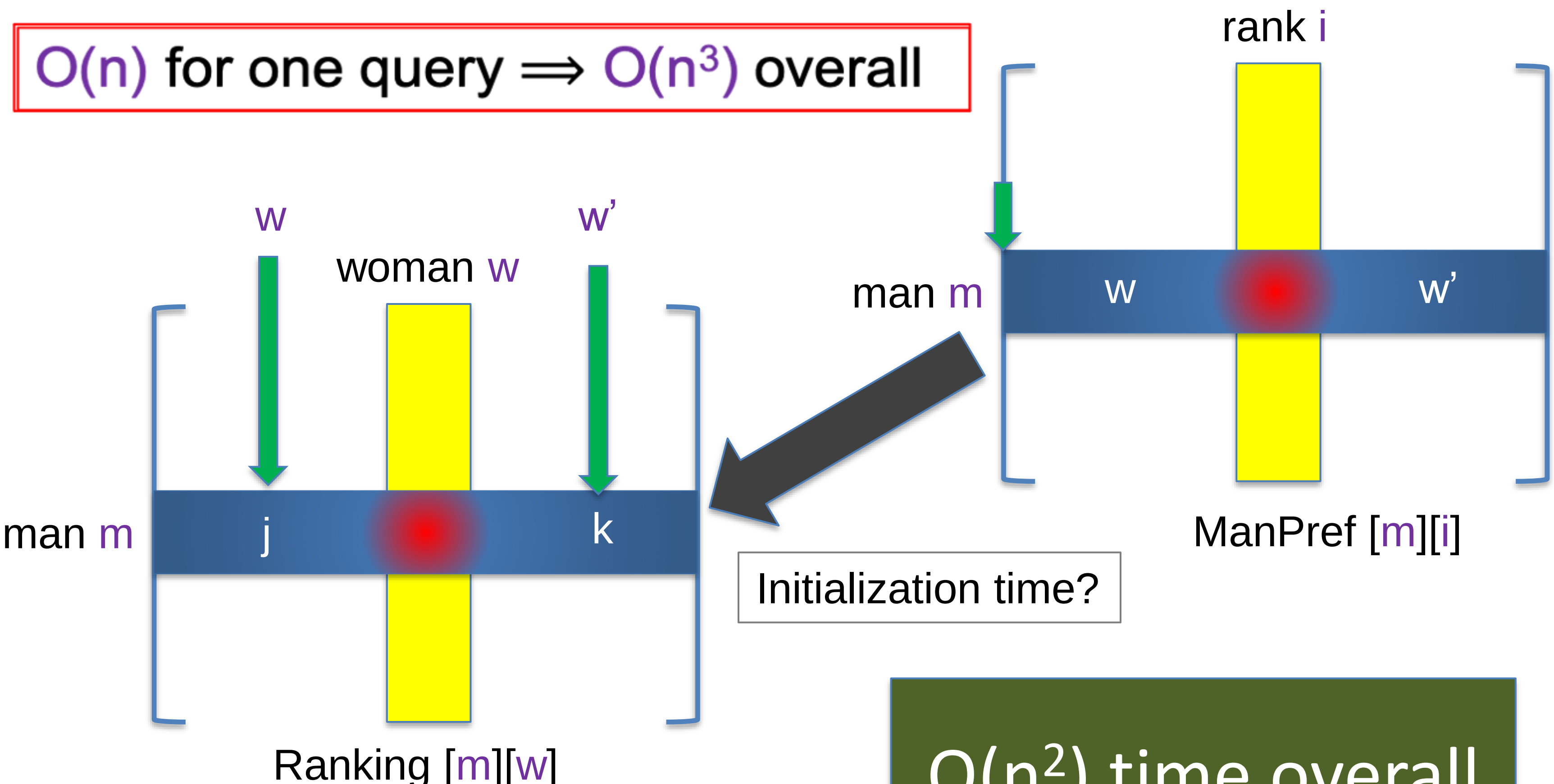
Init :

$O(n) + O(n) + O(n)$
$= O(n)$

Query/update : $O(1)$

Answering Q4

O(n) for one query $\Longrightarrow$ O(n^3) overall



O(1) query time

O(n^2) time overall

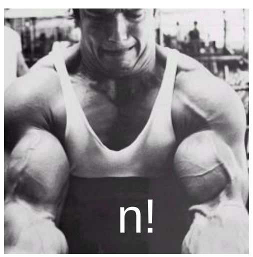
(4) How do we decide if m prefers w' to w?

Main Steps in Algorithm Design

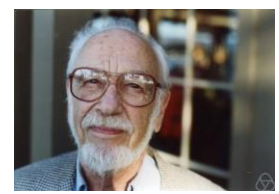
Problem Statement

NRMP
National Resident Matching Program

Problem Definition



Algorithm

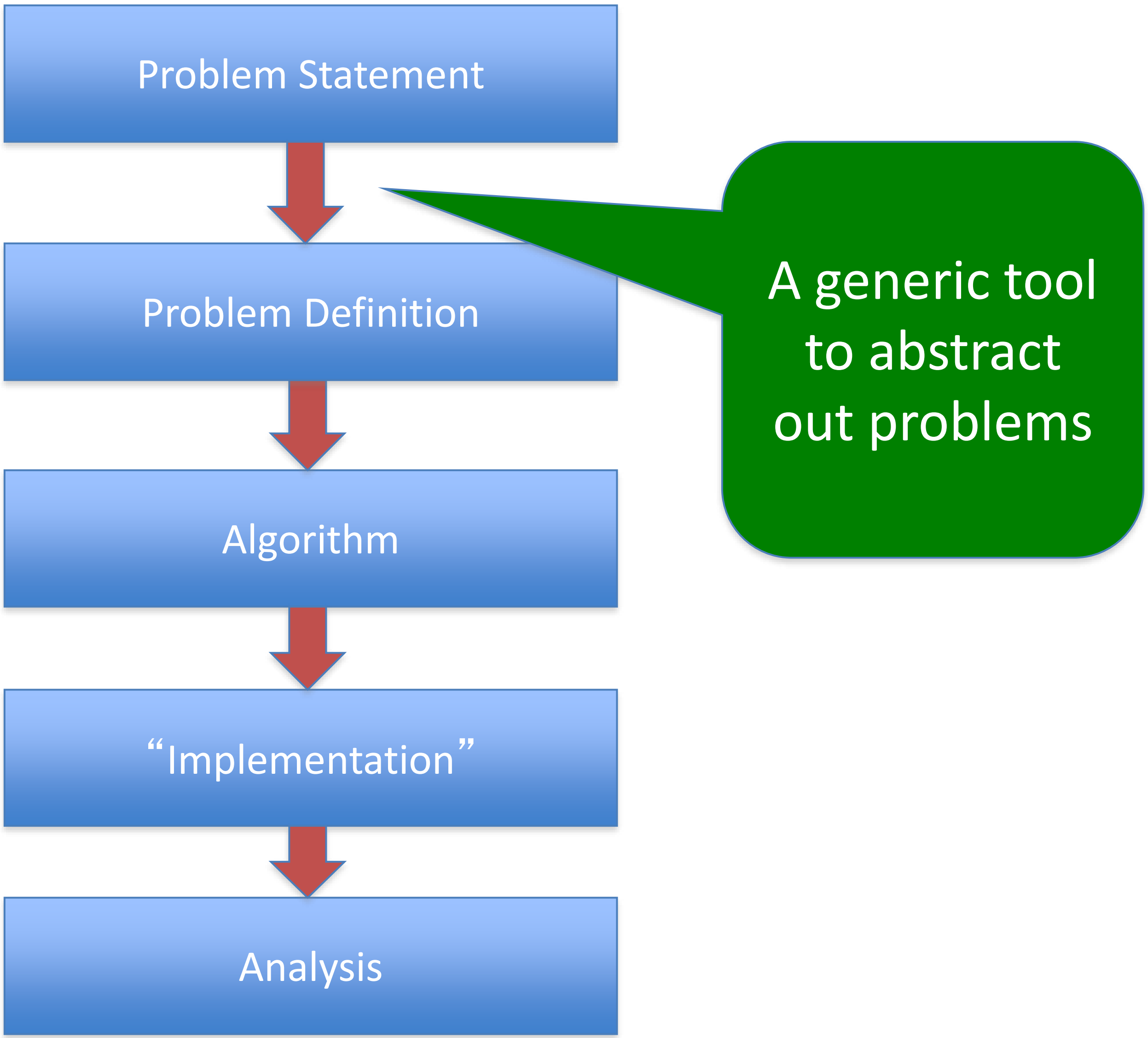


“Implementation”

Analysis

Correctness Analysis

Up Next….

