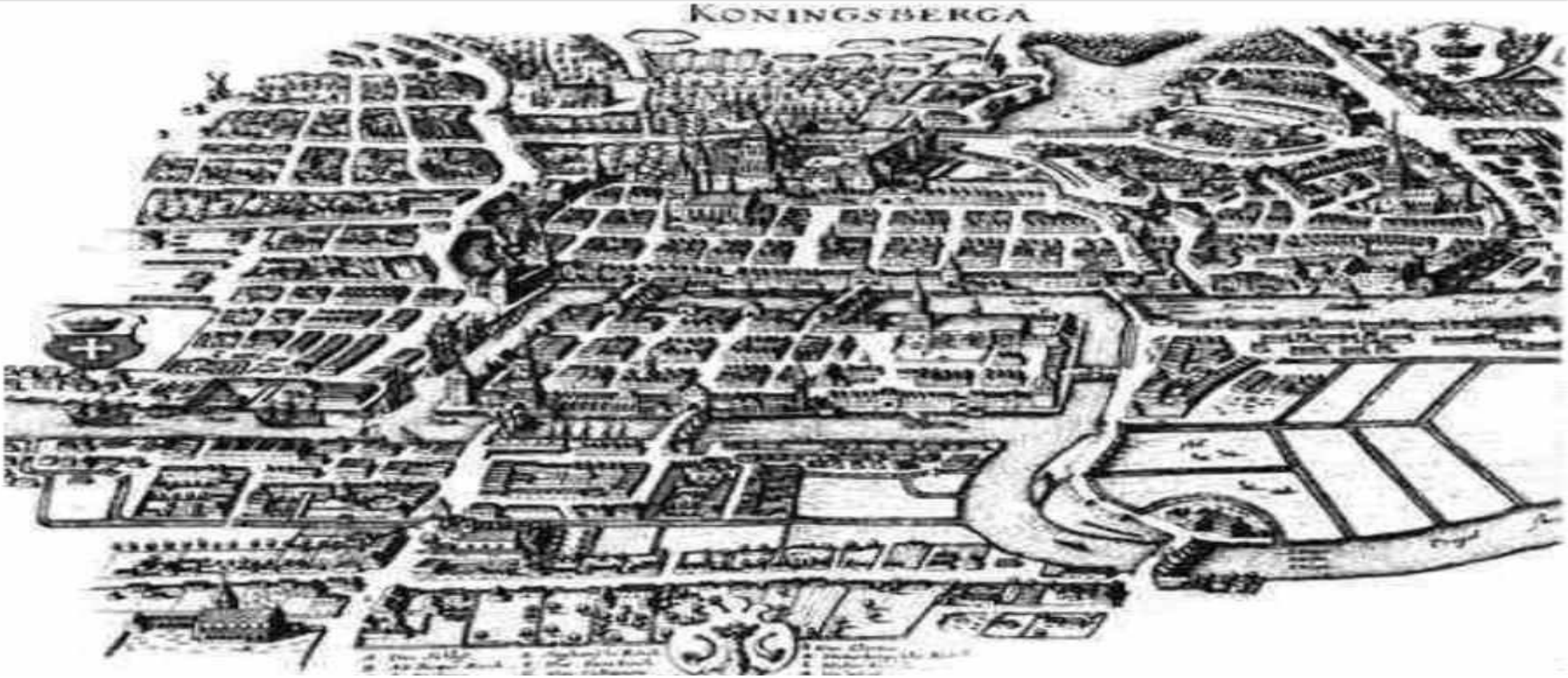
GRAPH
INTRODUCTION

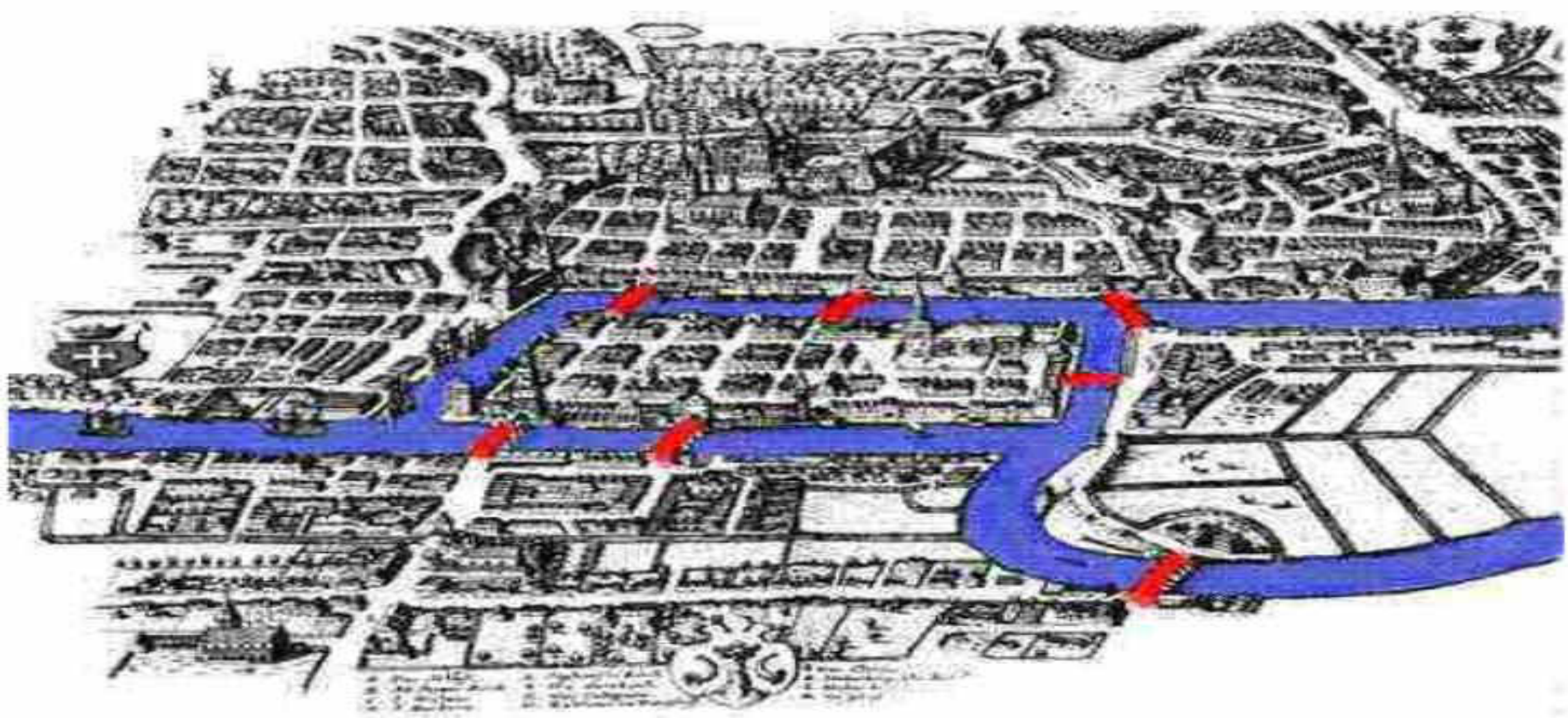
History of Graph | THEORY

- Origin
  - Problem: (of) **Königsberg** bridge, in 1735.
  - In 1735, the city of **Königsberg** (present-day Kaliningrad and part of Russia) was divided into four districts by the Pregel River.
  - The four districts were connected by seven bridges.

The Seven Bridges of Konigsberg

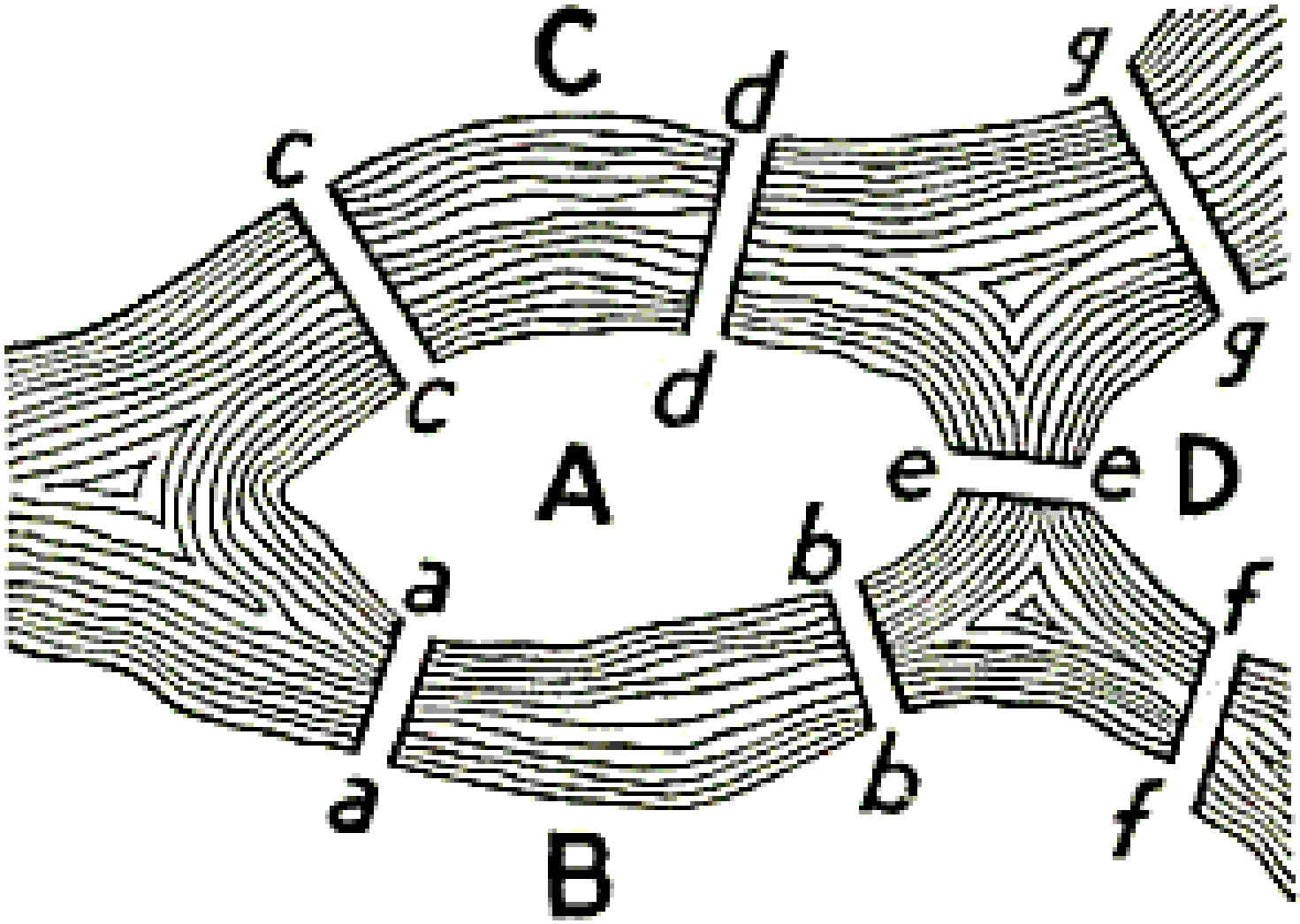


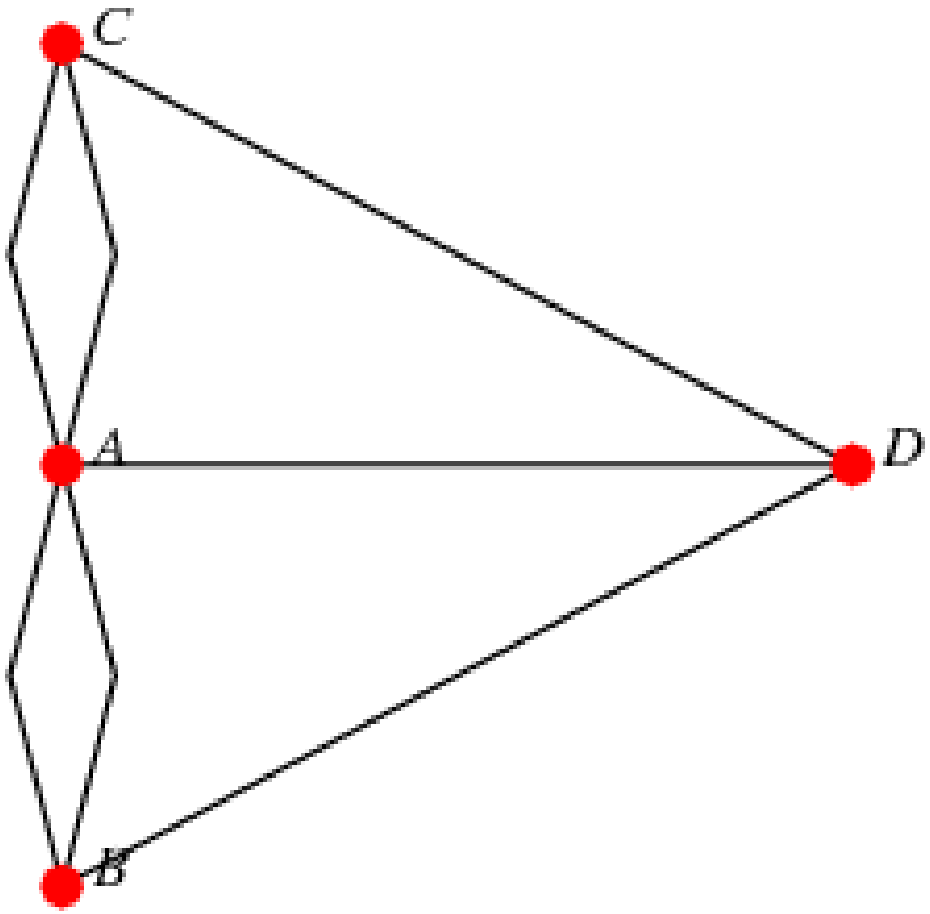
Is it possible to design a walking tour of Königsberg in which you cross each of the seven bridges exactly once?



