Last Time

- Wrapped up Optimization
Graphs...at long last
Define graphs

A graph is, minimally,

- a set of **vertices** (or nodes).
- a set of **edges** where each edge has a starting vertex and ending vertex.

Edges can be **directed**.

Vertices and Edges can each be **labeled**

Graphs are very general data structures.
You have seen Graphs before.
Trees are Graphs without cycles

Trees are Graphs
Graphs in Pop-Culture
Given the graph

Find:
1) the adjacency matrix $A$
2) the matrix giving the number of 3 step walks
3) the generating function for walks from point $i \rightarrow j$
4) the generating function for walks from points 1 \rightarrow 3.
Graphs in Pop-Culture
Example graphs

Undirected Graph

A -- B -- C
D -- E -- F
G -- A
Undirected Graph – SAME GRAPH as previous example
Example graphs

Directed Graph

- A
- B
- C
- D
- E
- F
- G
Multiple connected components

- D
- E
- C
- G
- A
- B
- F
Cycles

If there exists a path in $G$ that contains the same vertex twice, $G$ contains a cycle.

A **Path** is a series of vertices such that each adjacent vertex is connected by an edge.
Example Path: G, A, F
Example Cycle: A, F, B, E, A
NOT a Path: G, A, F
Example Path: C, D, E, A
Not a Cycle: C, D, E, B, C
Example Cycle: A, D, E, A
Search Trees are **Directed Acyclic Graphs** (DAGs).

- **Directed** - Parents have edges to children
- **Acyclic** - Trees never contain cycles
Forests

Graphs that are made up of multiple unconnected Trees are called Forests.
Edges can be labeled with weights.

These weights often correspond to some *cost* or *distance* to move from one vertex to another.
Graph coloring.

Also Vertices can be labeled.

In A* search, vertices are labeled with heuristic distances.

Example: The map coloring problem.
Map coloring

Given a map. What is the minimum number of colors which are required such that no two adjacent regions have the same color.
Regions are vertices.

Adjacent Regions are connected by undirected, unweighted edges.
Color the vertices with as few colors as possible, based on adjacency information.
Map as Graph

Vertex colors are then translated to map colors.

**Four Color Theorem**: Any map (containing planar contiguous regions) can be colored with four or fewer colors.
Graph problems

We’ll be addressing these in the next few classes.

- Reachability
- Searching in a Graph
- (Strongly) Connected Components
- Spanning Trees
- Shortest Paths
- Network Flows
- Matching Problems (vertex cover, edge cover)
Representations of Graphs

- Object Oriented
- Adjacency Matrix
- Edge Lists
Object Oriented Graphs

class Graph {
    vector<Node> nodes;
    vector<Edge> edges;
}

class Edge {
    Node source;
    Node destination;
}

class Node {
    vector<Edge> edges;
}
### Adjacency Matrix

<table>
<thead>
<tr>
<th>Source</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>D</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

The graph shown in the diagram corresponds to the adjacency matrix:

- **A** connects to **C** and **D**.
- **B** connects to **A** and **D**.
- **C** connects to **A** and **D**.
- **D** connects to **C** and **B**.
## Adjacency List

<table>
<thead>
<tr>
<th>Src</th>
<th>Dest</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>C</td>
</tr>
<tr>
<td>B</td>
<td>A → D</td>
</tr>
<tr>
<td>C</td>
<td>A → C</td>
</tr>
<tr>
<td>D</td>
<td>B → D</td>
</tr>
</tbody>
</table>

![Graph Diagram]
Complete Graph

Every vertex is connected to every other vertex
Bipartite Graph

The vertexes can be divided into two sets, $V_A$ and $V_B$, such that every edge contains exactly one member from $V_A$ and one from $V_B$. 
Run time of Graph algorithms

- We’ll be talking about runtimes in terms of $V$ the number of vertices and $E$ the number of edges.
- How many edges can there be in a graph?
We’ll be talking about runtimes in terms of $V$ the number of vertices and $E$ the number of edges.

How many edges can there be in a graph?

- $V^2$ (assuming unique edges)
- $V(V - 1)$ with no self loops.
Next time
- Traversing Graphs
- Reachability
- Strongly Connected Components