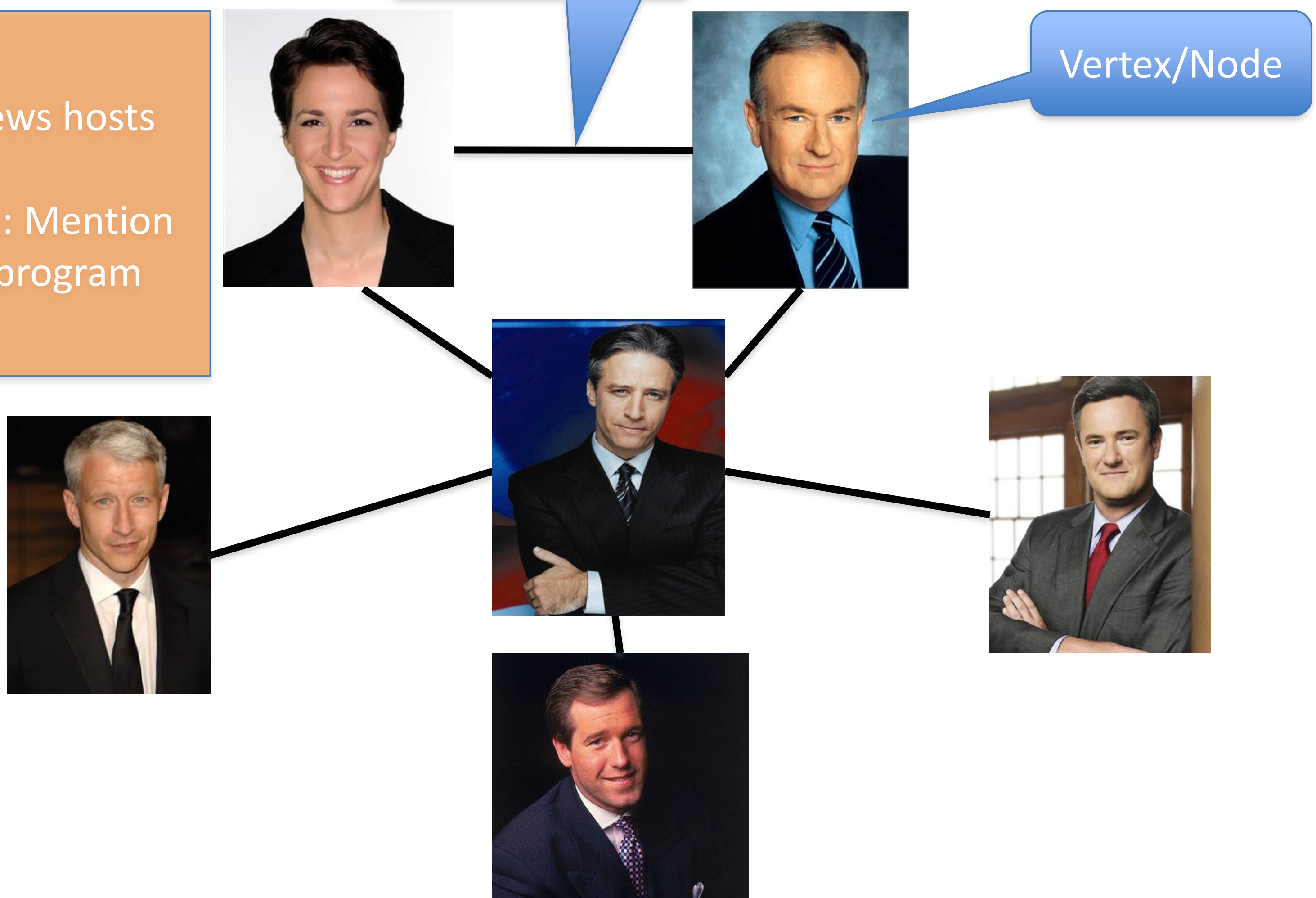

Problem Statement
Problem Definition
Algorithm
“Implementation”
Analysis
A generic tool to abstract out problems