Königsberg Bridge Problem

This problem was answered in the negative by Euler

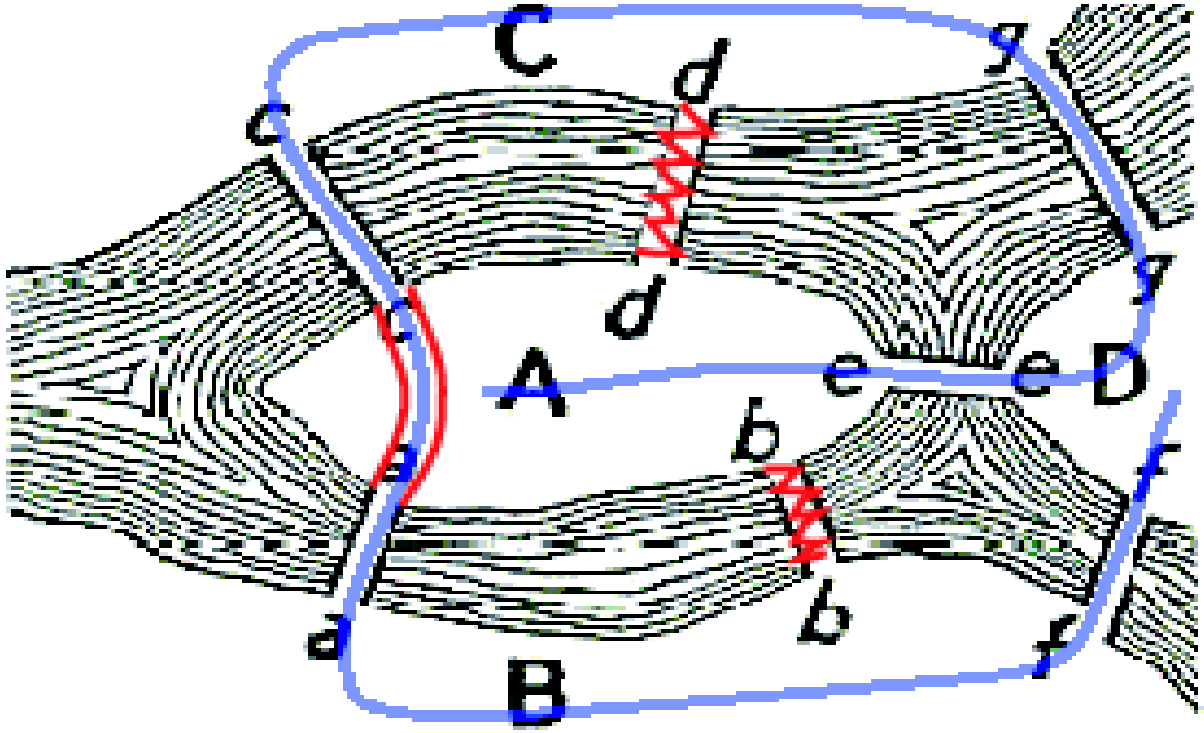


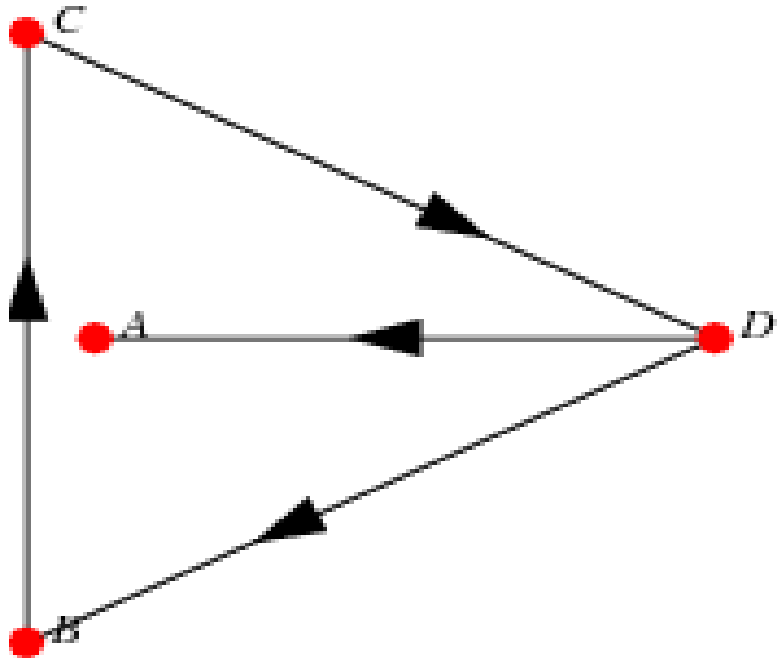


J. Kåhre observes

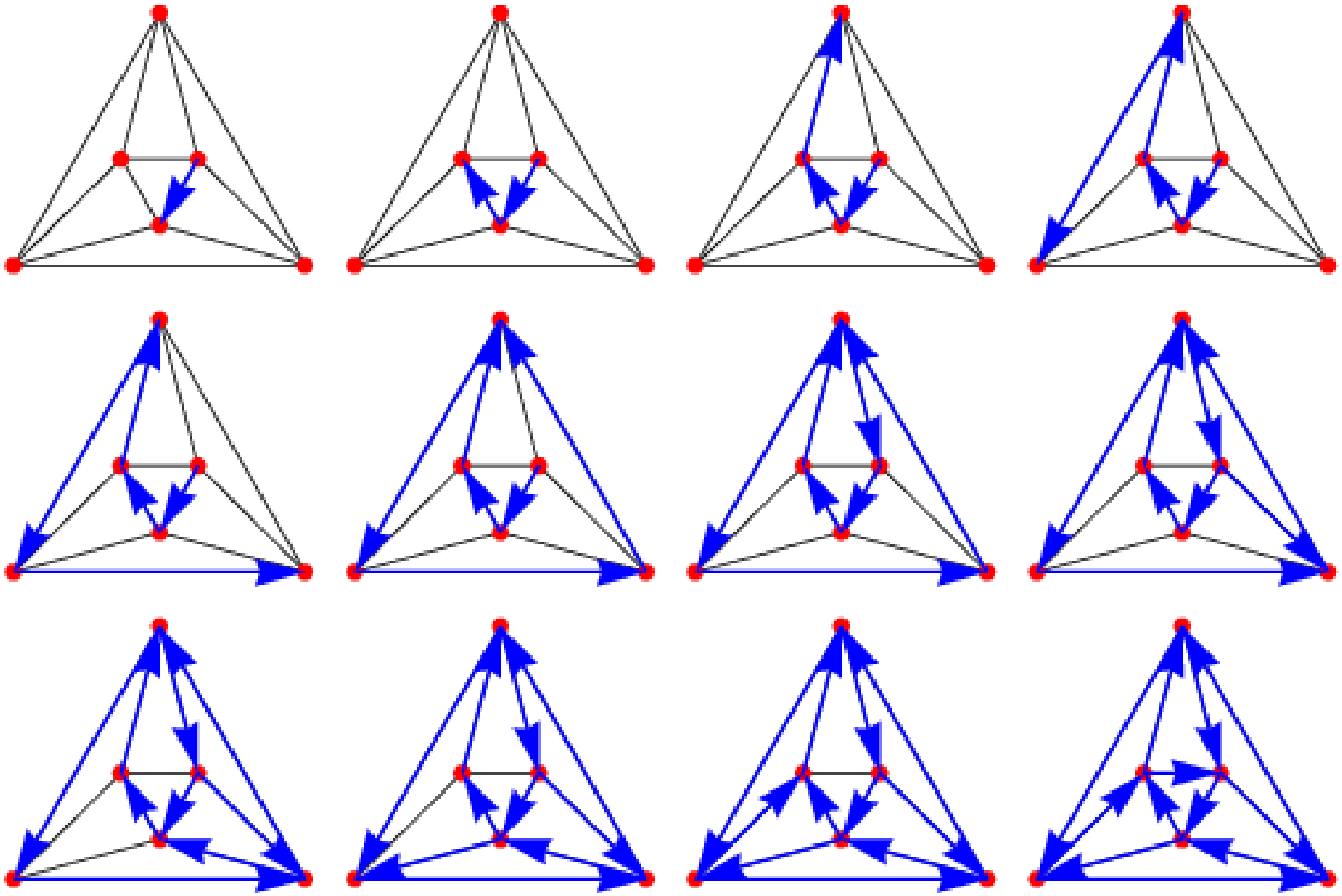
using the modern Königsberg bridges

that bridges ***b b*** and ***d d*** no longer exist and that ***a a*** and ***c c*** are now a single bridge passing above with a stairway in the middle leading down to . Even so, there is *still* no Eulerian cycle on the nodes ***A***,***B*** ,***C*** , and ***D***.





Eulerian Cycle

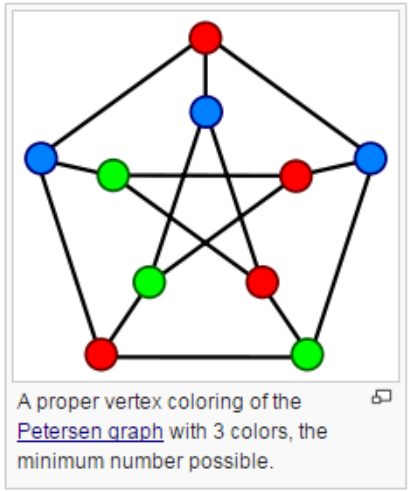


An Eulerian cycle, also called an Eulerian circuit, Euler circuit, Eulerian tour, or Euler tour, is a trace which starts and ends at the same graph vertex.

In other words, it is a graph cycle which uses each graph edge exactly once.

The Mathematics of Networks

- The mathematical models we need to solve the Konigsberg problem is a **graph**.
  - designing travel routes
  - connecting networks efficiently
  - scheduling tasks
  - coloring regions of maps



A proper vertex coloring of the Petersen graph with 3 colors, the minimum number possible.

Graphs

Graph theory can be defined as the study of *graphs*;

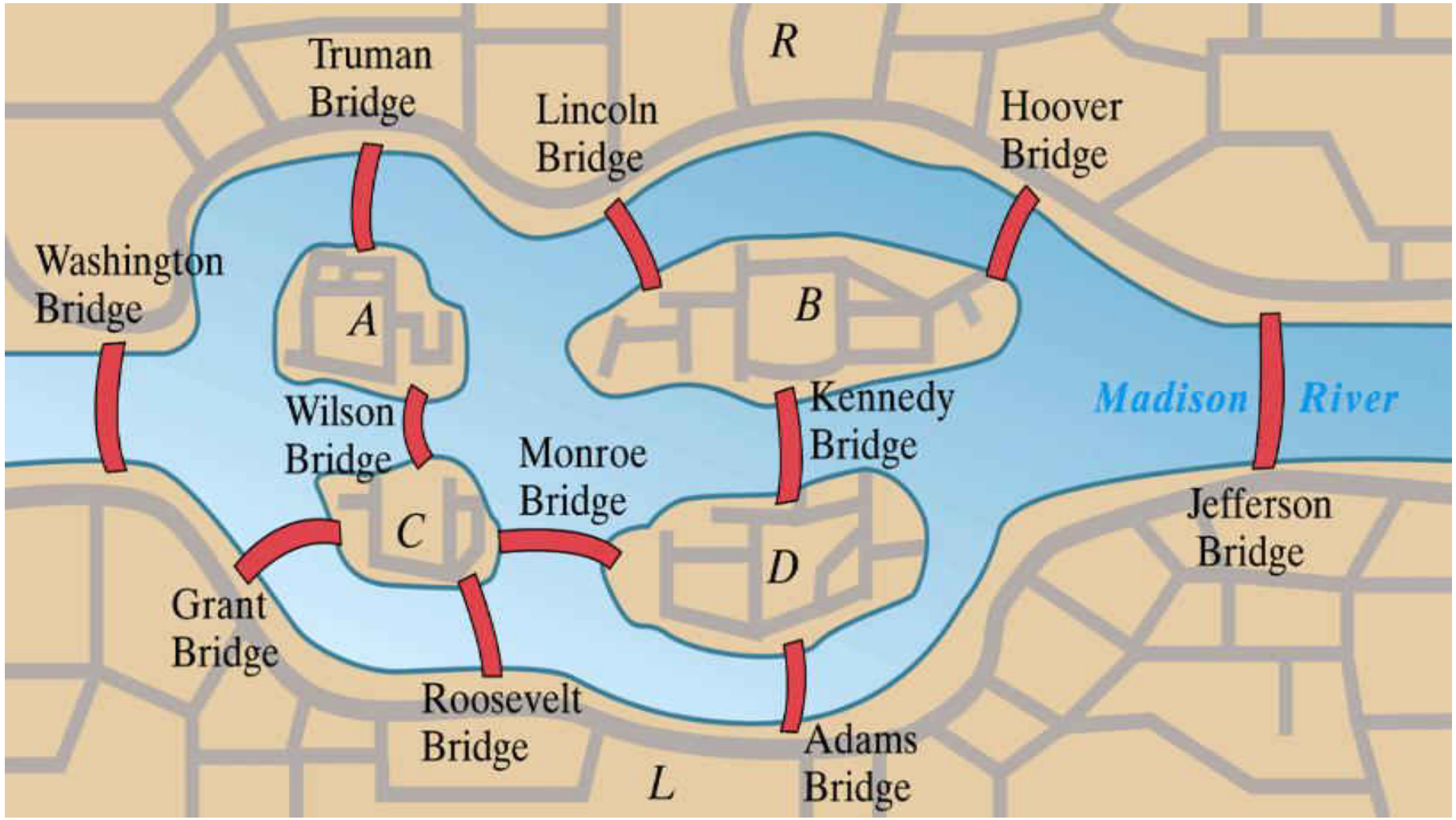
Graphs are mathematical structures used to model pair-wise relations between objects from a certain collection. **Graph** can be defined a set V of vertices and set of edges. Where, V is collection of |V| = n abstract data types. Vertices can be any abstract data types and can be presented with the points in the plane. These abstract data types are also called nodes. A line (line segment) connecting these nodes is called an edge. Again, more abstractly saying, edge can be an abstract data type that shows relation between the nodes (which again can be an abstract data types).

Leonhard Paul Euler (1707- 1783) was a pioneering Swiss mathematician, who spent most of his life in Russia and Germany. Euler (pronounced as OILER) solved the first problem using graph theory and thereby led the foundation of very vast and important field of graph theory. He created first graph to simulate a real time place and situation to solve a problem which was then considered one of the toughest problems.

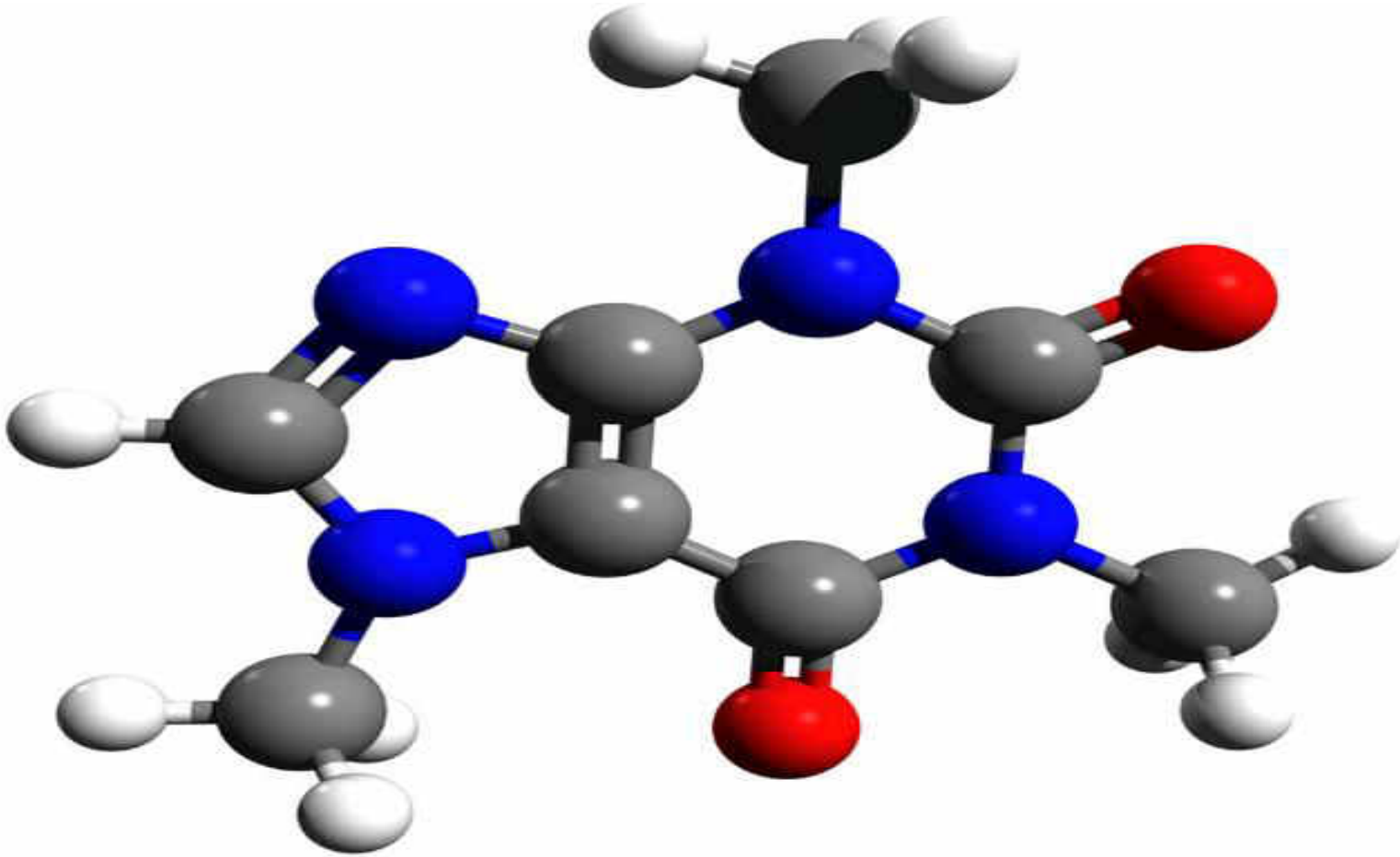
Graphs

- Things that can be modeled with graphs include
  - Maps
  - Molecules
  - flow charts
  - family trees
  - Internet (web pages connected by links)
  - Facebook/Google+ (people connected by friendship)

