A good picture is worth a thousand words

- Expressive power is the first explanation for a success of graphs
- More claims for graphs come later
- Example for a title above follows!

Greek pre-Socratic Philosophers

- Thales of Miletus influenced Anaximander, Pythagoras, Heraclitus and Anaximenes of Miletus
- Anaximander infl. Pythagoras
- Pherecides of Syros infl. Pythagoras
- Anaximander infl. Heraclitus
- Pythagoras infl. Heraclitus
- Pythagoras infl. Empedocles
- Pythagoras infl. Philolaus
- Pythagoras infl. Archytas
- Pythagoras infl. Alcmaeon of Croton
- Philolaus infl. Archytas
- Heraclitus infl. Parmenides
- Parmenides infl. Democritus
- Democritus infl. Philolaus
- Parmenides infl. Melissus of Samos
- Parmenides infl. Socrates
- Leucippus infl. Democritus, (and this is about 60% of the story)

Question for you:

Did Heraclitus infl. Archytas?

Next question:

Did Pythagoras infl. Melissus of Samos?
The whole Greek Pre-Socratic Philosopher in GRAPH, and same questions for you now:
Did Heraclitus infl. Archytas? Did Pythagoras infl. Melissus of Samos?

§ 9.1 What are Graphs?
- General meaning in everyday math:
  A plot or chart of numerical data using a coordinate system.
- Technical meaning in discrete mathematics:
  A particular class of discrete structures (to be defined) that is useful for representing relations and has a convenient webby-looking graphical representation.

Applications of Graphs
- Potentially anything (graphs can represent relations, relations can describe the extension of any predicate).
- Apps in networking, scheduling, flow optimization, circuit design, path planning.
- Genealogy analysis, computer game-playing, program compilation, object-oriented design, ...

Simple Graphs
- Correspond to symmetric binary relations \( R \).
- A simple graph \( G=(V,E) \) consists of:
  - a set \( V \) of vertices or nodes (\( V \) corresponds to the universe of the relation \( R \)),
  - a set \( E \) of edges (arcs, links): unordered pairs of (distinct) elements \( u,v \in V \), such that \( uRv \).

Note, in a simple graph there is only ONE EDGE between vertices & no ARROWS & no LOOPS.
Example of a Simple Graph

- Let $V$ be the set of states in the southeastern U.S.:
  - $V=\{\text{FL, GA, AL, MS, LA, SC, TN, NC}\}$
- Let $E=\{\{u,v\}|u \text{ adjoins } v\}$
  - $=\{(\text{FL,GA}), (\text{FL,AL}), (\text{GA,AL}), (\text{GA,SC}), (\text{GA,TN}), (\text{GA,NC}), (\text{AL,MS}), (\text{AL,TN}), (\text{MS,LA}), (\text{MS,TN}), (\text{TN,NC}), (\text{NC,SC})\}$

Multigraphs

- Like simple graphs, but there may be more than one edge connecting two given nodes.
- A multigraph $G=(V, E, f)$ consists of a set $V$ of vertices, a set $E$ of edges (as primitive objects), and a function $f: E \rightarrow \{\{u,v\}|u, v \in V \wedge u \neq v\}$.
- e.g., nodes are cities, edges are segments of major highways.

Pseudographs

- Like a multigraph, but edges connecting a node to itself are allowed.
- A pseudograph $G=(V, E, f)$ where $f: E \rightarrow \{\{u,v\}|u, v \in V\}$. Edge $e \in E$ is a loop if $f(e)=\{u,u\} = \{u\}$.
- e.g., nodes are campsites in a state park, edges are hiking trails through the woods.
Directed Graphs

- Correspond to arbitrary binary relations \( R \), which need not be symmetric.
- A directed graph \((V,E)\) consists of a set of vertices \( V \) and a binary relation \( E \) on \( V \).
- E.g.: \( V = \text{people}, \ E = \{(x,y) \mid x \text{ loves } y\} \)

Directed Multigraphs

- Like directed graphs, but there may be more than one arc from a node to another.
- A directed multigraph \( G=(V, E, f) \) consists of a set \( V \) of vertices, a set \( E \) of edges, and a function \( f:E \rightarrow V \times V \).
- E.g., \( V = \text{web pages}, \ E = \text{hyperlinks}. The \text{WWW is a directed multigraph...} \)

Types of Graphs: Summary

- Summary of the book’s definitions.
- Keep in mind this terminology is not fully standardized...