Graphs

Representation of relation Edge airs of entities/elements

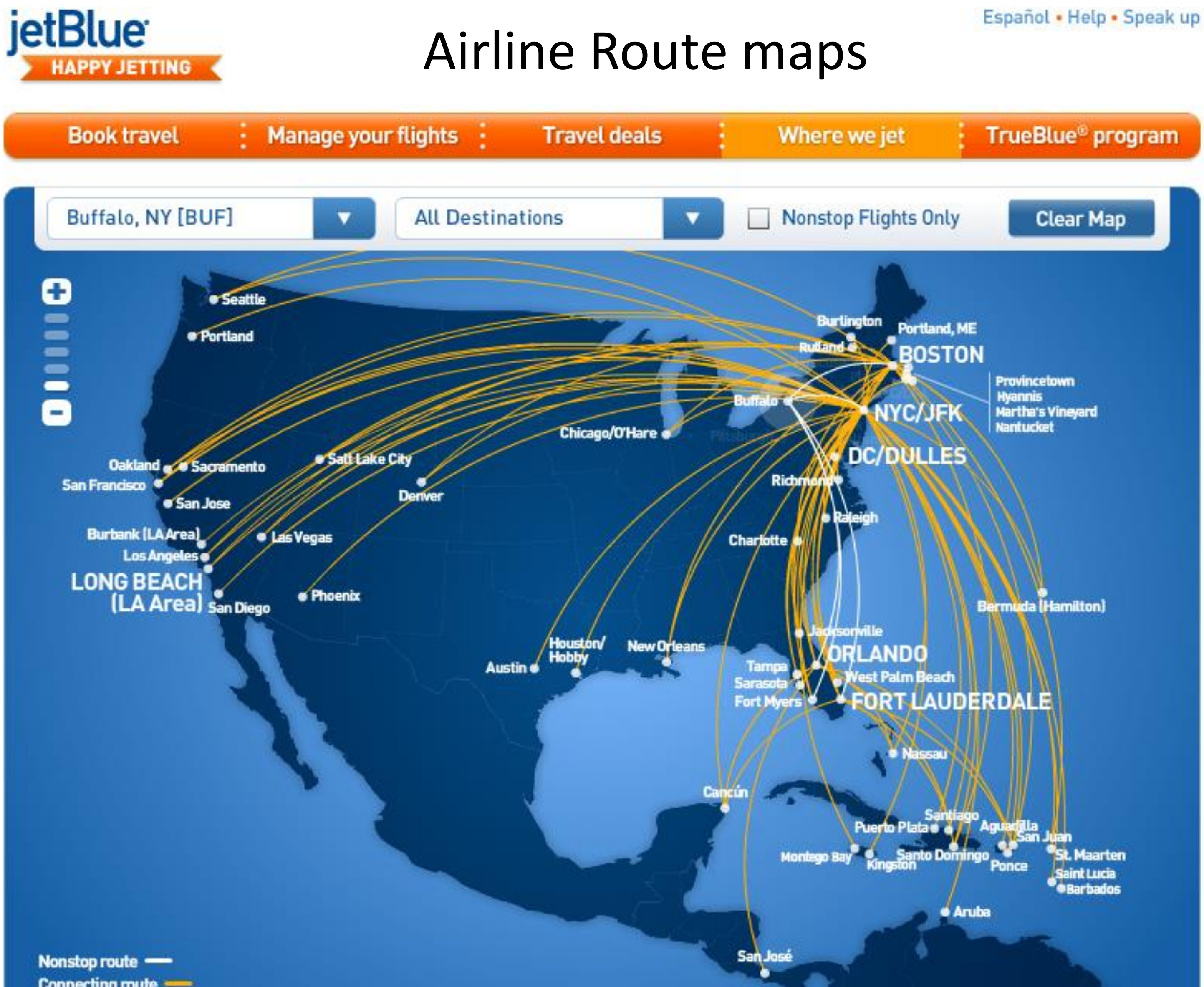
Entities: News hosts

Relationship: Mention in other's program

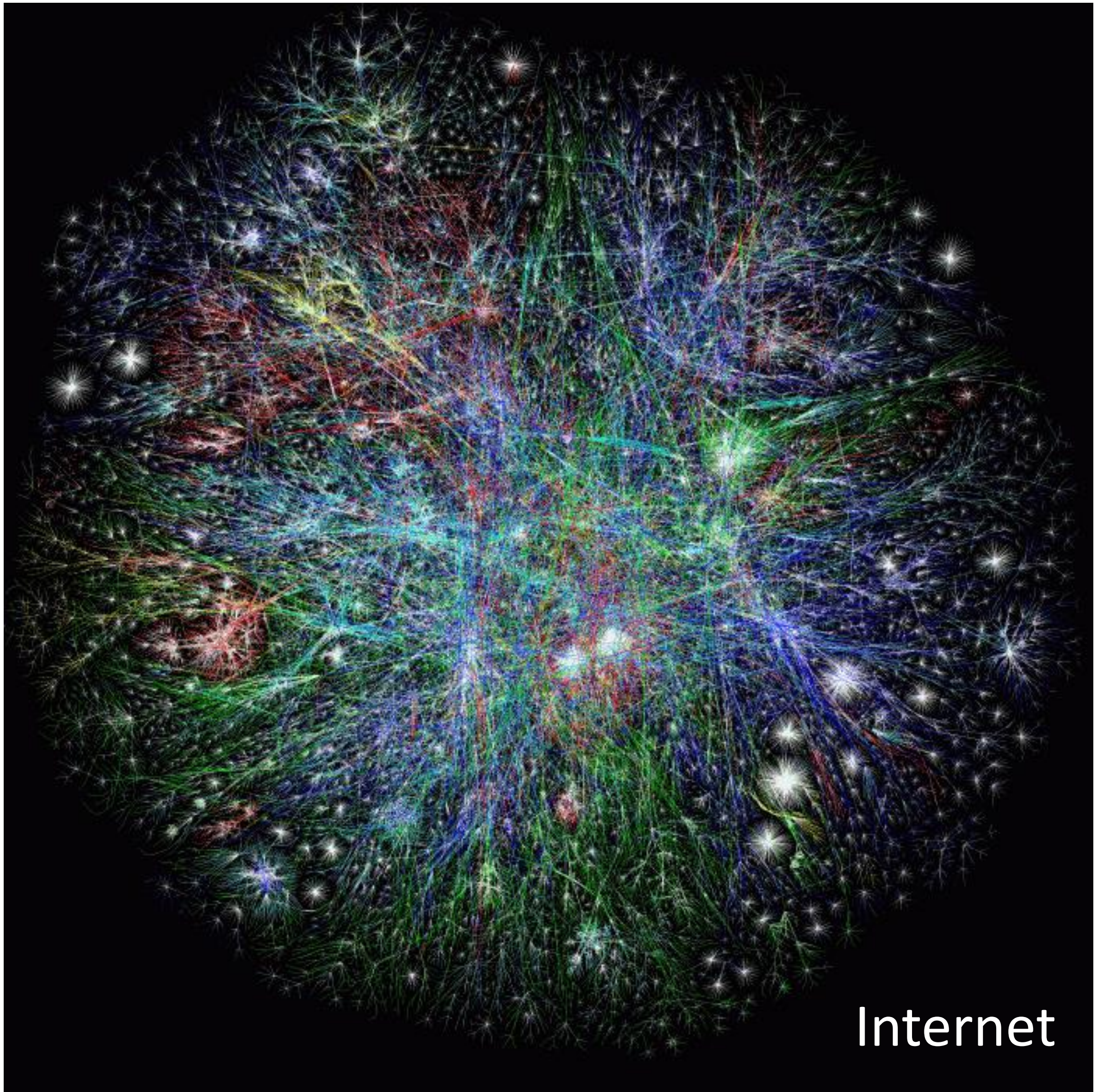


Graphs are omnipresent

Airline Route maps

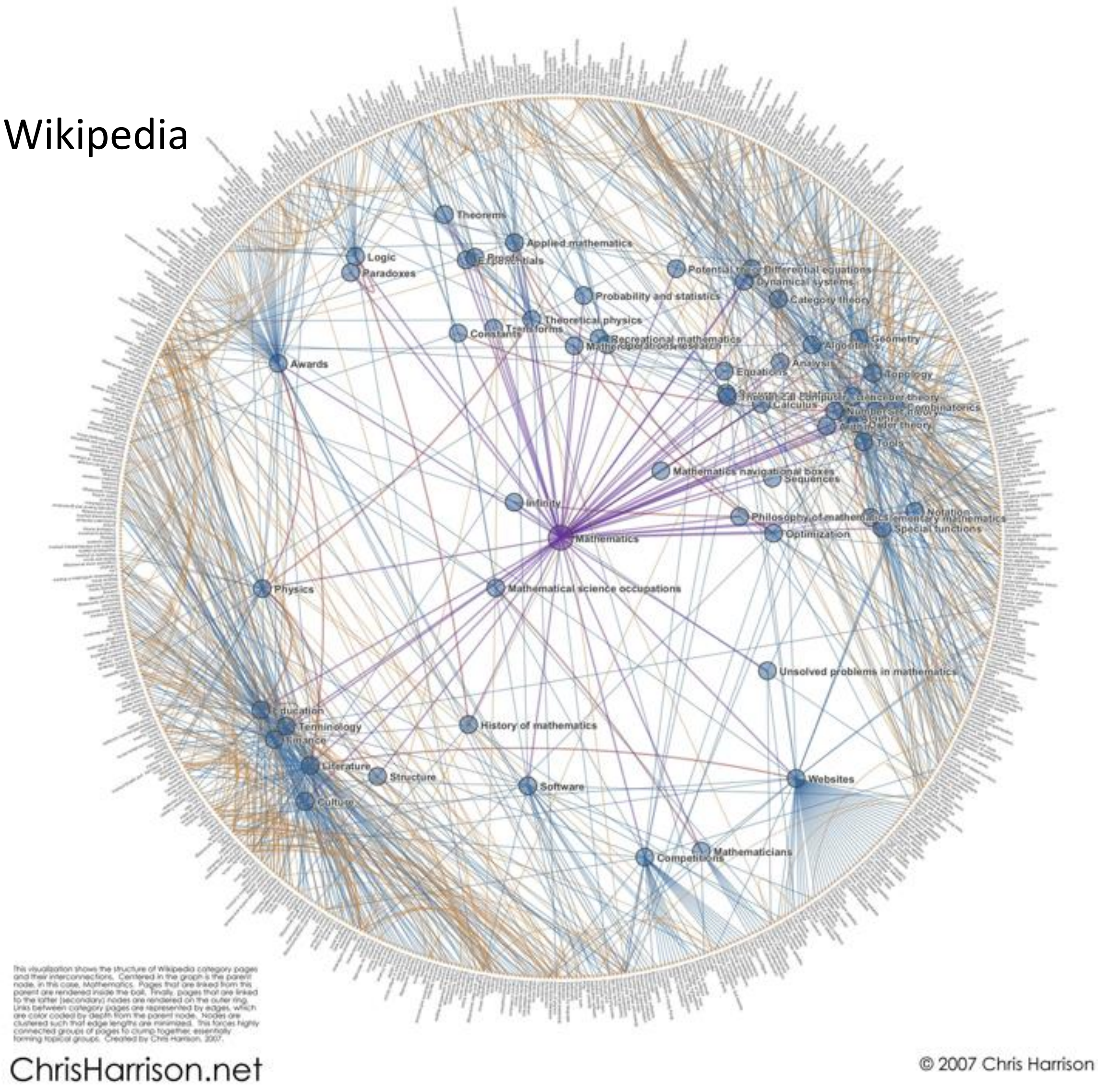


What does this graph represent?

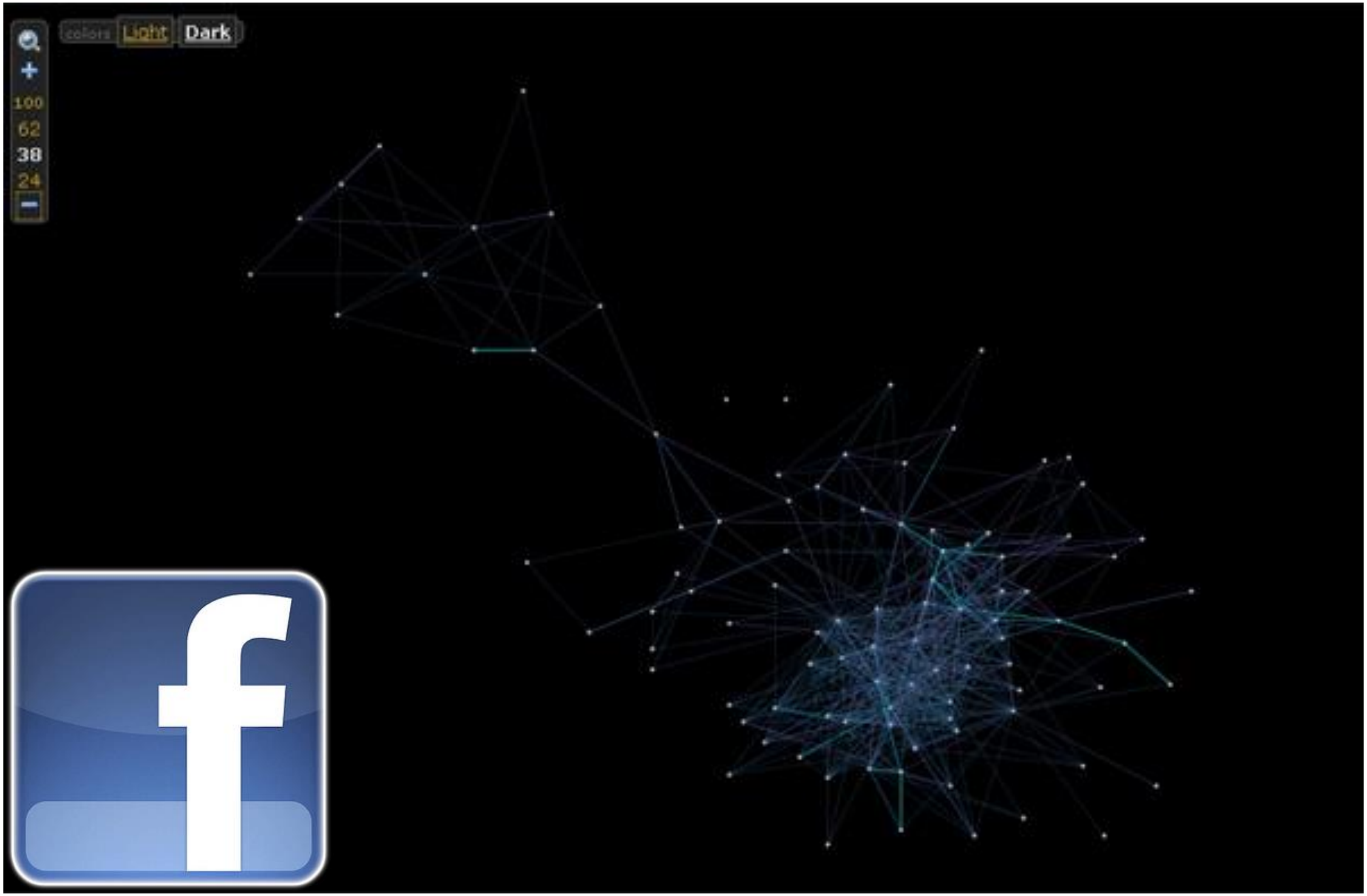


And this one?