Maps

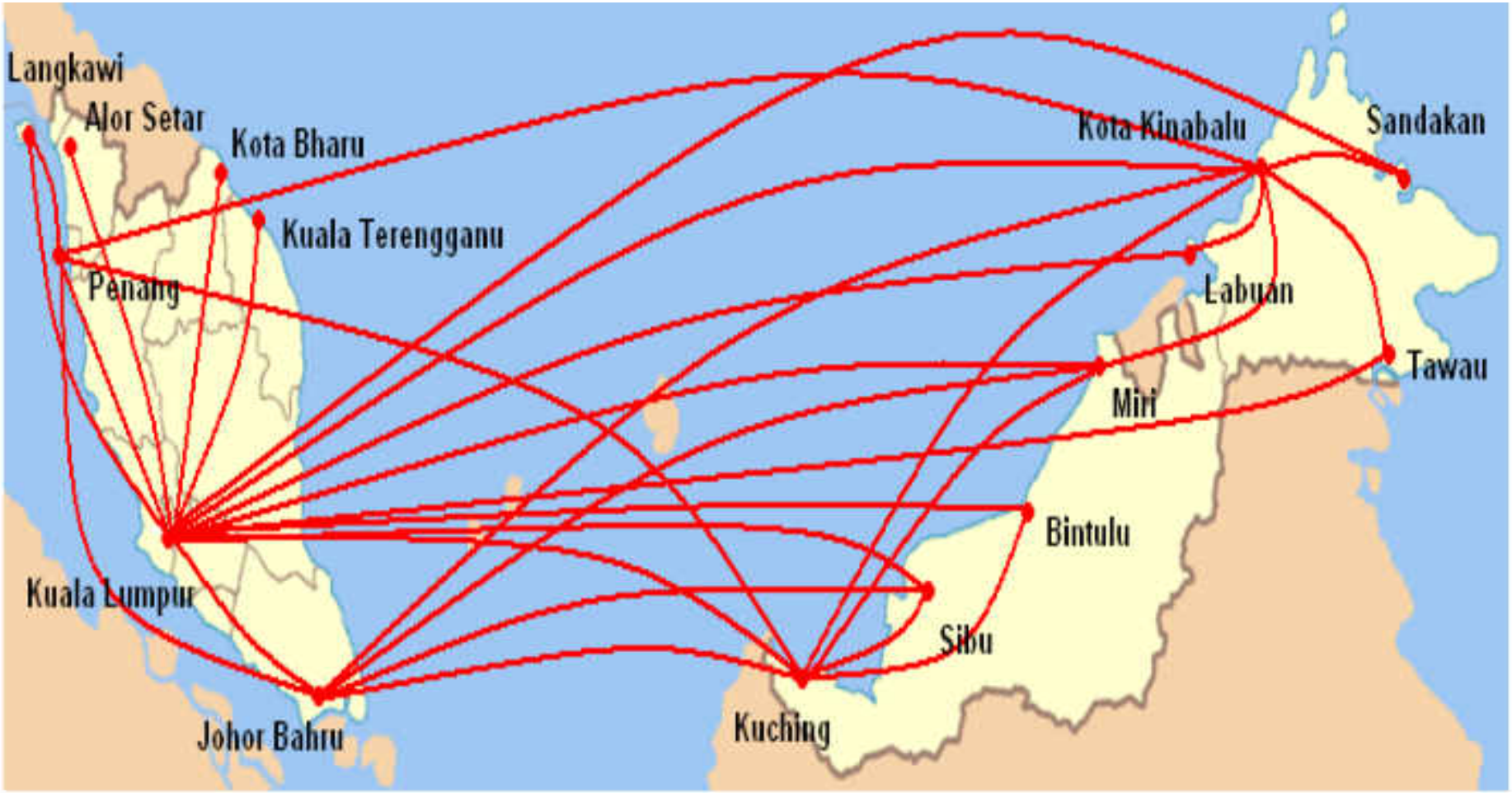

R
Truman Bridge
Lincoln Bridge
Hoover Bridge
Washington Bridge
A
B
Wilson Bridge
Kennedy Bridge
Madison River
Monroe Bridge
C
D
Jefferson Bridge
Grant Bridge
Roosevelt Bridge
Adams Bridge
L

Molecules



Source: http://commons.wikimedia.org/wiki/File:Caffeine_3d_structure.png

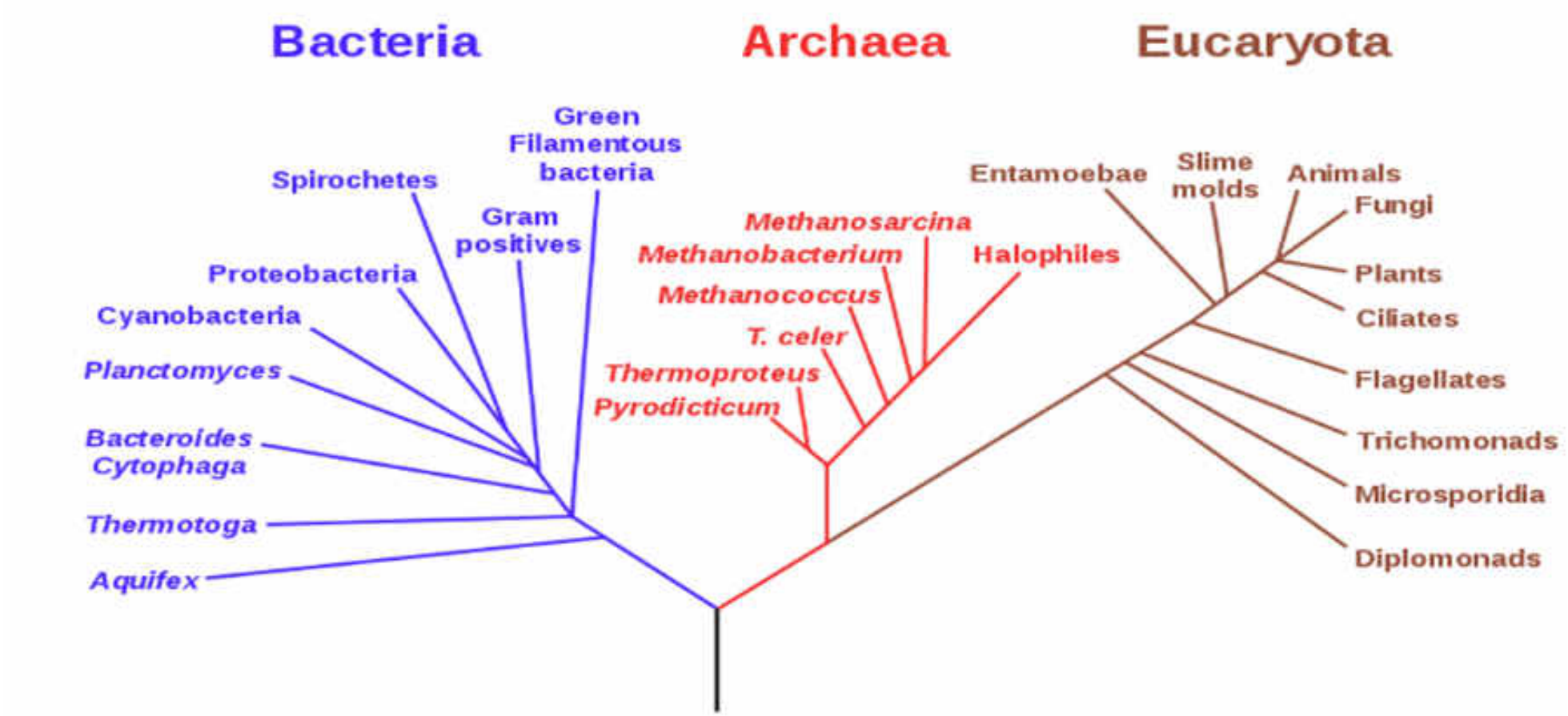
Plane Route Map



Source: upload.wikimedia.org/wikipedia/commons/2/20/AA_route_map.PNG

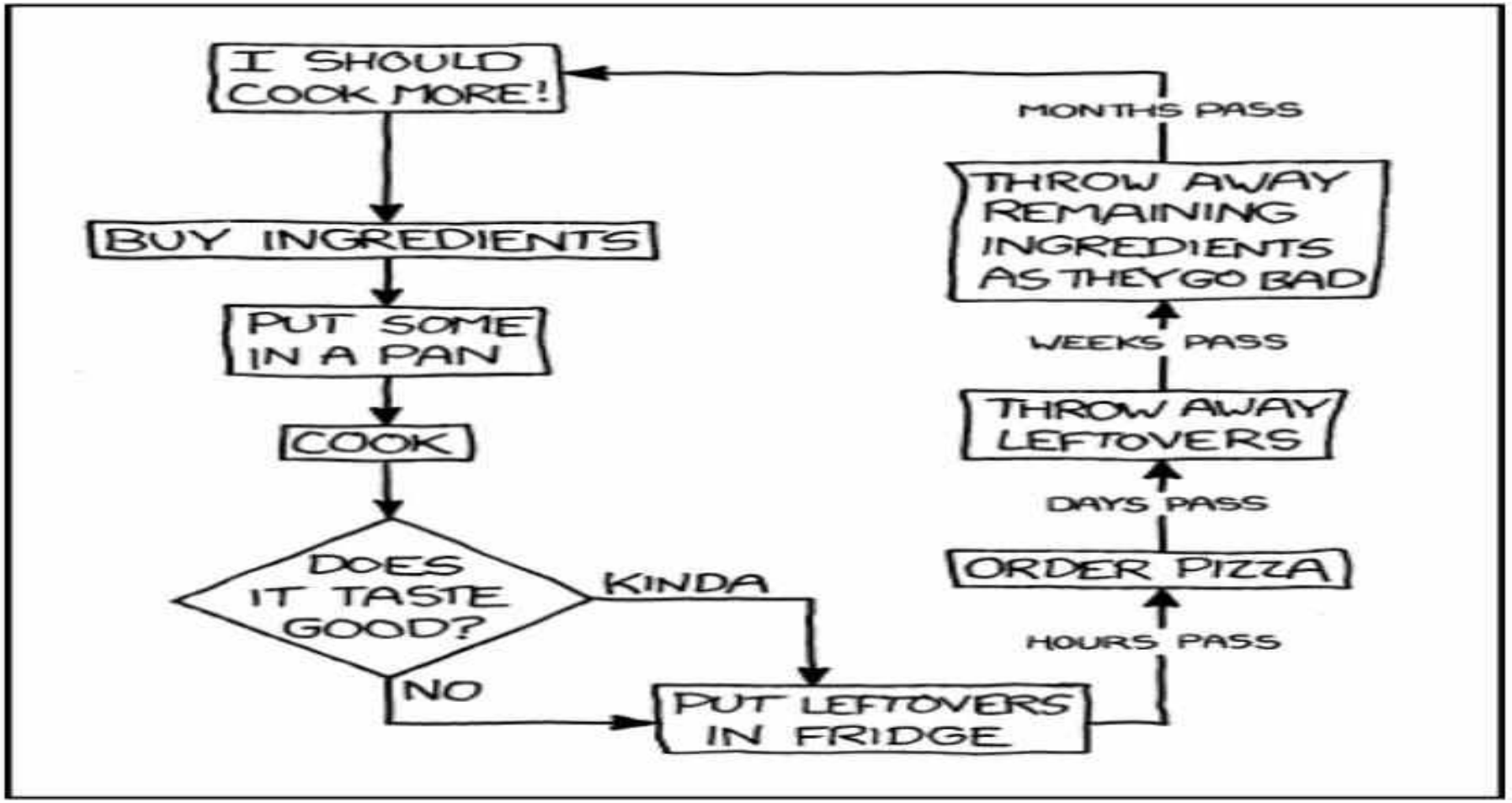
Transportation Route Map

Phylogenetic Tree



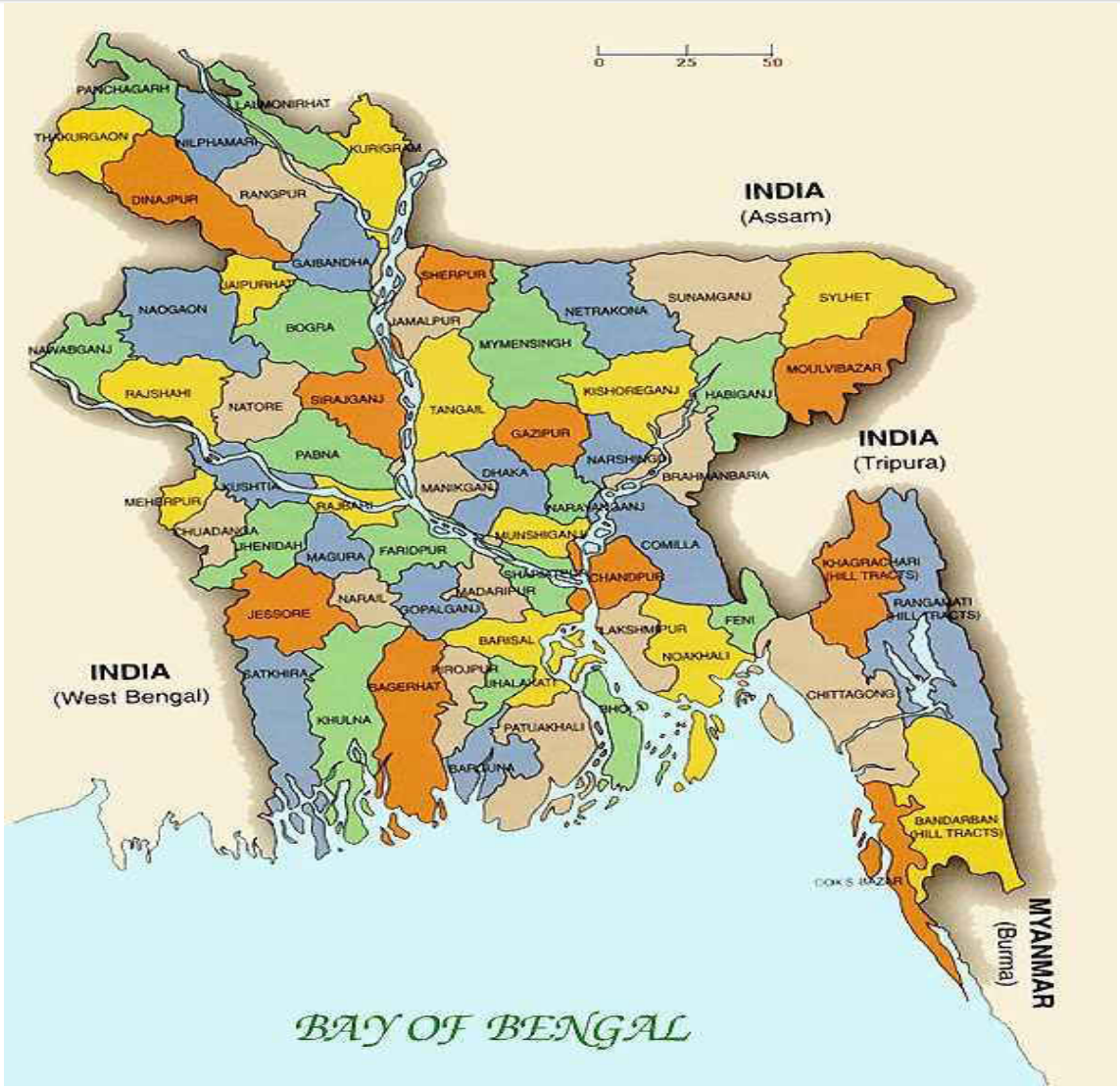
Source: http://commons.wikimedia.org/wiki/File:Phylogenetic_tree.svg