<table>
<thead>
<tr>
<th>Term</th>
<th>Edge type</th>
<th>Multiple edges ok?</th>
<th>Self-loops ok?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple graph</td>
<td>Undir.</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Multigraph</td>
<td>Undir.</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Pseudograph</td>
<td>Undir.</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Directed simple graph</td>
<td>Directed</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Directed multigraph</td>
<td>Directed</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

§ 9.2 Graph Terminology

- adjacent, or neighboring
- degree,
- connects,
- endpoints,
- initial,
- terminal,
- in-degree,
- out-degree,
- complete,
- cycles,
- wheels,
- \( n \)-cubes,
- bipartite,
- subgraph,
- union.
Adjacency

Let $G$ be an undirected graph with edge set $E$. Let $e \in E$ be (or map to) the pair $\{u, v\}$.

(Note that $u$ and $v$ are vertices!)

Then we say:
- $u$, $v$ are adjacent / neighbors / connected.
- Edge $e$ is incident with vertices $u$ and $v$.
- Edge $e$ connects $u$ and $v$.
- Vertices $u$ and $v$ are endpoints of edge $e$.

Degree of a Vertex

- Let $G$ be an undirected graph, $v \in V$ a vertex.
- The degree of $v$, $\deg(v)$, is its number of incident edges. (Except that any self-loops are counted twice.)
- A vertex with degree 0 is isolated.
- A vertex of degree 1 is pendant.

Handshaking Theorem

- Let $G$ be an undirected (simple, multi-, or pseudo-) graph with vertex set $V$ and edge set $E$.
- Then \[ \sum_{v \in V} \deg(v) = 2|E| \]
- Proof: Each edge contributes twice to the degree count of all vertices
- Corollary: Any undirected graph has an even number of vertices of odd degree.

Example:

If a graph has 5 vertices, can each vertex have degree 3? 4?

Solution:
- The sum is $3 \times 5 = 15$ which is an odd number. Not possible.
- The sum is $20 = 2 \times |E|$ and $20/2 = 10$. May be possible.

Question for a class: Is it possible to have a graph of 5 vertices each having degree 1?

Answer: It is not!!! (Sum of the degrees of graph is then five, and we know that it must be EVEN)
**Directed Adjacency**

- Let $G$ be a directed (possibly multi-) graph, and let $e$ be an edge of $G$ that is (or maps to) $(u,v)$. Then we say:
  - $u$ is adjacent to $v$, $v$ is adjacent from $u$
  - $e$ comes from $u$, $e$ goes to $v$
  - $e$ connects $u$ to $v$, $e$ goes from $u$ to $v$
  - the initial vertex of $e$ is $u$
  - the terminal vertex of $e$ is $v$

**Directed Degree**

- Let $G$ be a directed graph, $v$ a vertex of $G$.
  - The in-degree of $v$, $\deg^-(v)$, is the number of edges going to $v$.
  - The out-degree of $v$, $\deg^+(v)$, is the number of edges coming from $v$.
  - The degree of $v$, $\deg(v) = \deg^-(v) + \deg^+(v)$, is the sum of $v$’s in-degree and out-degree.

**Directed Handshaking Theorem**

- Let $G$ be a directed (possibly multi-) graph with vertex set $V$ and edge set $E$. Then:
  \[ \sum_{v \in V} \deg^-(v) = \sum_{v \in V} \deg^+(v) = \frac{1}{2} \sum_{v \in V} \deg(v) = |E| \]

- Note that the degree of a node is **unchanged** by whether we consider its edges to be directed or undirected.

**Special Graph Structures**

Special cases of undirected graph structures:
- Complete graphs $K_n$
- Cycles $C_n$
- Wheels $W_n$
- $n$-Cubes $Q_n$
- Bipartite graphs
- Complete bipartite graphs $K_{m,n}$
Complete Graphs

• For any $n \in \mathbb{N}$, a complete graph on $n$ vertices, $K_n$, is a simple graph with $n$ nodes in which every node is adjacent (connected) to every other node: $\forall u, v \in V: u \neq v \leftrightarrow \{u, v\} \in E$.

Note that $K_n$ has $\sum_{i=1}^{n} \frac{n(n-1)}{2}$ edges.

Cycles

• For any $n \geq 3$, a cycle on $n$ vertices, $C_n$, is a simple graph where $V = \{v_1, v_2, \ldots, v_n\}$ and $E = \{\{v_1, v_2\}, \{v_2, v_3\}, \ldots, \{v_{n-1}, v_n\}, \{v_n, v_1\}\}$.

How many edges are there in $C_n$?