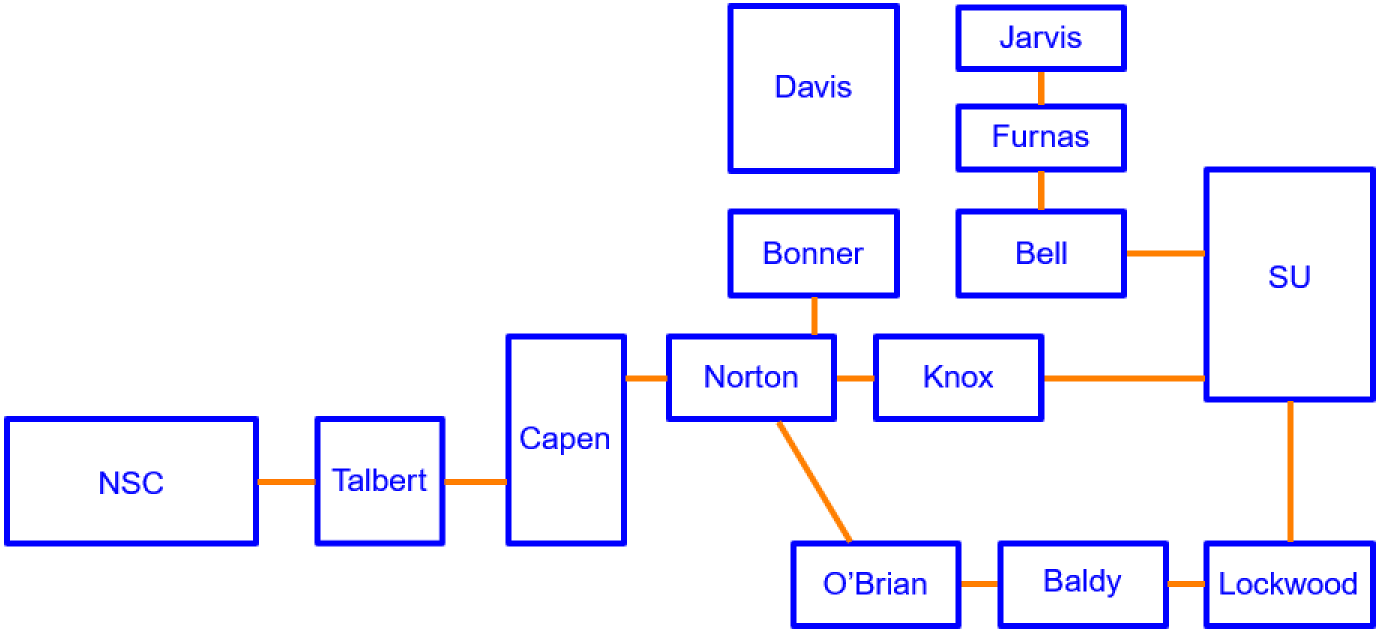
Math articles on Wikipedia



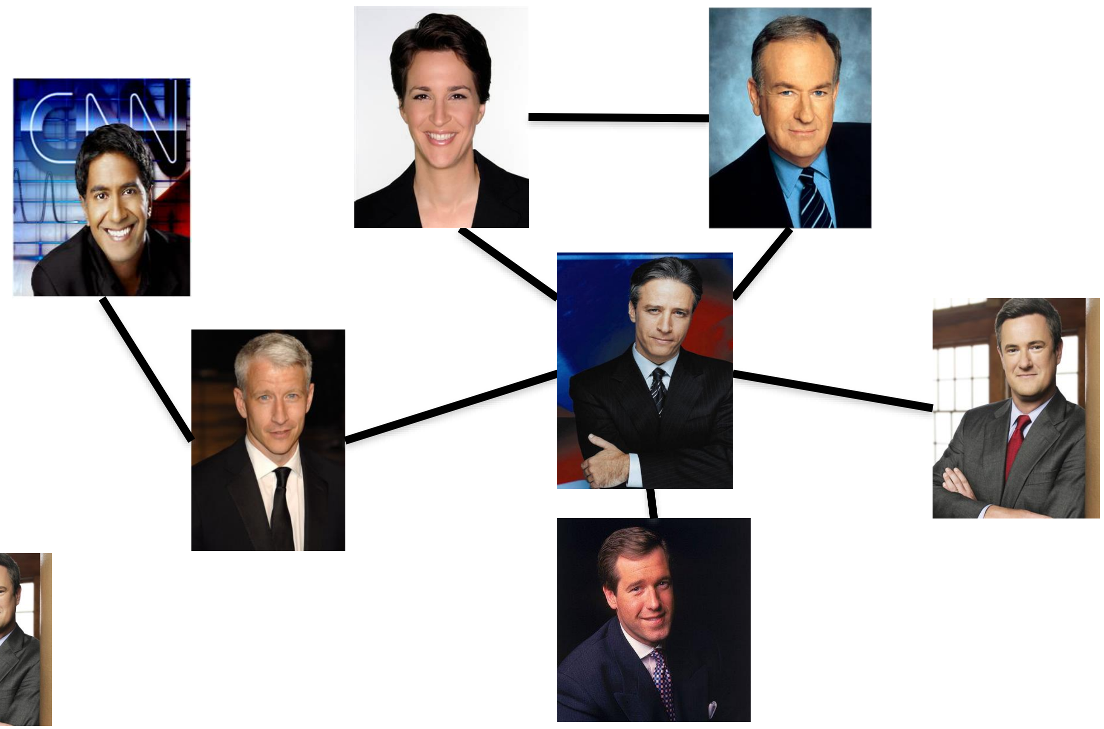
And this one?