Flow Chart

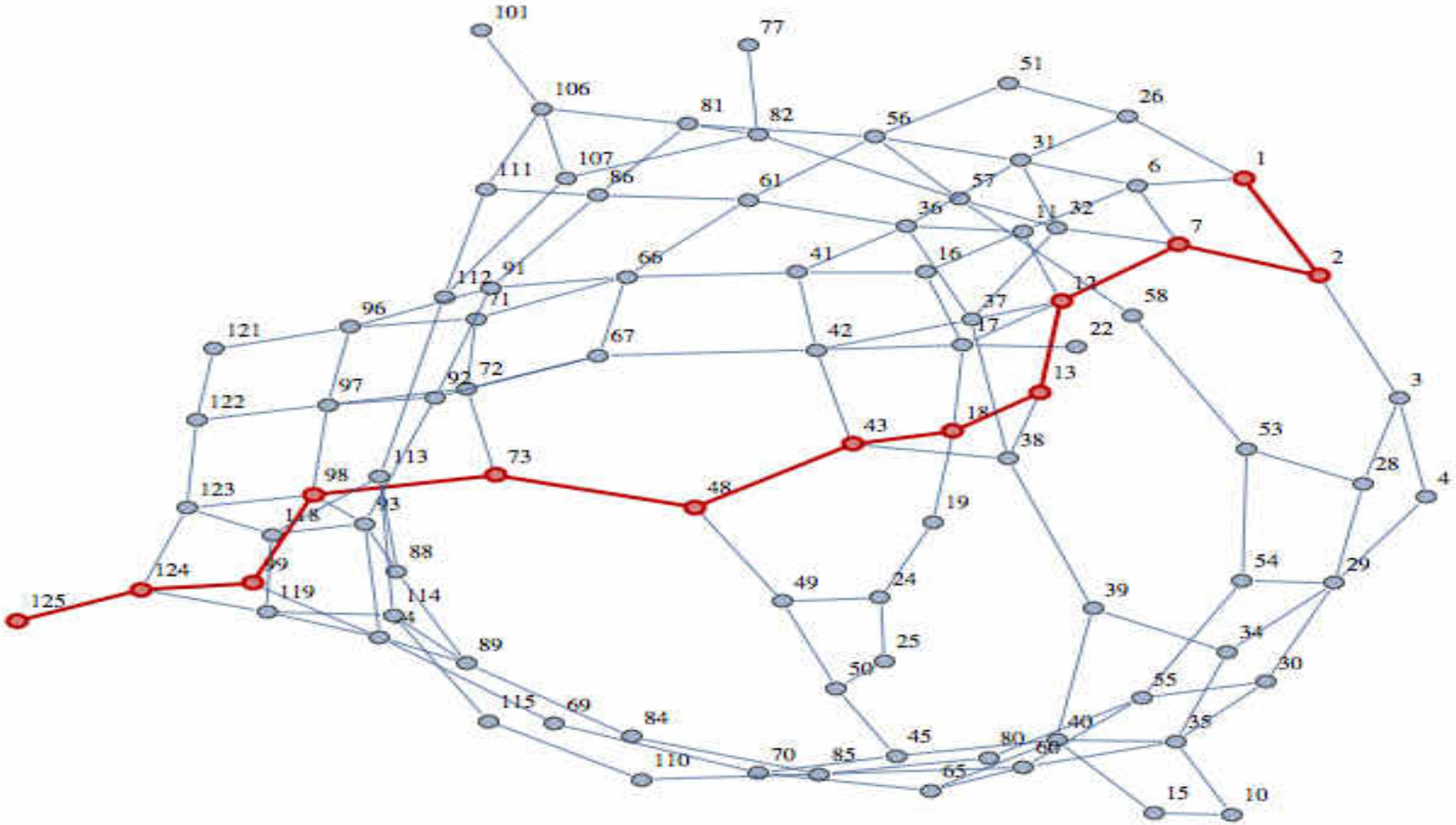


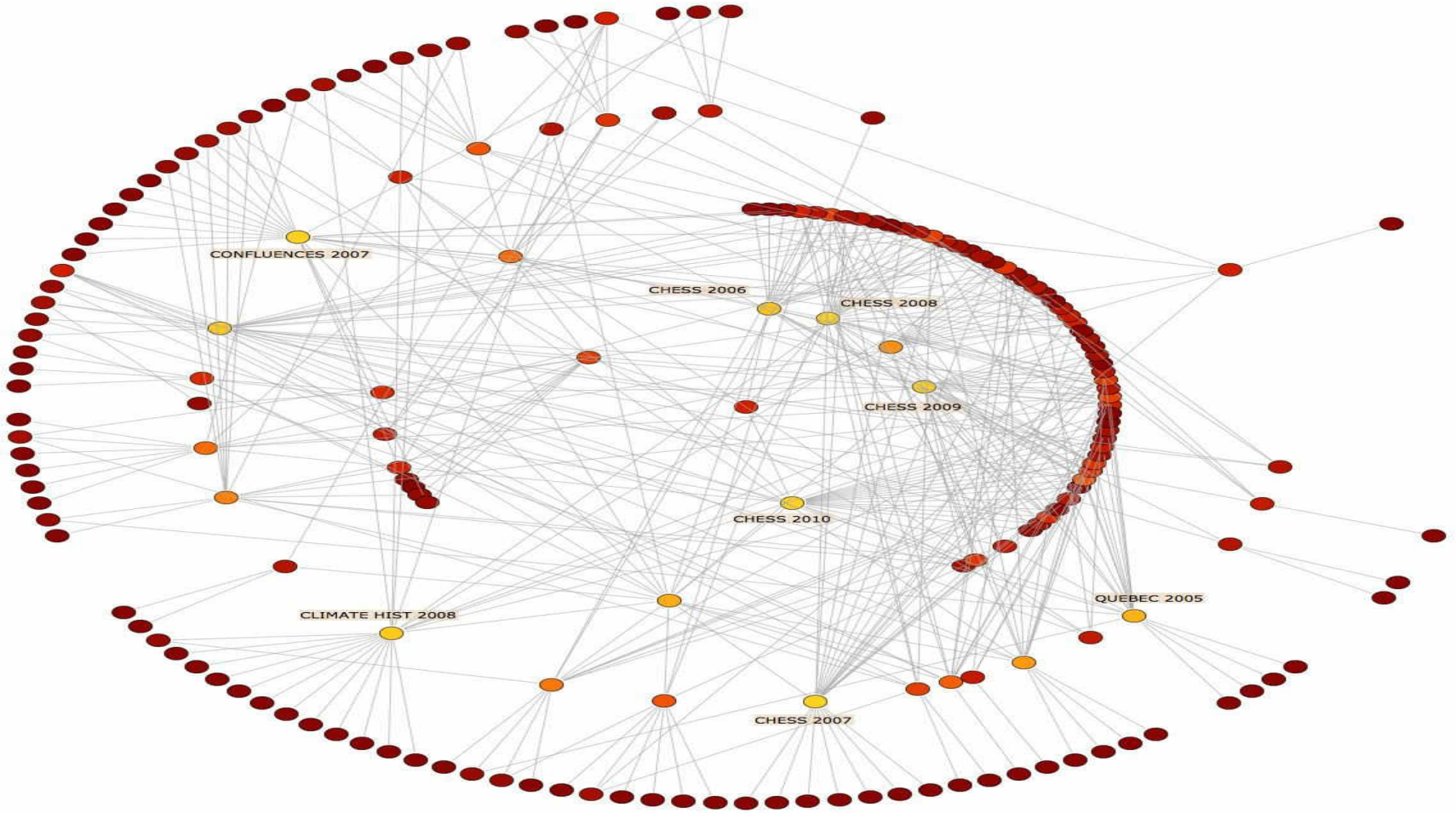
Source: http://xkcd.com/854/

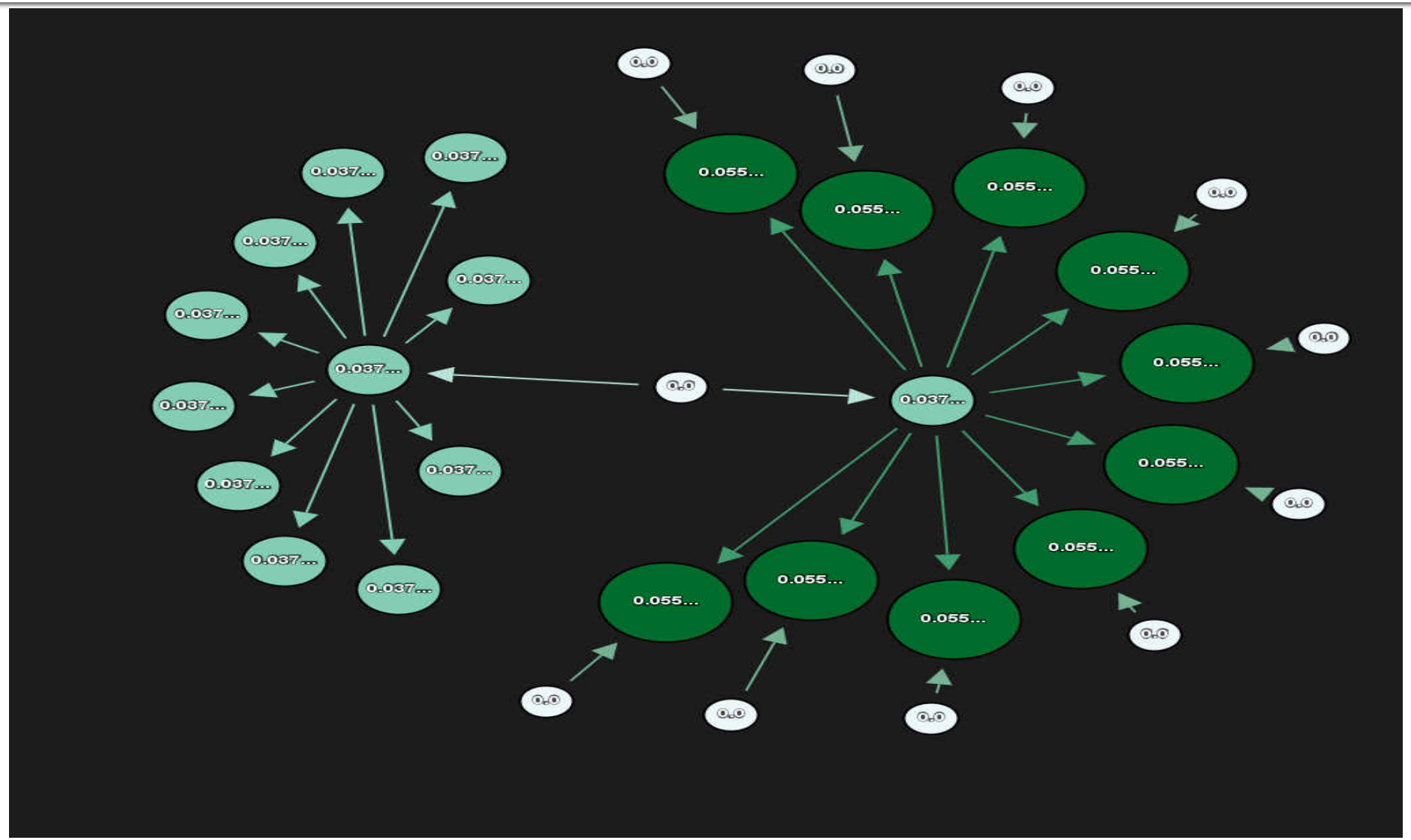
Graph Coloring



Shortest Path

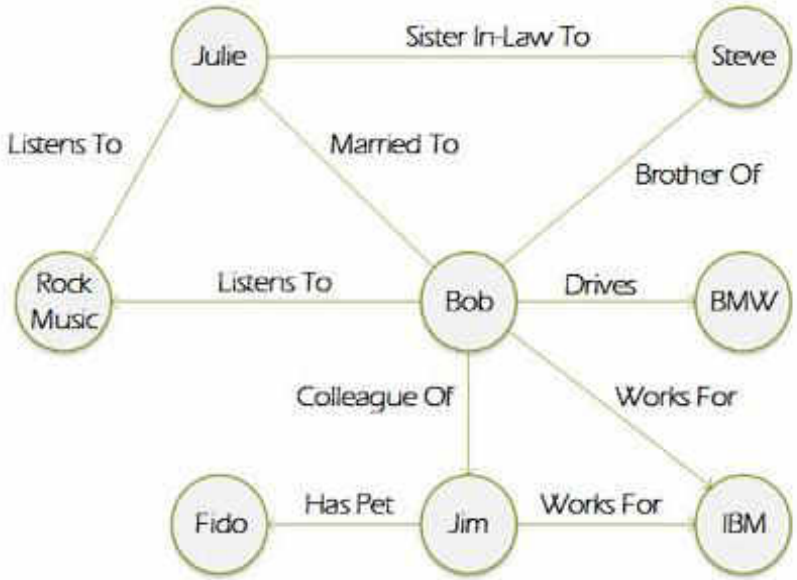



CONFLUENCES 2007
CHESS 2006
CHESS 2008
CHESS 2009
CHESS 2010
CLIMATE HIST 2008
QUEBEC 2005
CHESS 2007


0.0
0.0
0.0
0.055...
0.055...
0.055...
0.0
0.055...
0.055...
0.0
0.055...
0.0
0.055...
0.0
0.055...
0.055...
0.055...
0.0
0.0
0.0
0.0
0.037...
0.037...
0.037...
0.037...
0.037...
0.037...
0.037...
0.037...
0.037...
0.037...
0.037...
0.037...
0.037...
0.0

Graph applications | in computer science

- Data base designing
  - A graph database, also called a graph-oriented database, is a type of NoSQL database that uses graph theory to store, map and query relationships.
  - Graph database uses graph representation with nodes, edges, and properties to represent and store data.
  - Graph databases are well-suited for analyzing interconnections, which is why there has been a lot of interest in using graph databases to mine data from social media.