Note that in order to get a visually looking cycle, vertices should be sorted around a 'circle'.

Wheels

• For any $n \geq 3$, a wheel $W_n$, is a simple graph obtained by taking the cycle $C_n$ and adding one extra vertex $v_{\text{hub}}$ and $n$ extra edges $\{\{v_{\text{hub}}, v_1\}, \{v_{\text{hub}}, v_2\}, \ldots, \{v_{\text{hub}}, v_n\}\}$.

How many edges are there in $W_n$?

$n$-cubes (hypercubes)

• For any $n \in \mathbb{N}$, the hypercube $Q_n$ is a simple graph consisting of two copies of $Q_{n-1}$ connected together at corresponding nodes. $Q_0$ has 1 node.

Number of vertices: $2^n$. Number of edges: An exercise to try in class!

$|E_n| = 2^i |E_{n-1}| + |V_{n-1}|$
**n-cubes (hypercubes)**

- For any $n \in \mathbb{N}$, the hypercube $Q_n$ can be defined recursively as follows:
  - $Q_0=\{v_0, \emptyset\}$ (one node and no edges)
  - For any $n \in \mathbb{N}$, if $Q_n=(V,E)$, where $V=\{v_1, \ldots, v_a\}$ and $E_n=\{e_1, \ldots, e_b\}$, then $Q_{n+1}=(V \cup \{v_1', \ldots, v_a'\}, E_n \cup \{e_1', \ldots, e_b', \{v_1, v_1'\}, \{v_2, v_2'\}, \ldots, \{v_a, v_a'\}\}$ where $v_1', \ldots, v_a'$ are new vertices, and where if $e_i=\{v_j, v_k\}$ then $e_i'=\{v_j', v_k'\}$.

**Bipartite Graphs**

- Skipping this topic for this semester...

**Complete Bipartite Graphs**

- Skip...

**Subgraphs**

- A subgraph of a graph $G=(V,E)$ is a graph $H=(W,F)$ where $W \subseteq V$ and $F \subseteq E$. 
Subgraphs

Notice that the 2-cube $Q_2$ occurs inside the 3-cube $Q_3$. In other words, $Q_2$ is a subgraph of $Q_3$:

Q: How many $Q_2$ subgraphs does $Q_3$ have? 6, see next slide!

Subgraphs

A: Each face of $Q_3$ is a $Q_2$ subgraph so the answer is 6, as this is the number of sides of a 3-cube:

Graph Unions

In previous example one can actually reconstruct the 3-cube from its 6 2-cube faces:

Unions

If we assign the 2-cube sides (i.e., squares) the names $S_1$, $S_2$, $S_3$, $S_4$, $S_5$, $S_6$ then $Q_3$ is the union of its faces:

$Q_3 = S_1 \cup S_2 \cup S_3 \cup S_4 \cup S_5 \cup S_6$
Graph Unions

- The union $G_1 \cup G_2$ of two simple graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ (where $V_1, V_2$ may or may not be disjoint) is the simple graph $(V_1 \cup V_2, E_1 \cup E_2)$, i.e.,
  
  $G_1 \cup G_2 = (V_1 \cup V_2, E_1 \cup E_2)$

A similar definition can be created for unions of digraphs, multigraphs, pseudographs, etc.

§ 9.3 Graph Representations & Isomorphism

- Graph representations:
  - Adjacency lists.
  - Adjacency matrices.
  - Incidence matrices.

- Graph isomorphism:
  - Two graphs are isomorphic iff they are identical except for their node names.

Adjacency Lists

- A table with 1 row per vertex, listing its adjacent vertices.

Directed Adjacency Lists

- 1 row per node, listing the terminal nodes of each edge incident from that node.
We already saw a way of representing relations on a set with a Boolean matrix: 
\[ R \rightarrow \text{digraph}(R) \]

Note, the matrix is for a DIRECTED GRAPH on the left.

Can easily generalize to directed multigraphs by putting in the number of edges between vertices, instead of only allowing 0 and 1:

For a directed multigraph \( G = (V,E) \) define the matrix \( A_G \) by:
- Rows, Columns – one for each vertex in \( V \)
- Value at \( i^{th} \) row and \( j^{th} \) column is
  - The number of edges with source the \( i^{th} \) vertex and target the \( j^{th} \) vertex
Adjacency Matrix - Directed Multigraphs

A:

\[
\begin{pmatrix}
0 & 3 & 0 & 1 \\
0 & 1 & 2 & 0 \\
0 & 1 & 2 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}
\]

Adjacency Matrix-General

Undirected graphs can be viewed as directed graphs by turning each undirected edge into two oppositely oriented directed edges, except when the edge is a self-loop in which case only 1 directed edge is introduced. EG:

Q: What’s the adjacency matrix?