Buildings on North Campus connected by tunnels



Paths



Connected

Sequence of vertices connected by edges

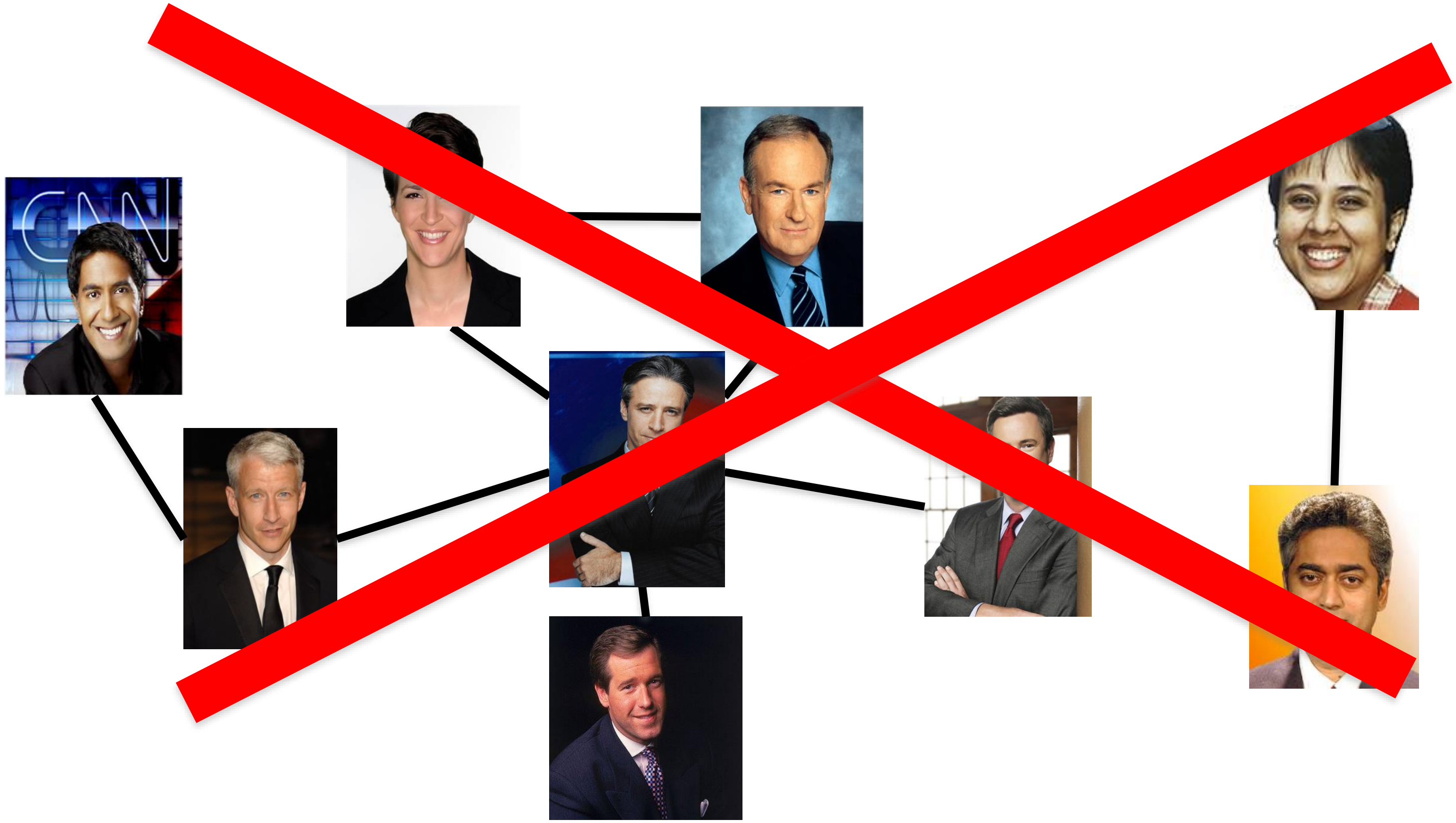
Path length 3

Connectivity

u and w are connected iff there is a path between them

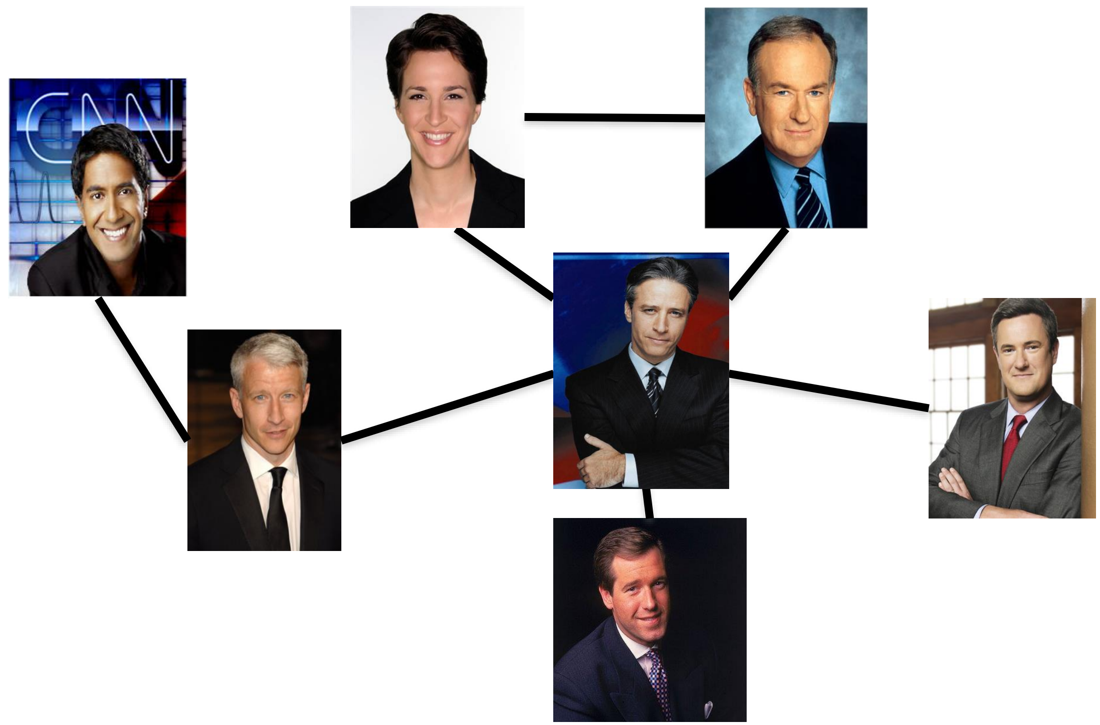
A graph is connected iff all pairs of vertices are connected

Connected Graphs



Every pair of vertices has a path between them

Cycles

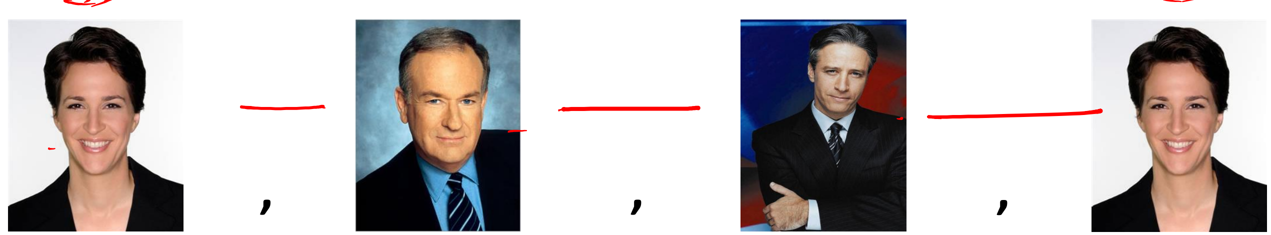


$k=4$

$\underline{k-1} = 3$

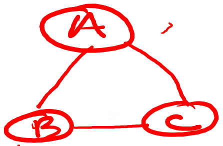
Sequence of k vertices connected by edges, first k-1 are distinct

1 1



Basic Graph definitions

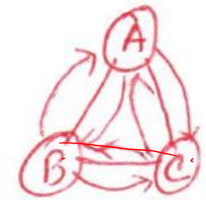
Graphs



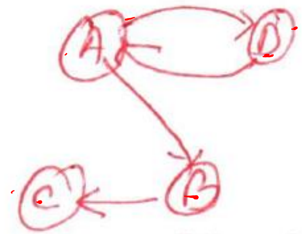
Graphs

$G = (V, E)$ $\quad E \subseteq V \times V$

Set of vertices — Set of edges



Undirected



directed

Defaults: $|V| = n$; $|E| = m$

$V = \{A, B, C\}$

$E = \{(A,B), (B,C), (C,A), (B,A), (C,B), (A,C)\}$

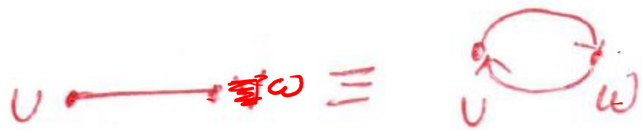
$V = \{A, B, C, D\}$

$E = \{(A,B), (B,C), (A,D), (D,A)\}$

$n = 4$

$m = 4$

Def: G is undirected $\iff$ (iff) $\forall\, u \neq w \in V,\ (u,w) \in E \iff (w,u) \in E$



Q: (i) Airline map (undirected) (ii) Wikipedia articles map (directe

Graphs

Default: G is undirected.

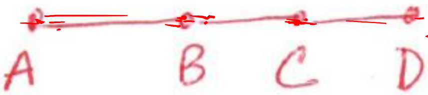
Claim: Every undirected graph is also a directed graph

Pf. idea: replace every

Graphs

A   C → D

Paths:



is path?

→ D, C, B, A ✓	3
A, B, C, D ✓	3
A, B, C, B ✓	3
A, C, D ✗	

D, C, ~~B~~ B, A ✗

A, B, C, D ✓

A, B, C, B ✗

A, C, D ✗

A   C → D

$(u_1, u_2) \in E$

$u_1, u_2, \dots, u_n$

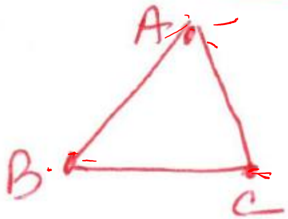
$(u_{n-1}, u_n) \in E$

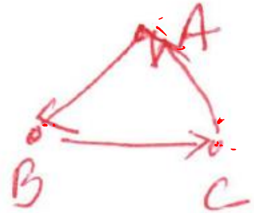
Def: A path in (directed) $G = (V, E)$ is a sequence of vertices $u_1, \dots, u_K$ {$u_1 - u_K$ path} s.t. $\forall\, i \in [K-1] = \{1, \dots, K-1\}$ $(u_i, u_{i+1}) \in E$

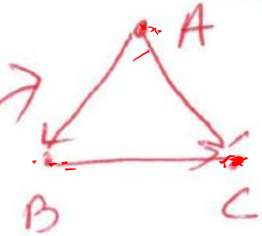
Notes: (i) u_i need not be distinct (ii) holds for directed G

Graphs

Cycles:







cycle? : A, C, B, A ✓
A, B, C, A ✓

A, C, B, A ✗
A, B, C, A ✓

A, C, B, A ✗
A, B, C, A ✗

Directed acyclic graph (DAG)