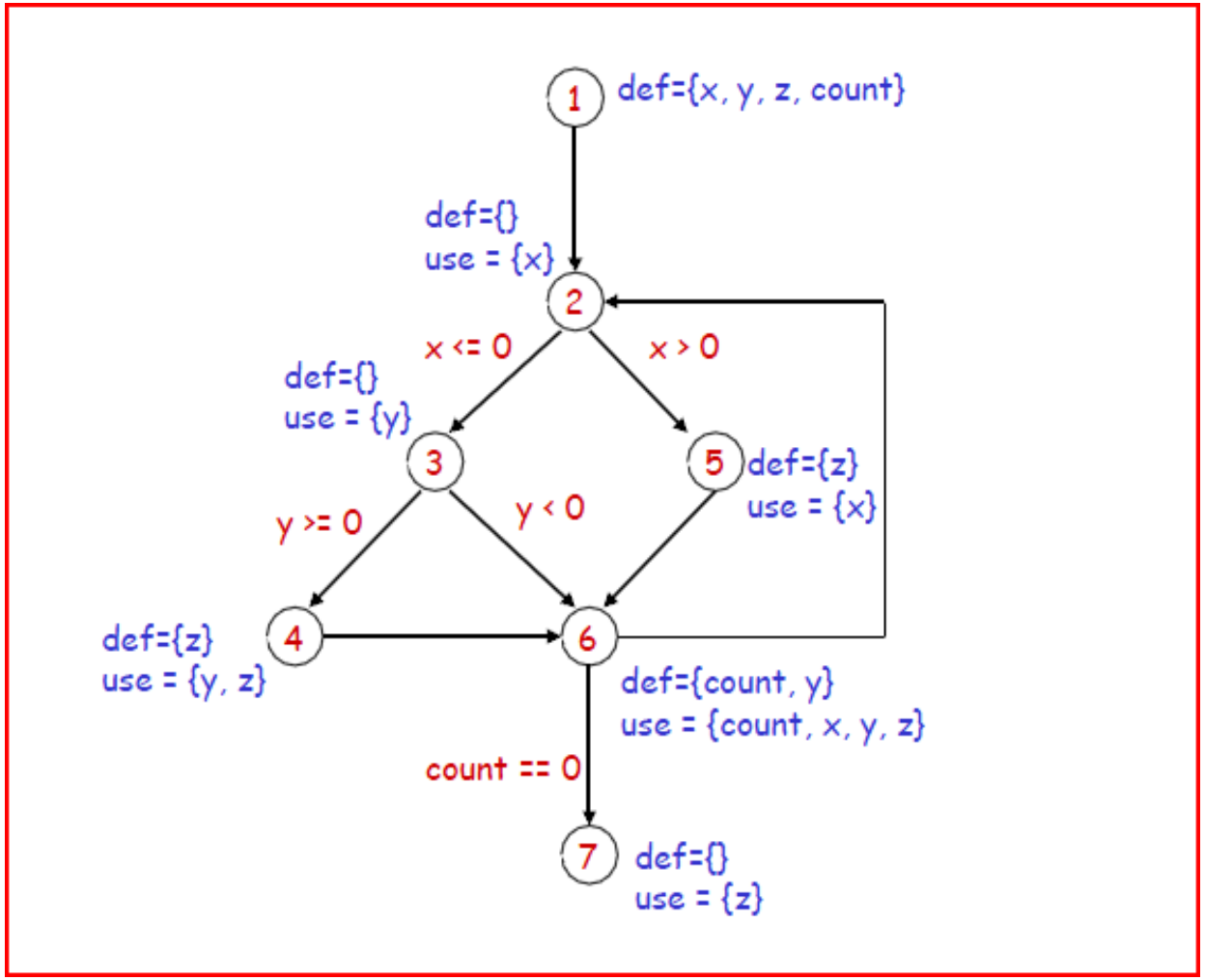


Software engineering

- During Requirements Specification, Data Flow diagrams are used where vertices represent transformations and edges represents the data flows.
- During Design phase, graphical design is used for describing relations among modules;
- During Testing, the control flow of a program associated with McCabe's complexity measure which employs directed graphs for addressing the sequence of executed instructions and etc.
- Control Flow Graph, Data Flow Graph, etc..

Data Flow Graph

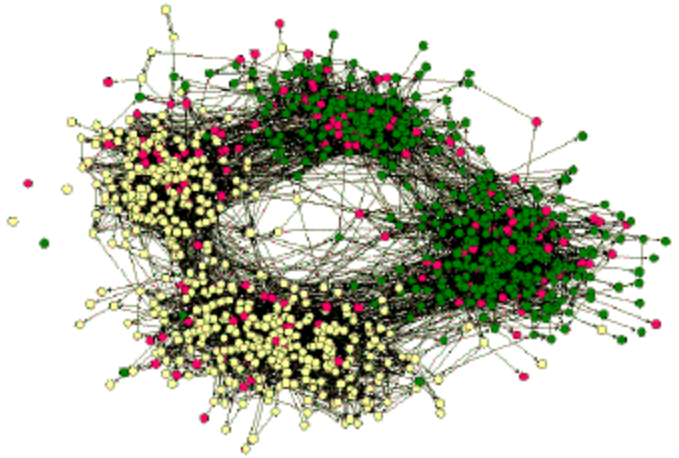
```
1.  begin
2.    float x, y, z = 0.0;
3.    int count;
4.    input (x, y, count);
5.    do {
6.      if (x <= 0) {
7.        if (y >= 0) {
8.          z = y * z + 1;
9.        }
10.     }
11.     else {
12.       z = 1/x;
13.     }
14.     y = x * y + z;
15.     count = count - 1;
16.   while (count > 0)
17.   output (z);
18. end
```



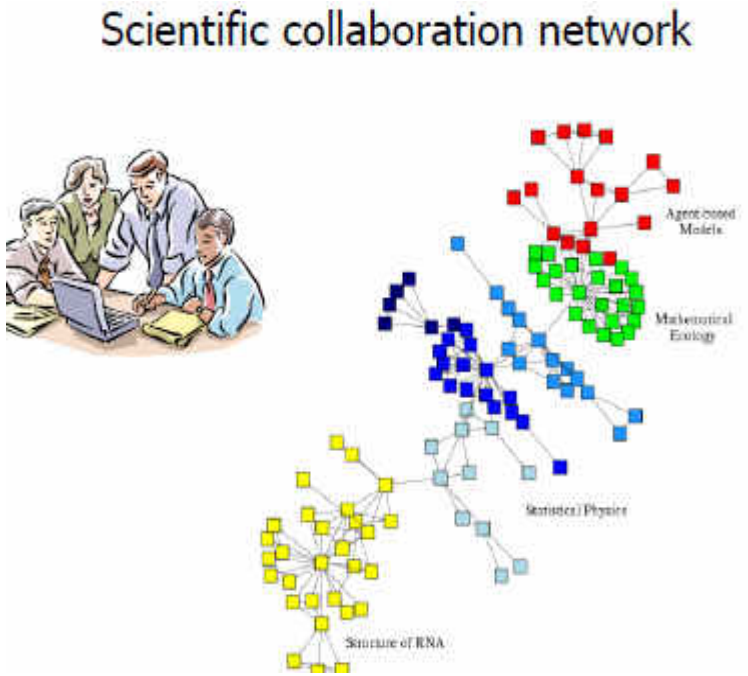
Network System

- The term graph and network are equal. Both refer to a type of structure in which there exists vertices (i.e. nodes, dots) and edges (i.e. links, lines).
- There are numerous types of graphs and networks which yield more or less structure

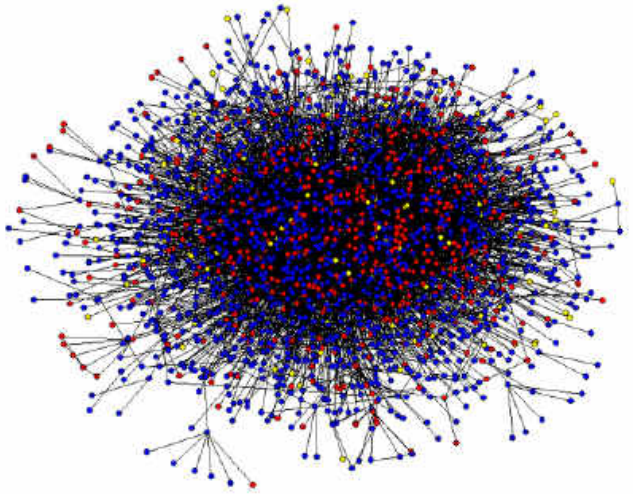
Friendship Network



Scientific collaboration network



Protein-Protein Interaction Networks



Computer hardware

- Graph theory concepts are used in hardware world to provide:
  - Register allocation by graph coloring
  - Representation of instruction sequences by graphs by adjacency matrix
  - In instruction parallel processing
  - Process of allocation scheduling

Data structure

- The logical or mathematical model of a particular organization of data is called a "data structure". The choice of data model depends on two considerations:
  - It must be rich enough in structure to mirror actual relationship of data in real world.
  - The structure should be simple enough that one can effectively process data when necessary.

These two considerations is fulfilled by the graph theoretical concepts.

Image processing

- The applications of graphs in image processing are:
  - To find edge boundaries using graph search algorithms in segmentation.
  - To calculate the alignment of the picture
  - Finding mathematical constraints such as entropy by using minimum spanning tree.
  - Finding distance transforms of the pixels and calculates the distance between the interior pixels by using shortest path algorithms.

adjacency relations: pixels and their neighbors

Data mining

- Graph mining represents the relational aspect of data.
- There are five theoretical based approaches of graph based data mining.
- They are sub graph categories, sub graph isomorphism, graph invariants, mining measures and solution methods

Operating system

- Many practical problems can be solved with the help of graph in the field of operating system such as job scheduling and resource allocation problems.

- For example graph coloring concept can be applied in job scheduling problems of CPU, jobs are assumed as vertices of the graph and there will be an edge between two jobs that cannot be executed simultaneously and there will be one to one relationship between feasible scheduling of graphs

Website designing

- There are many advantages of using graph representation in website development such as:
  - Searching and community discovery.
  - Graph representation (directed graph) in web site utility evaluation and link structure.
  - Finding all connected component and provide easy detection.

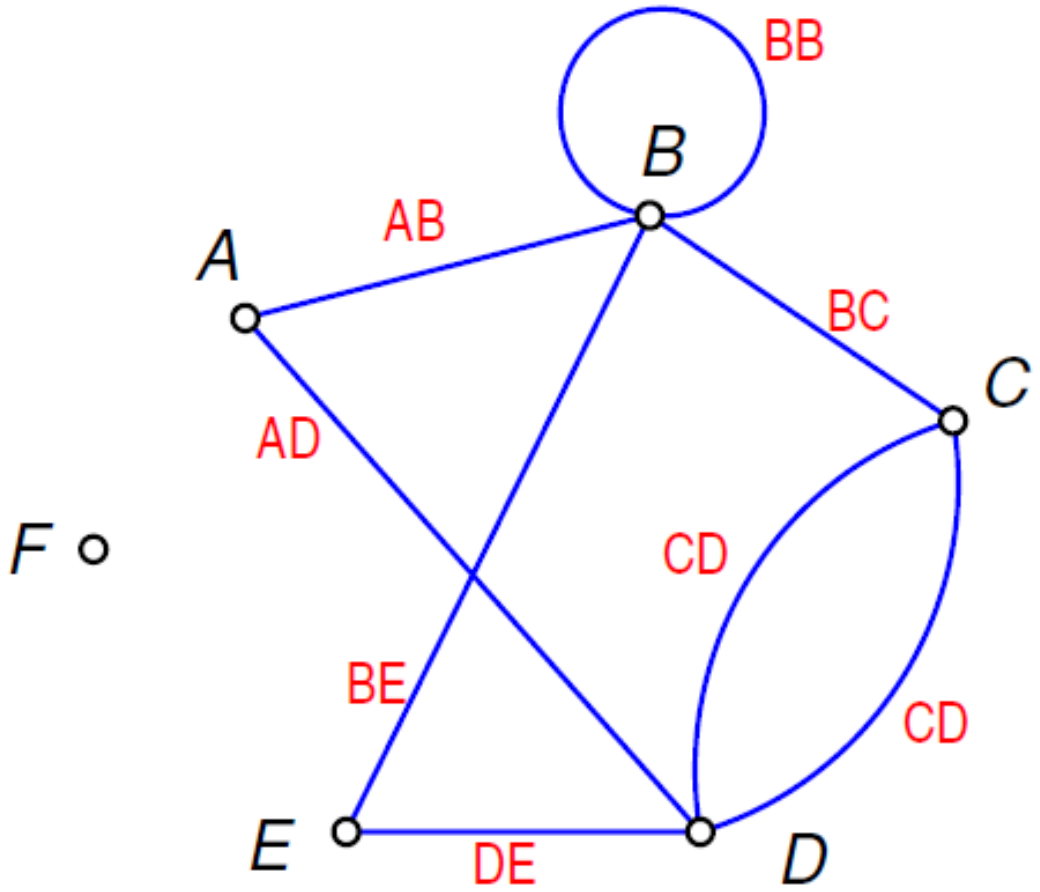
Graphs

- Graphs provide a convenient way to represent various kinds of mathematical objects.
- Essentially, any graph is made up of two sets:
  - 1- A set of vertices
  - 2- A set of edges.
- Graph **G** = (**V**, **E**)
  - **V** = set of vertices
  - **E** = set of **e** edges

Graphs

- **In mathematics**
  - A **graph G** is composed by a set **N** of **vertices**, or nodes, connected through a set **E** of **edges**, or links.
- **In computing**
  - A **graph** is an **abstract data structure** that implements the **mathematical definition** of a graph.

Graph Terminology



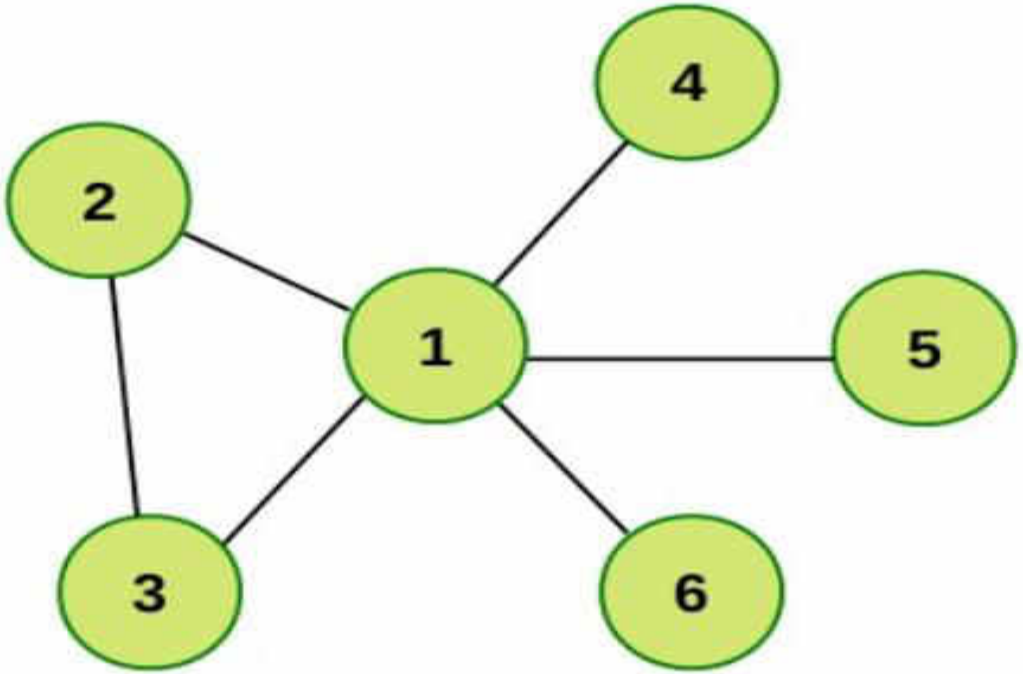
Vertex set:

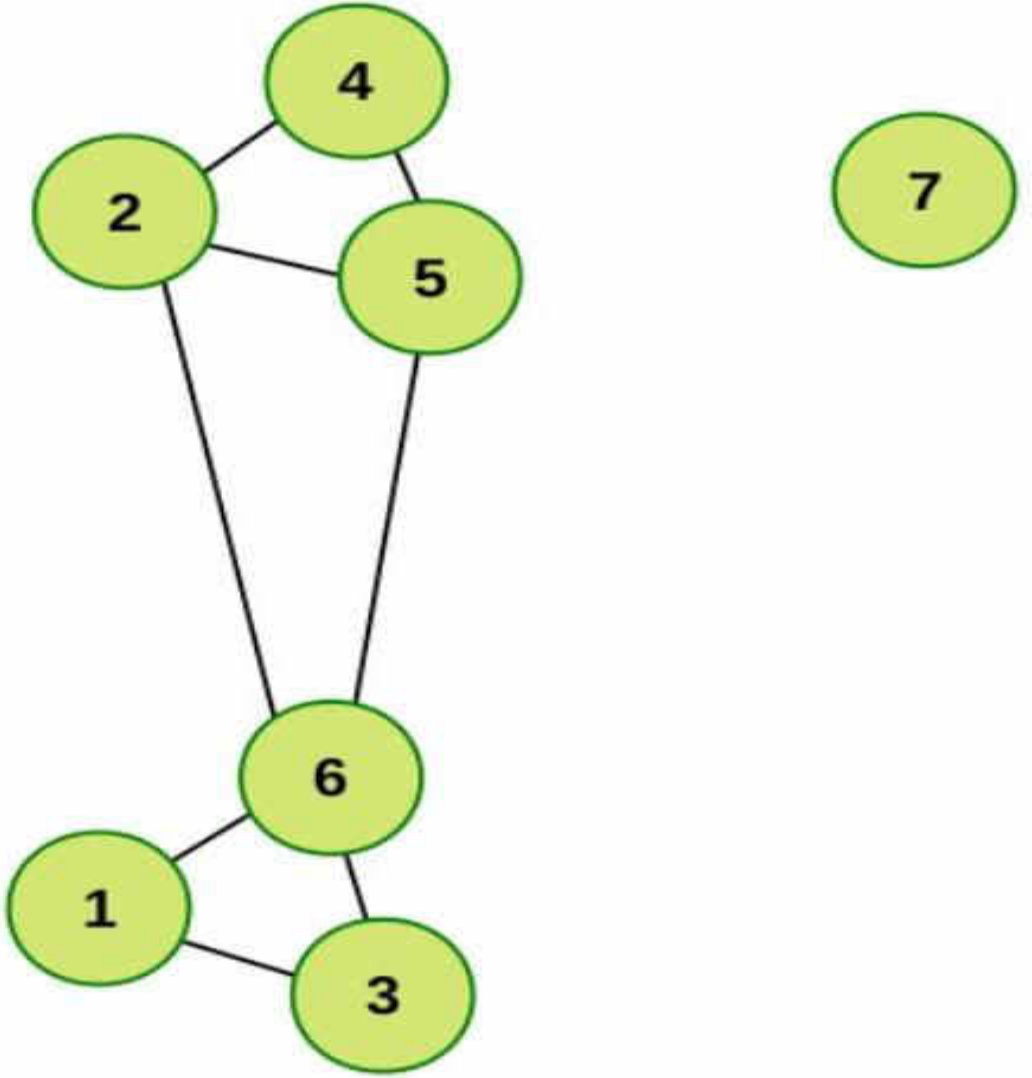
$$\mathcal{V} = \{A, B, C, D, E, F\}$$

Edge set:

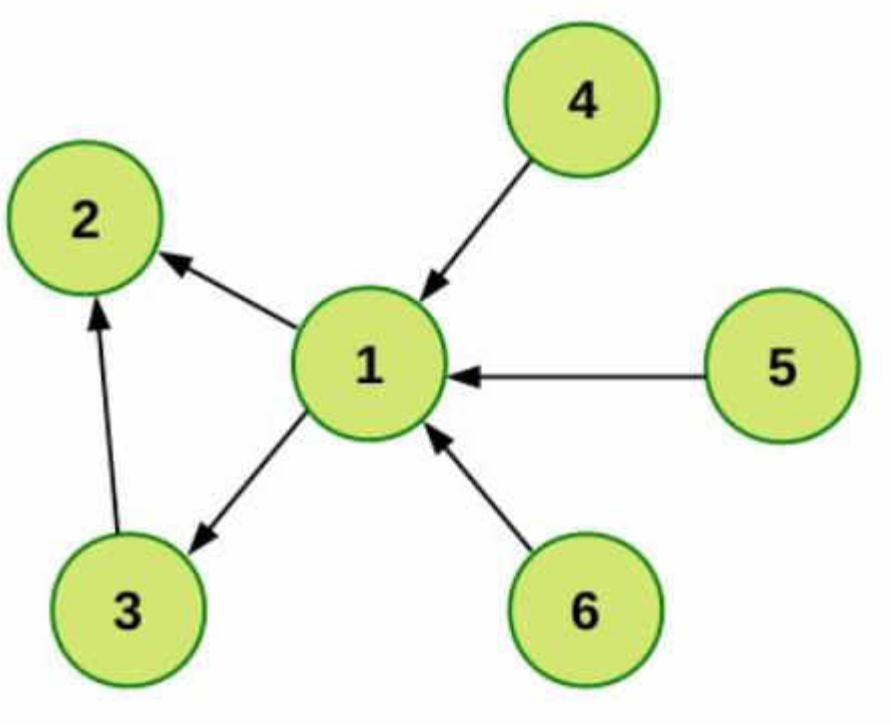
$$\mathcal{E} = \{AB, BB, BC, CD, CD, DE, BE, AD\}$$

Graph Examples





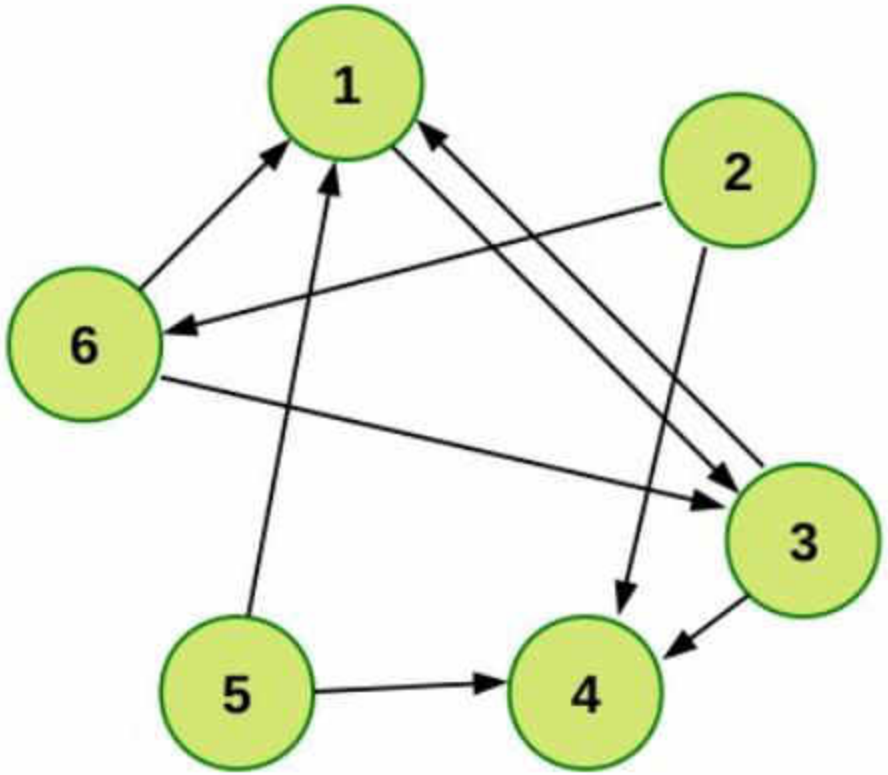
Directed & Undirected Graph



Undirected graph:

edge (**u**, **v**) = edge (**v**, **u**)

no self-loops

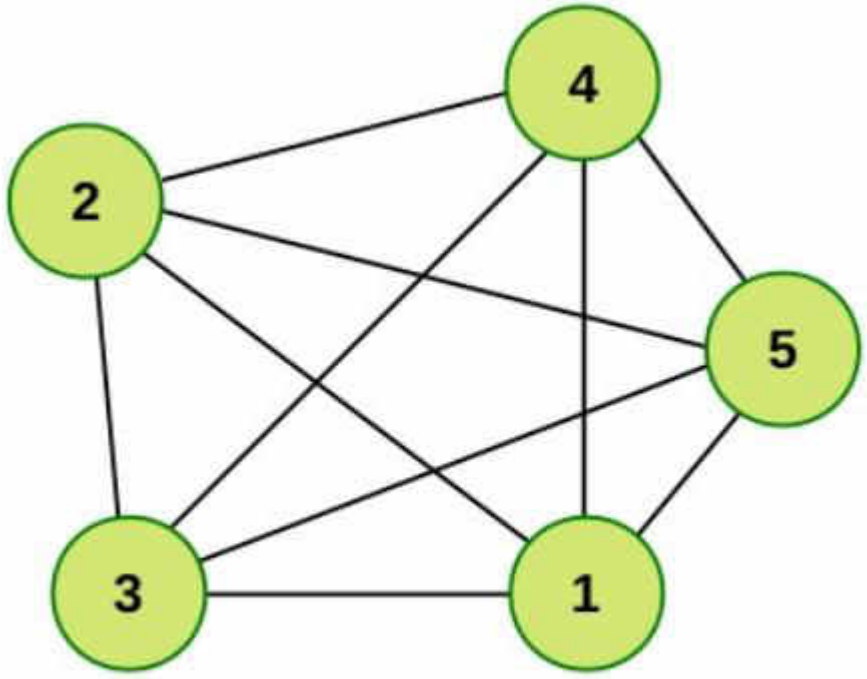


Directed graph (digraph):

edge (**u**, **v**) goes from vertex **u** to vertex **v**

Complete Graph

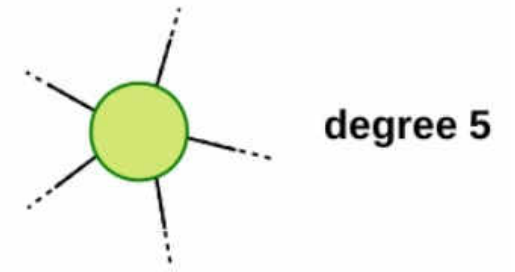
- Every pair of node is connected with an edge



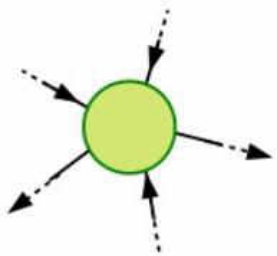
complete graph

Degree

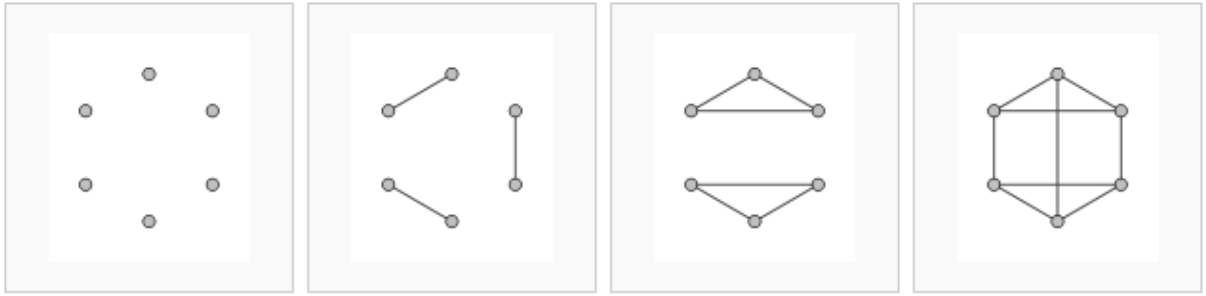
- Degree (Valence) of a Vertex
  - Is the number of adjacent edges
  - In directed graph each node contains and in-degree and out-degree
  - In regular graph each vertex represent the same degree



degree 5

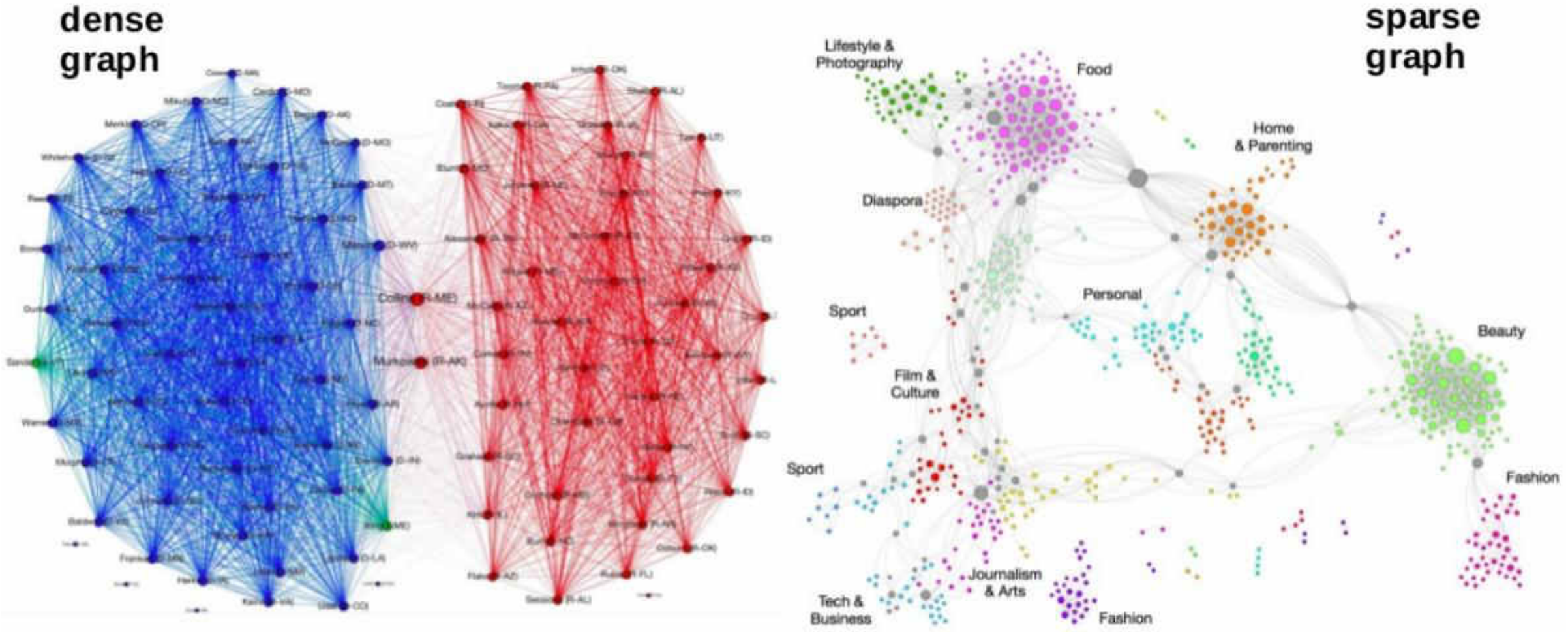


in-degree 3
out-degree 2



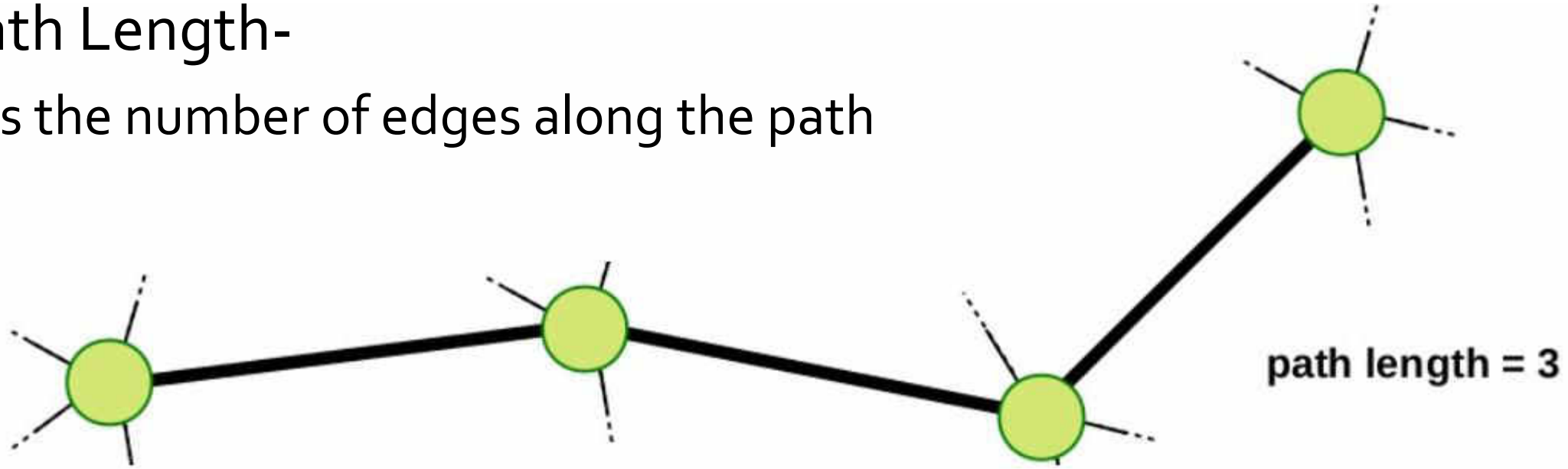
0-regular graph | 1-regular graph | 2-regular graph | 3-regular graph