A: Notice that answer is symmetric.

\[
\begin{pmatrix}
0 & 2 & 1 & 0 \\
2 & 2 & 1 & 0 \\
1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{pmatrix}
\]
For an undirected graph $G = (V,E)$ define the matrix $A_G$ by:
- Rows, Columns – one for each element of $V$
- Value at $i^{th}$ row and $j^{th}$ column is the number of edges incidents with vertices $i$ and $j$.

**Isomorphism**

The two graphs below are really the same graph.

One is drawn so that no edges intersect (planar).

We say these graphs are *isomorphic*.

**Graph Isomorphism**

- Formal definition:
  - Simple graphs $G_1=(V_1, E_1)$ and $G_2=(V_2, E_2)$ are *isomorphic* iff $\exists$ a bijection $f: V_1 \rightarrow V_2$ such that $\forall a,b \in V_1$, $a$ and $b$ are adjacent in $G_1$ iff $f(a)$ and $f(b)$ are adjacent in $G_2$.
  - $f$ is the “renaming” function that makes the two graphs identical.
  - Definition can easily be extended to other types of graphs.

**Graph Invariants under Isomorphism**

* Necessary but not *sufficient* conditions for $G_1=(V_1, E_1)$ to be isomorphic to $G_2=(V_2, E_2)$:
  - $|V_1|=|V_2|$, $|E_1|=|E_2|$.
  - The number of vertices with degree $n$ is the same in both graphs.
  - For every proper subgraph $g$ of one graph, there is a proper subgraph of the other graph that is isomorphic to $g$. 
Are the following 2 graphs isomorphic?

Note! Proving isomorphism is a very hard problem. Doing it by hand is a bummer!! **Why?**

**Invariants** - things that $G_1$ and $G_2$ must have in common to be isomorphic:
- the same number of vertices
- the same number of edges
- degrees of corresponding vertices are the same.
- if one is bipartite, the other must be
- if one is complete, the other must be
- if one is a wheel, the other must be etc.

We first try the relabeling using i) in each case to get the function $f:
1 \rightarrow 2, 2 \rightarrow 1, 3 \rightarrow 5, 4 \rightarrow 3, 5 \rightarrow 4.$

- permute the rows and columns of the adjacency matrix of $G_1$ using the above map to see if we get the adjacency matrix of $G_2.$

or

- change the labels of the graph $G_2$ to produce the graph $G_2^{i*}$ according to the above permutation and recalculate the adjacency matrix. **Recall:**

$$
\begin{array}{c|c|c|c|c|c}
1 & 2 & 3 & 4 & 5 \\
1 & 0 & 0 & 0 & 0 & 0 \\
2 & 0 & 0 & 1 & 0 & 0 \\
3 & 0 & 1 & 0 & 0 & 0 \\
4 & 0 & 0 & 0 & 0 & 1 \\
5 & 0 & 0 & 1 & 0 & 0 \\
\end{array}
$$

$$
G_1 = 
\begin{bmatrix}
0 & 1 & 0 & 1 & 1 \\
1 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 1 & 0 \\
1 & 1 & 1 & 0 & 1 \\
1 & 1 & 0 & 1 & 0 \\
\end{bmatrix}
$$

$$
G_2 = 
\begin{bmatrix}
0 & 0 & 1 & 1 & 1 \\
0 & 0 & 1 & 0 & 1 \\
0 & 1 & 0 & 1 & 0 \\
1 & 1 & 0 & 1 & 1 \\
1 & 1 & 1 & 1 & 0 \\
\end{bmatrix}
$$

Solution

Check...

- They have the same number of vertices = 5
- They have the same number of edges = 8
- They have the same number of vertices with the same degrees: 2, 3, 3, 4, 4.
- Now we try to construct the isomorphism $f$ using the degrees of vertices to help us.

$$
\begin{array}{c|c|c|c|c|c}
1 & 2 & 3 & 4 & 5 \\
\text{deg}(u_3) = \text{deg}(v_2) = 2 \ so \\
\text{deg}(u_3) = \text{deg}(v_1) = \text{deg}(v_4) = 3 \ so \\
\text{we must have either} \\
1) f(u_2) = v_1 \ and \ f(u_3) = v_4 \\
2) f(u_2) = v_1 \ and \ f(u_3) = v_4 \\
\end{array}
$$