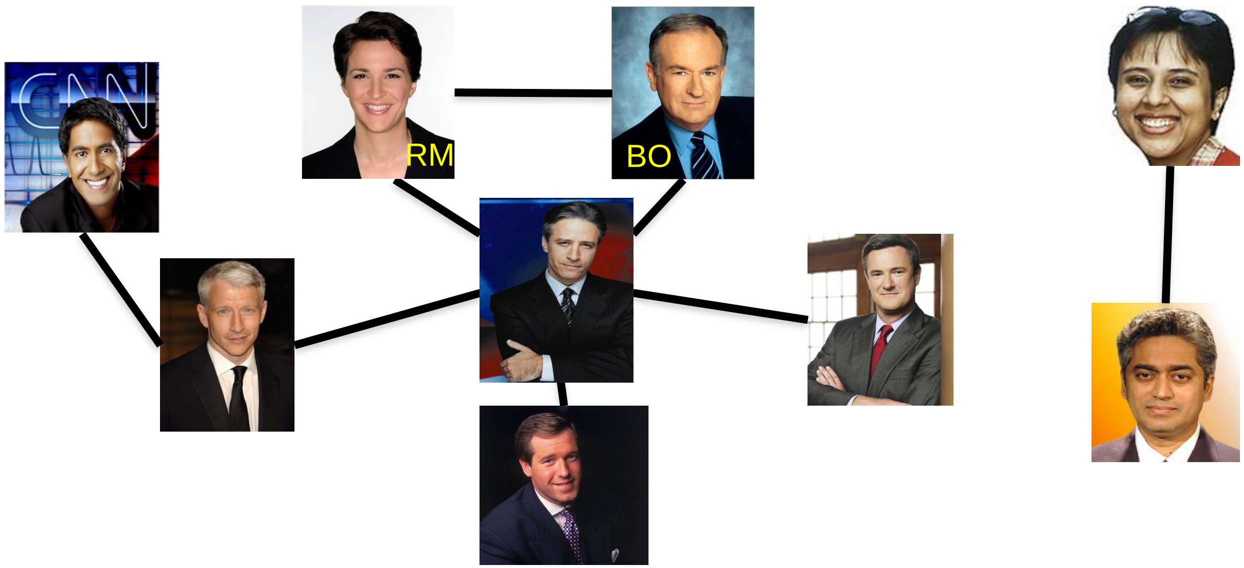
Def: A cycle is a sequence

$v_1, v_2, \dots, v_k = v_1$ $v_1, \dots, v_{k-1}$ are distinct

$\forall i \in [k-1]$, $(v_i, v_{i+1}) \in E$

Distance between u and v

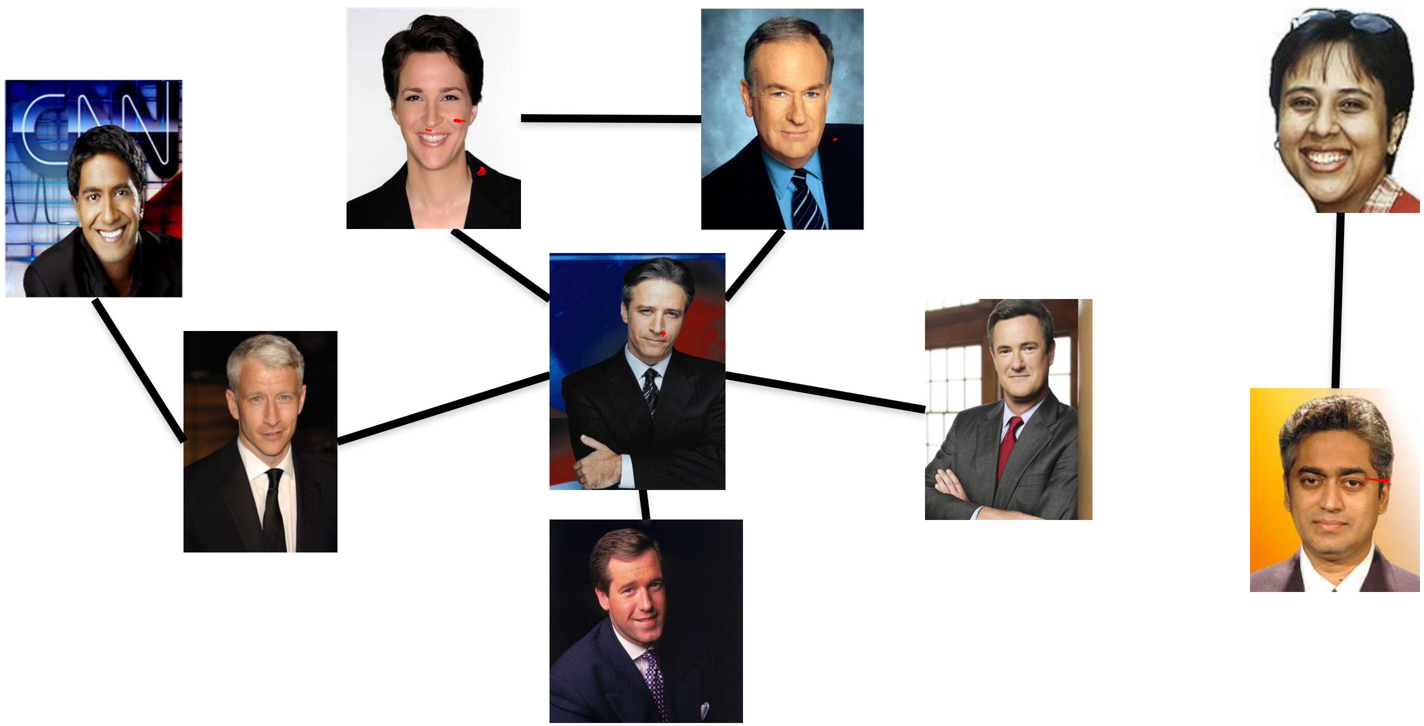
Length of the shortest length path between u and v



Distance between RM and BO? 1

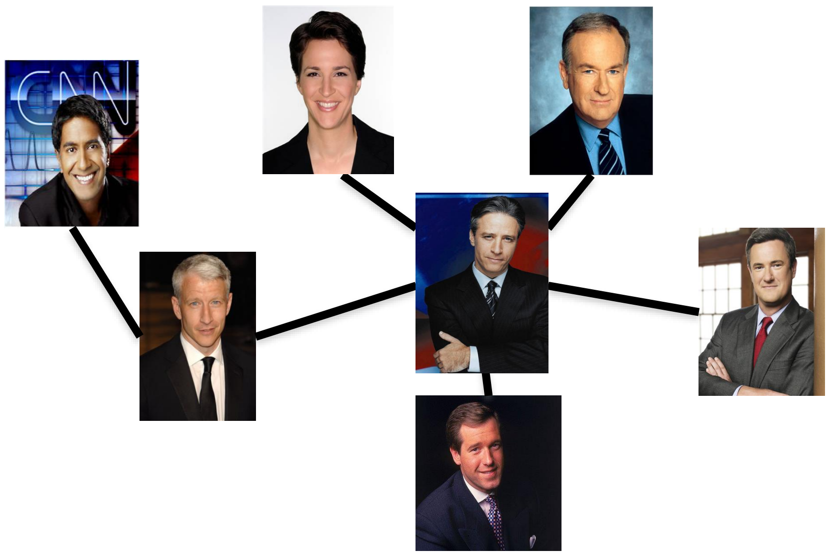
Tree

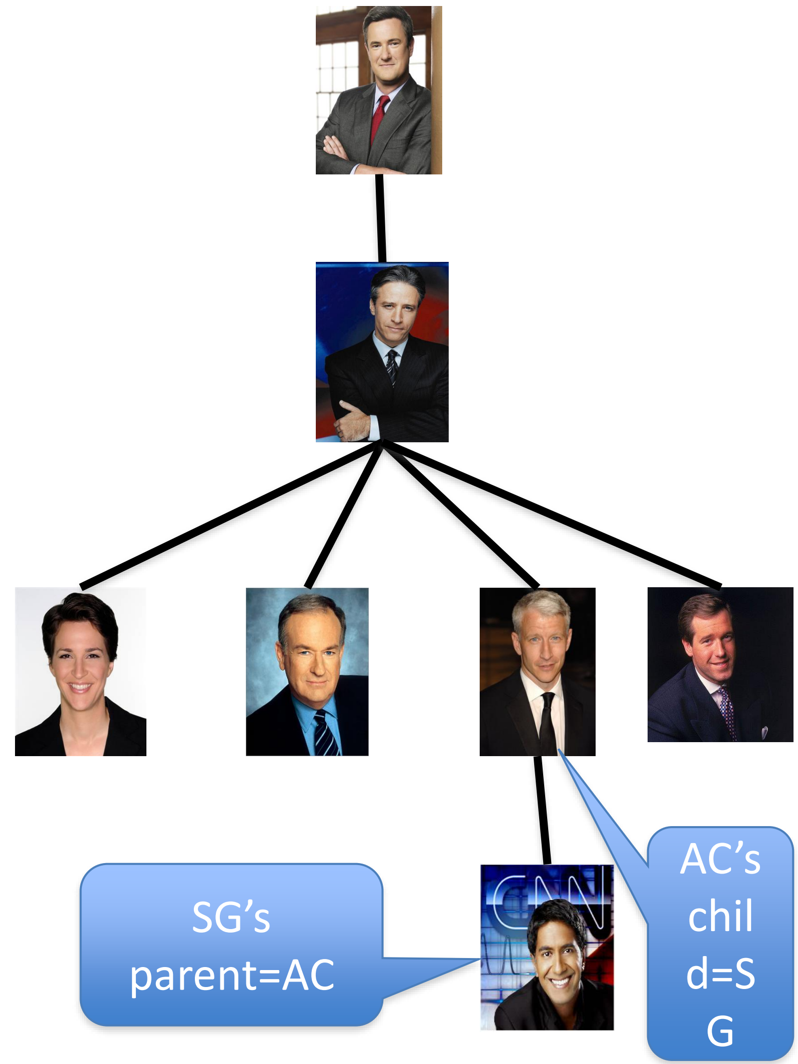
Connected undirected graph with no cycles



Rooted Tree

A rooted tree





Pick any vertex as root

Let the rest of the tree hang under “gravity”

Every n vertex tree has n-1 edges

Trees

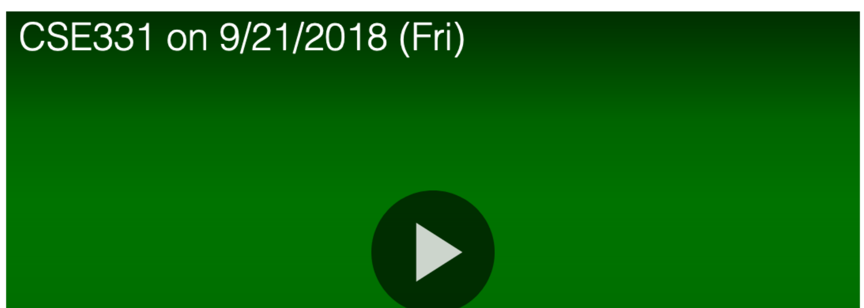
This page collects material from previous incarnations of CSE 331 on trees, especially the proof that trees with n nodes have exactly $n-1$ edges.