Sparse or Dense Graph



Path

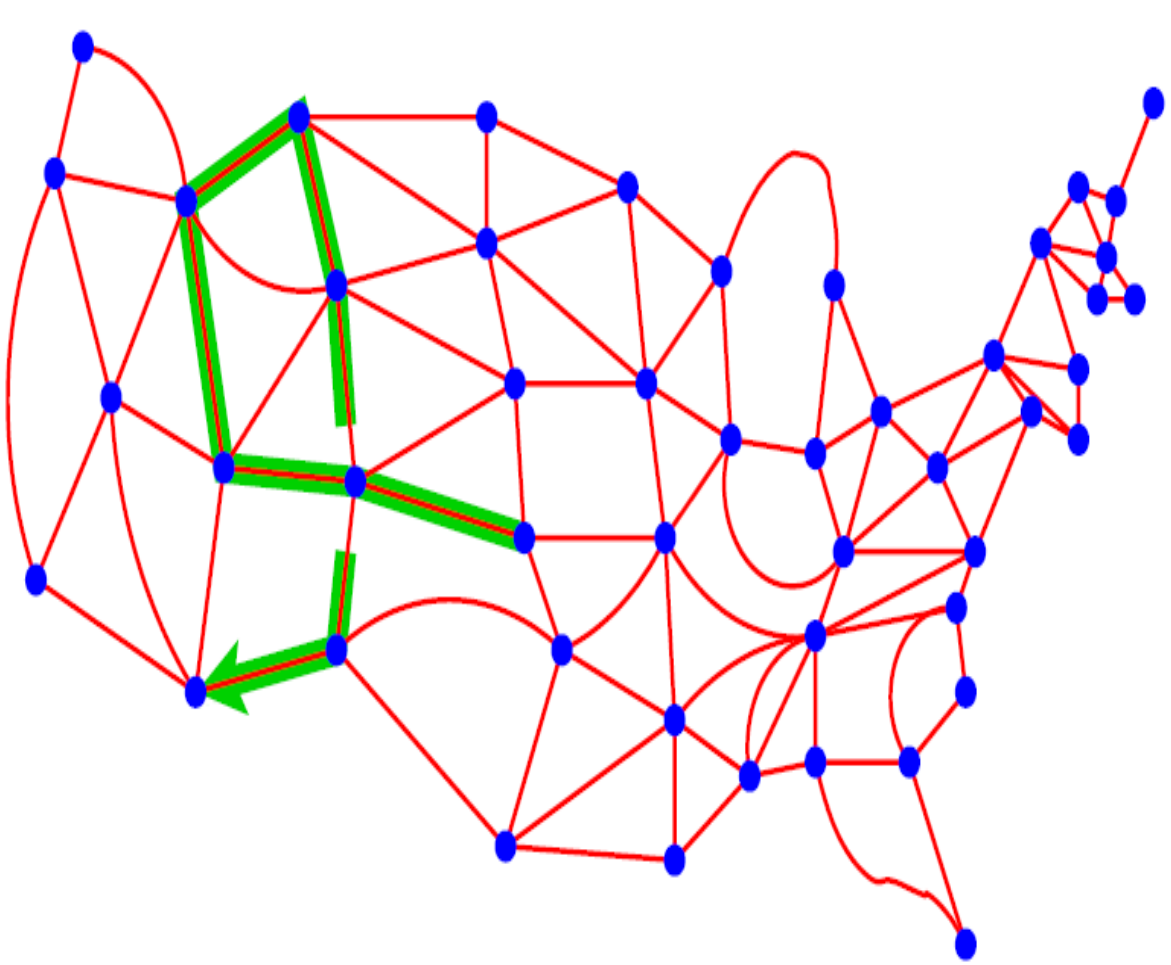
- In a sequence of nodes where adjacent nodes are connected with an edge.
- In a directed graph, the edge must be directed according to the direction of the path.
- Path Length-
  - Is the number of edges along the path

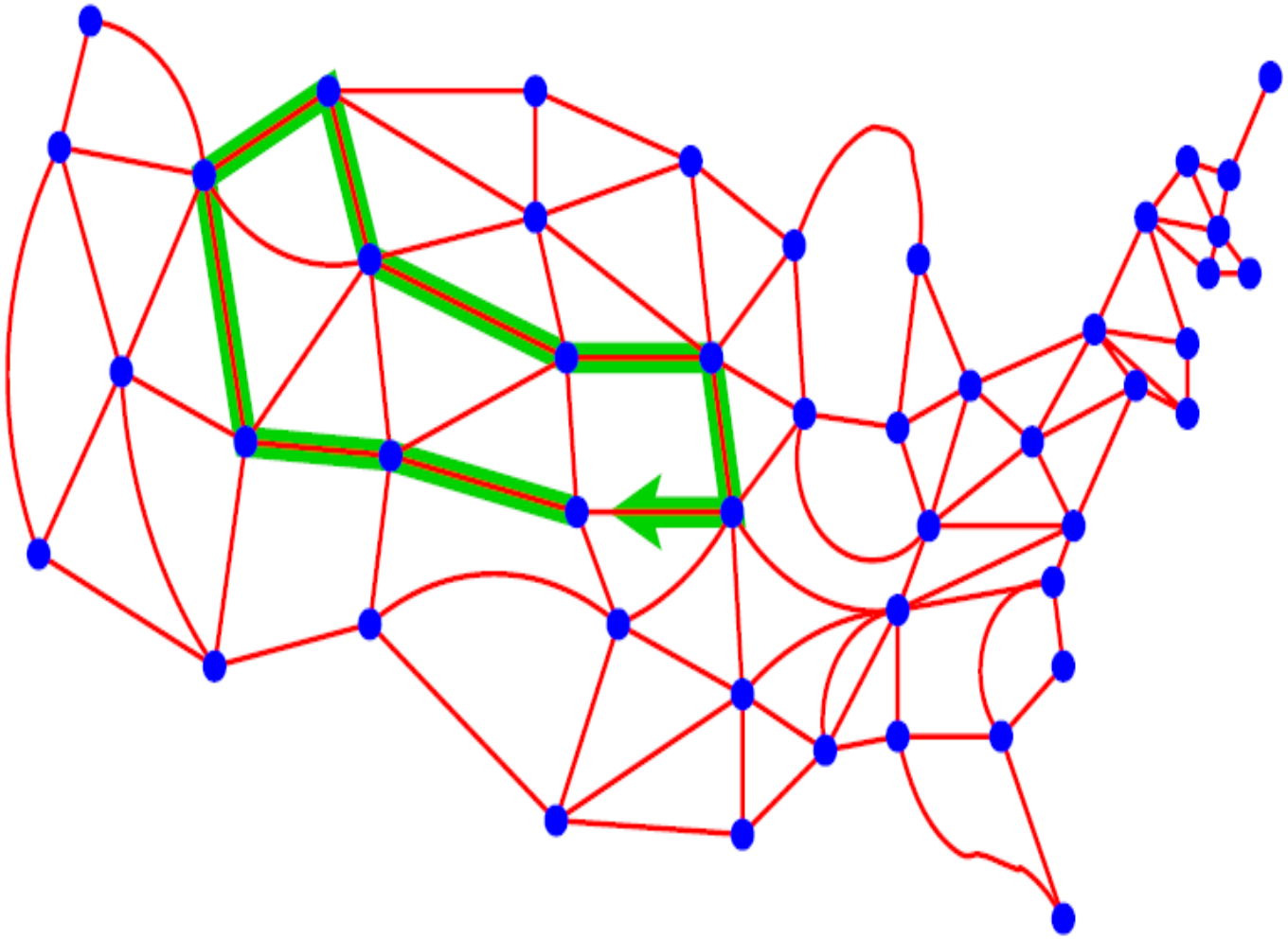


Paths and Circuits

- Rule: A trip cannot use the same edge more than once, but it may pass through the same vertex more than once.
- A trip is called a path if its starting and ending vertices are different.
- It is called a circuit if the starting and ending vertices are the same.

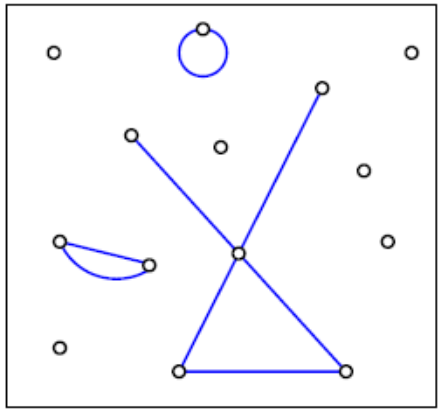
Identify Paths and Circuits



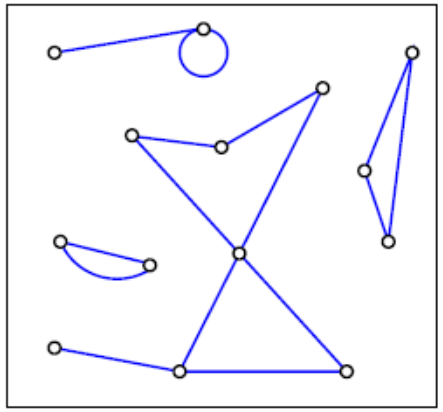


Connectedness

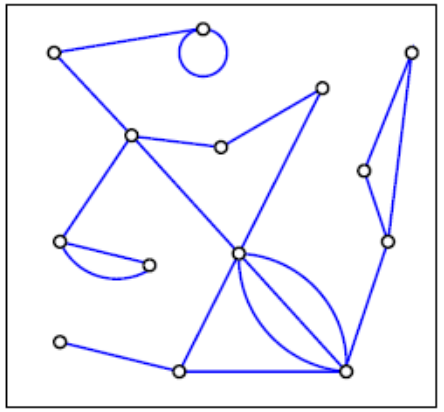
- A graph is called connected if any two vertices can be linked by a path. Otherwise, it is disconnected.
  - *Connected* is usually associated with undirected graphs (two way edges): there is a **path** between every two nodes.



Isolated vertices
Disconnected



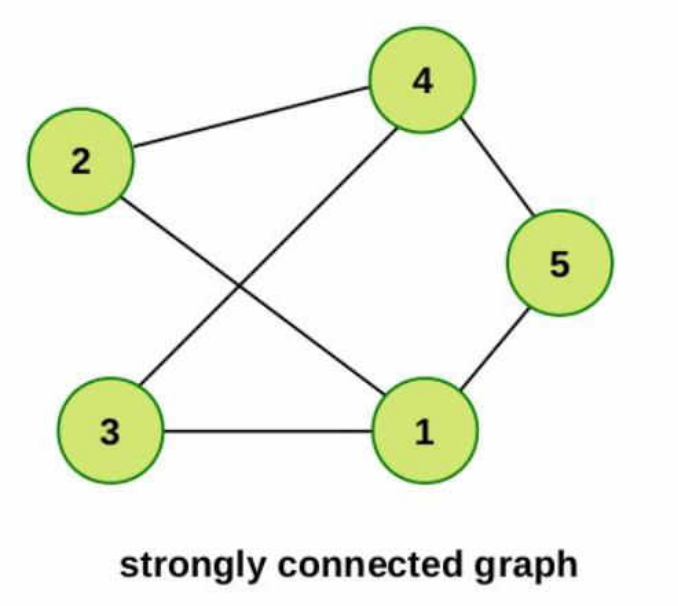
No isolated vertices
Disconnected



Connected

Strongly Connected Graph

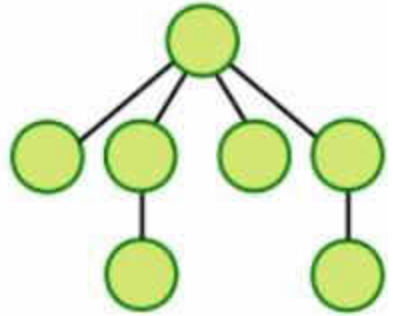
- Strongly Connected Graph represent a path between every pair of nodes
  - *Strongly connected* is usually associated with directed graphs (one way edges): there is a **route** between every two nodes.



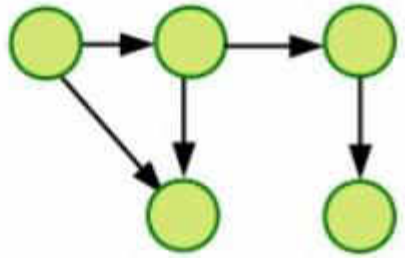
strongly connected graph

Cyclic or Acyclic

- Connected acyclic undirected graph: trees
  - A tree is an undirected graph in which any two vertices are connected by exactly one path. In other words, any connected graph without simple cycles is a tree.
  - A forest is a disjoint union of trees.