Perhaps either choice will work.

$$
P = 
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 \\
\end{bmatrix}
$$

$$
G_1 r = P \times G_1 \\
G_2 \_star = G_1 r \times P
$$

MATLAB CODE relabeling $i$

Instructor only: Run the code isomorphism_graphs.m

$$
G_1 = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{bmatrix}
$$

$$
P = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}
$$

$$
G_1 r = P \times G_1 \\
G_2 \_star = G_1 r \times P
$$
MATLAB CODE relabeling $ii$

$G_1 = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{bmatrix}$

$P = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}$

Multiply from LEFT to permute the ROWS:

$G_{1r} = P^T G_1$

$G_{1r} = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \end{bmatrix}$

Multiply from RIGHT to permute the COLUMNS:

$G_{2*} = G_{1r} P$

$G_{2*} = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \end{bmatrix}$

Solution, cont.

If we did the same steps by multiplying $G_2$ by $P$ from left and right we would have got

The new labeling of $G_2$, $G_2^*$, becomes

The new adjacency matrix becomes:

$G_2^* = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{bmatrix}$

which is the same adjacency matrix as for $G_1$. Hence we have found an isomorphism!

Isomorphism Example

- If isomorphic, label the 2nd graph to show the isomorphism, else identify difference.

Are These Isomorphic?

- If isomorphic, label the 2nd graph to show the isomorphism, else identify difference.

* Same # of vertices
* Same # of edges
* Different # of verts of degree 2! (1 vs 3)
Which of the graphs below are isomorphic?

A & R,
F & T,
K & X,
M, S, V & Z

§ 9.4 Connectivity

- In an undirected graph, a path of length \( n \) from \( u \) to \( v \) is a sequence of adjacent edges going from vertex \( u \) to vertex \( v \).
- A path is a circuit if \( u=v \), i.e., if it ends at \( u \).
- A path traverses the vertices along it.
- A path is simple if it contains no edge more than once.

Note: There is nothing to prevent traversing an edge back and forth to produce arbitrarily long paths. This is usually not interesting which is why we define a simple path.

Example:

There are many paths from \( u_1 \) to \( u_3 \) in \( G_1 \):

1) \( u_1, u_4, u_2, u_3 \); length = 3, the path is simple
2) \( u_1, u_5, u_4, u_1, u_2, u_3 \); length = 5, the path is simple and it contains a circuit \( u_1, u_5, u_4, u_1 \).
3) \( u_1, u_2, u_5, u_4, u_3 \); length = 4, the path is simple

How many simple paths are there?

Paths in Directed Graphs

- Same as in undirected graphs, but the path must go in the direction of the arrows.
**Connectedness**

- An *undirected* graph is *connected* iff there is a path between *every pair of distinct vertices* in the graph.
- Theorem: There is a *simple* path between any pair of vertices in a *connected* undirected graph.
- *Connected component:* connected subgraph
- A *cut vertex* or *cut edge* separates 1 connected component into 2 if removed.

**Directed Connectedness**

- A *directed* graph is *strongly connected* iff there is a directed path from $a$ to $b$ for any two vertices $a$ and $b$.
- It is *weakly connected* iff the *underlying undirected graph* (i.e., with edge directions removed) is connected.
- Note *strongly* implies *weakly* but not vice-versa.
Paths & Isomorphism

• Note that connectedness, and the existence of a circuit or simple circuit of length $k$ are graph invariants with respect to isomorphism.

Counting Paths w Adjacency Matrices

• Let $A$ be the adjacency matrix of graph $G$.
• The number of paths of length $k$ from $v_i$ to $v_j$ is equal to $(A^k)_{ij}$. (The notation $(M)_{ij}$ denotes $m_{ij}$ where $[m_{ij}] = M$.)

Example:

Caution!!!

• We are analyzing undirected graphs here
• So, there will be differences in respect to math we used in Transitive Closures
• There, we used Boolean Product
• Here, we'll use a classic/standard matrix product
• Hence, the matrices we'll get will tell us some new stories. They will give us some novel and different insights.

**How many 2-pathses are there between vertices 1-1-2 ?**

**How many 3-pathses are there between vertices 1-1-2-2-2 ?**

Reminder for lecturer only!
RUN GRAPHS.M, now!!!
Here, the **Graphs** stories end, and the **Chapter 9 on Trees** start.

As it may be suspected, **Trees** are just special subgroups of **Graphs** but, due to their importance and overall usefulness Trees are treated separately and in details!!!