Where does the textbook talk about this?

`Section 3.1` in the textbook has the lowdown on trees.

Fall 2018 material

Here is the lecture video:



Every n vertex tree has n-1 edges

Let G be an undirected graph on n nodes

Then ANY two of the following implies the third:

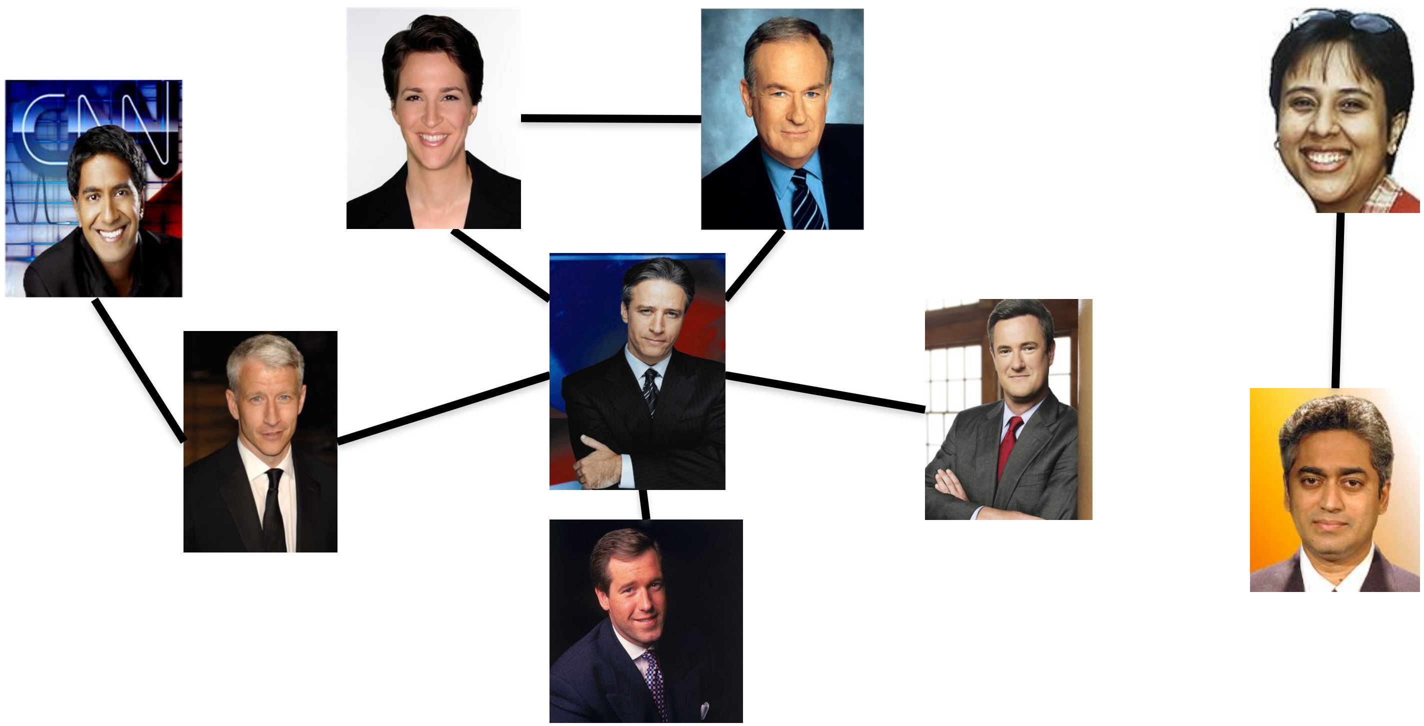
T is connected

T has no cycles

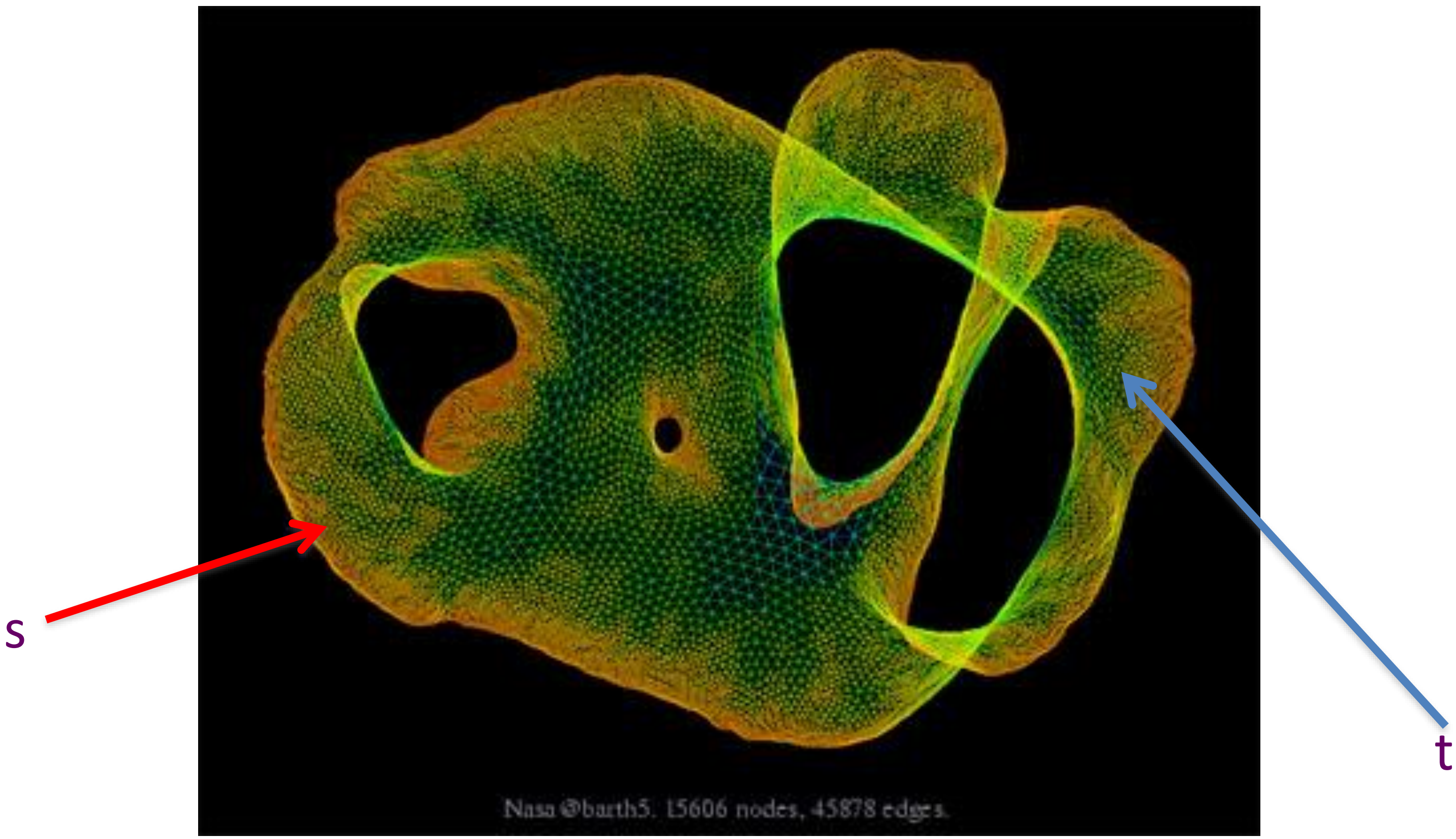
T has n-1 edges

Algorithms for checking connectivity

Checking by inspection



What about large graphs?



Are s and t connected?

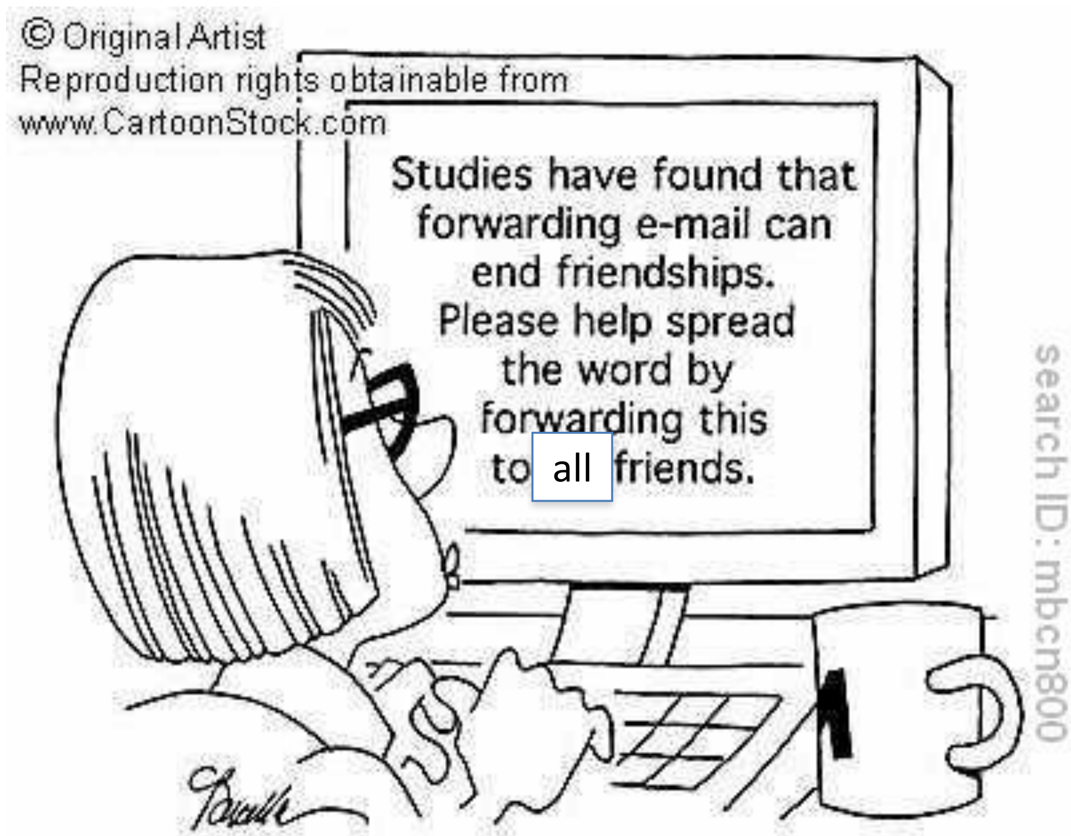
Brute-force algorithm?

List all possible vertex sequences between s and t

n^n such sequences

Check if any is a path between s and t

Algorithm motivation



Breadth First Search (BFS)

BFS via examples

In which we derive the breadth first search (BFS) algorithm via a sequence of examples.

Expected background

These notes assume that you are familiar with the following:

- Graphs and their representation. In particular,
  - Notion of connectivity of nodes and connected components of graphs
  - Adjacency list representation of graphs
  - Notation:
    - $G = (V, E)$
    - $n = |V|$ and $m = |E|$
    - $CC(s)$ denotes the connected component of s
- Trees and their basic properties

The problem

In these notes we will solve the following problem:

Connectivity Problem

Input: Graph $G = (V,E)$ and s in V

Output: All t connected to s in G

Connected component of s

Breadth First Search (BFS)

Build layers of vertices connected to s

$L_0 = \{s\}$

Assume $L_0,..,L_j$ have been constructed

L_{j+1} set of vertices not chosen yet but are connected by an edge to L_j

Stop when new layer is empty

BFS Tree

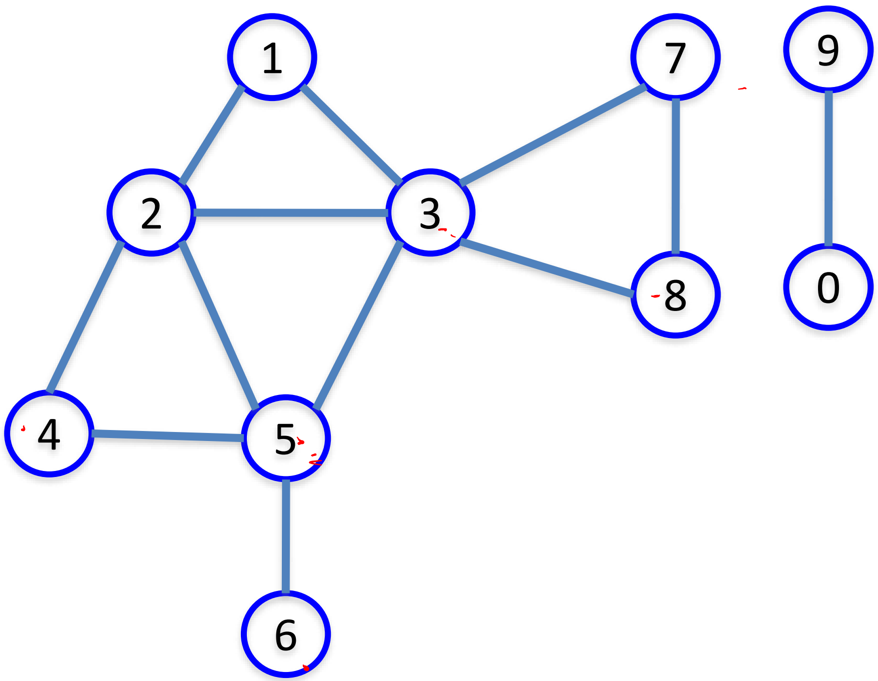
L_j

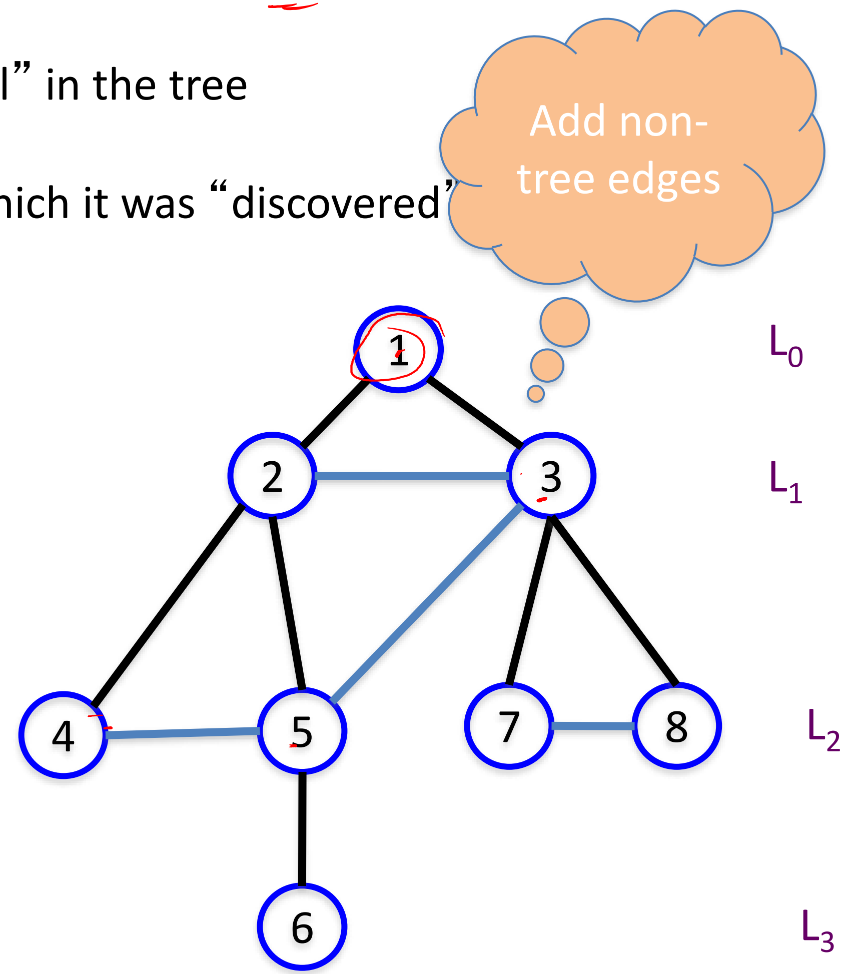
L_{j+1}

BFS naturally defines a tree rooted at s

L_j forms the jth "level" in the tree

u in L_{j+1} is child of v in L_j from which it was "discovered"





Computing Connected component

PROPOSITION :

Proposition : let T be a BFS tree for $G=(V,E)$.

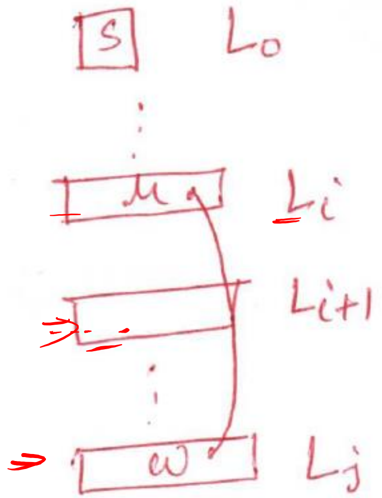
If $(u,w) \in E$ s.t. $u \in L_i$, $w \in L_j$

$\Rightarrow |i-j| \leq 1 \iff i \in \{j-1, j, j+1\}$

Pf (idea): WLOG (without loss of generality) assume $i \leq j$ [if $i > j$, change the roles of i & j]

For contradiction assume $|i-j| > 1 \Rightarrow j > i+1$

$\Rightarrow j \geq i+2$



Consider the situation when BFS is creating L_{i+1}

$\Rightarrow u \in L_i,\ w \notin L_0, \ldots, L_i$

$\rightarrow (u,w) \in E$

$\Rightarrow w$ will be added by BFS to L_{i+1}

$\Rightarrow$ contradicts $w \in L_j$ for $j \geq i+2$ ∎