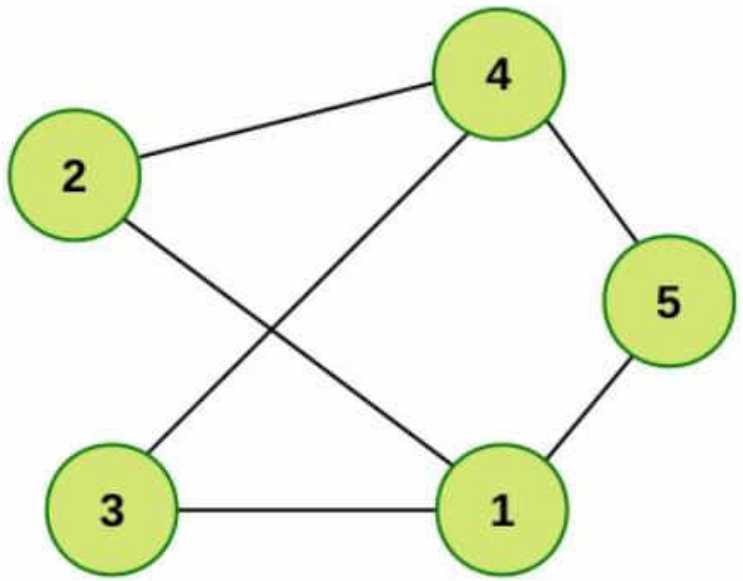


DAG

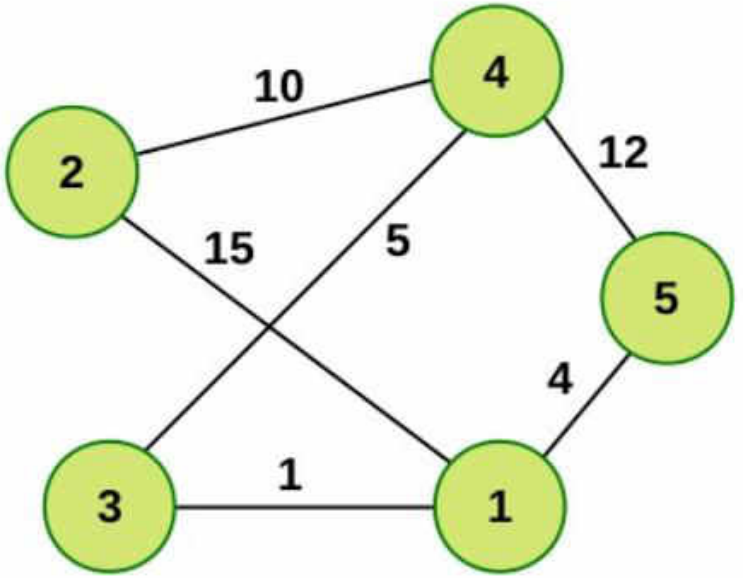


- Directed Acyclic Graph (DAG)

Unweighted & Weighted

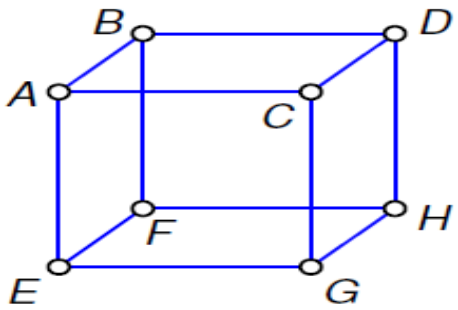


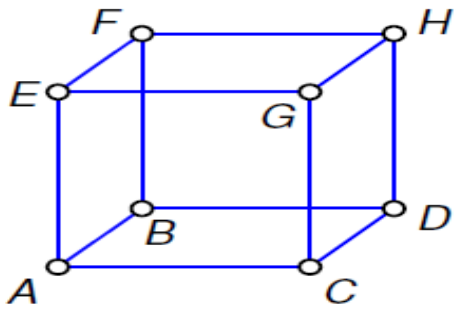
unweighted graph

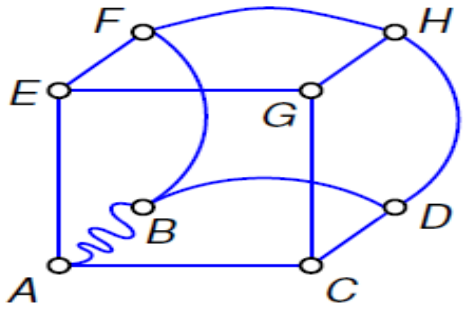


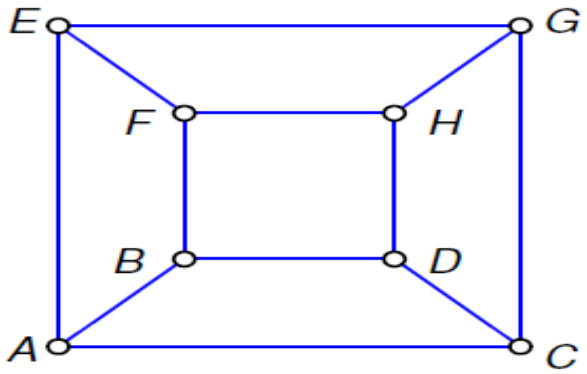
weighted graph

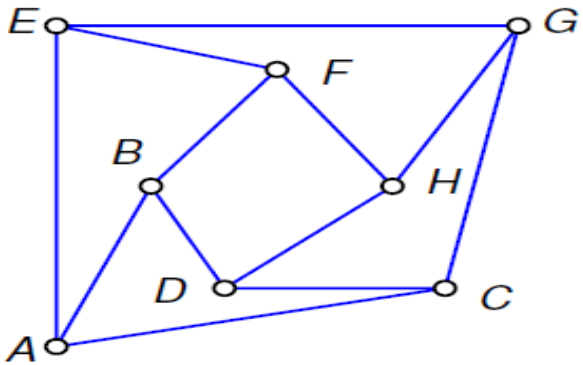
Different Representations of Graphs











These five figures all represent the same graph!

Recognizing and Representing a Graph

Recognizing a graph problem

- The first key to solving a graph related problem is recognizing that it is a graph problem.
- Nearly all graph problems will somehow use a grid or network in the problem, but sometimes these will be well masked.
- Secondly, if you are required to find a path of any sort, it is usually a graph problem as well
- Some common keywords associated with graph problems are:
  - vertices, nodes, edges, connections, connectivity, paths, cycles and direction.

An example

- Bob has become lost in his neighborhood/area.
- He needs to get from his current position back to his home.
- Bob's neighborhood is a 2 dimensional grid, that starts at (0, 0) and (width – 1, height – 1).
- There are empty spaces upon which Bob can walk with no difficulty, and houses, which Bob cannot pass through.
- Bob may only move horizontally or vertically by one square at a time.

An example

- Bob's initial position will be represented by a 'B' and the house location will be represented by an 'H'.
- Empty squares on the grid are represented by 'o' and houses are represented by 'X'.
- Find the minimum number of steps it takes Bob to get back home, but if it is not possible for Bob to return home, return -1.

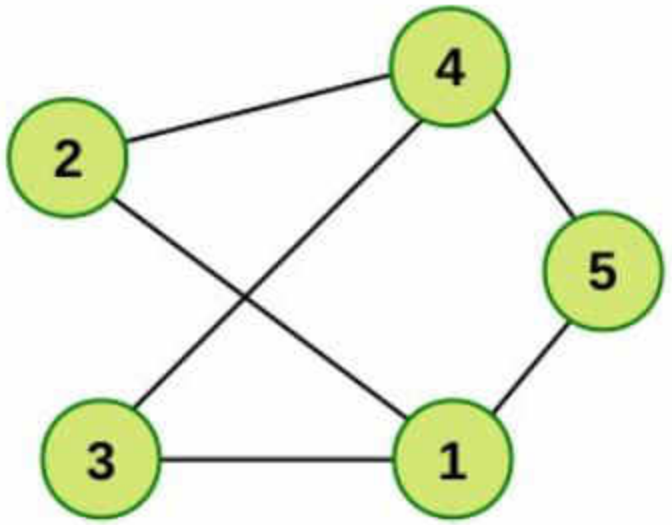
o	o	o	X	o	o	X	o	o	B
o	o	o	o	o	o	o	o	o	o
o	X	o	X	X	o	X	o	X	o
o	o	X	o	o	o	X	o	o	o
o	o	H	o	X	o	X	o	X	o
o	X	o	o	o	X	o	o	o	X
o	o	X	o	X	o	o	X	o	o
o	o	o	o	o	o	o	o	o	o

Graph Implementation

Once you have recognized that the problem is a graph problem it is time to start building up your representation of the graph in memory.

Adjacency Matrix

- Matrix is used to store the connectivity information
- The Matrix is symmetric for undirected graph.



graph

	1	2	3	4	5
1	0	1	1	0	1
2	1	0	0	1	0
3	1	0	0	1	0
4	0	1	1	0	1
5	1	0	0	1	0

adjacency matrix

Space cost:
$O(V^2)$!

Array representation

graph

.numVertices 7

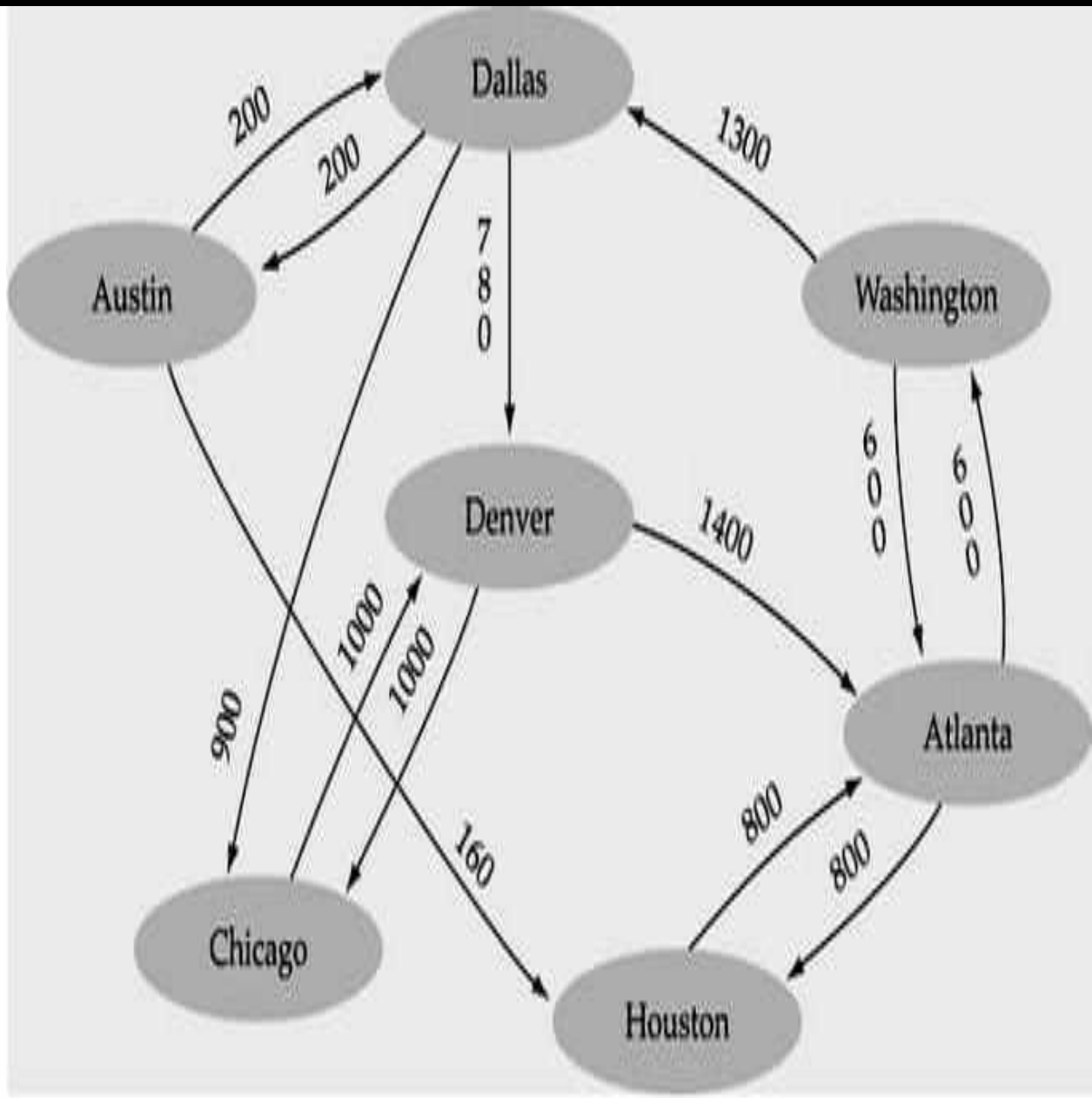
.vertices

[0]	"Atlanta"
[1]	"Austin"
[2]	"Chicago"
[3]	"Dallas"
[4]	"Denver"
[5]	"Houston"
[6]	"Washington"
[7]	
[8]	
[9]	

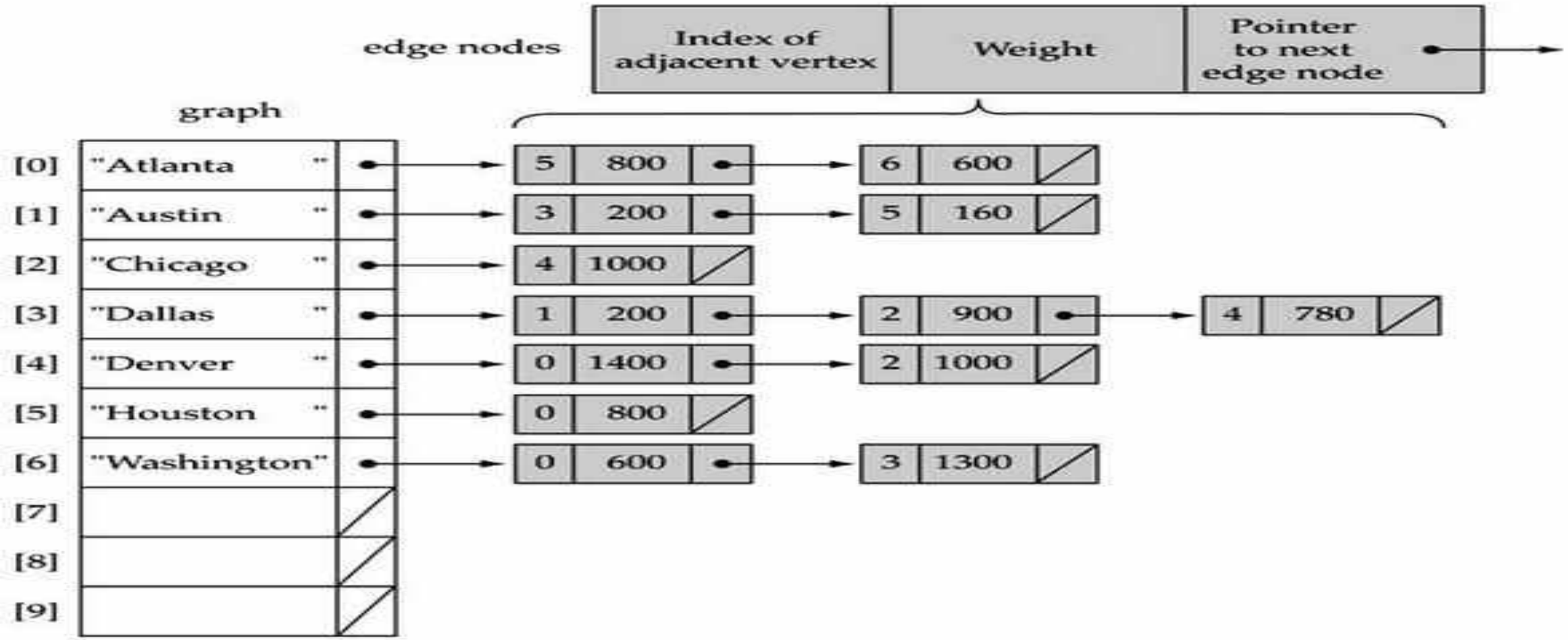
.edges

	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]
[0]	0	0	0	0	0	800	600	•	•	•
[1]	0	0	0	200	0	160	0	•	•	•
[2]	0	0	0	0	1000	0	0	•	•	•
[3]	0	200	900	0	780	0	0	•	•	•
[4]	1400	0	1000	0	0	0	0	•	•	•
[5]	800	0	0	0	0	0	0	•	•	•
[6]	600	0	0	1300	0	0	0	•	•	•
[7]	•	•	•	•	•	•	•	•	•	•
[8]	•	•	•	•	•	•	•	•	•	•
[9]	•	•	•	•	•	•	•	•	•	•

(Array positions marked '•' are undefined)

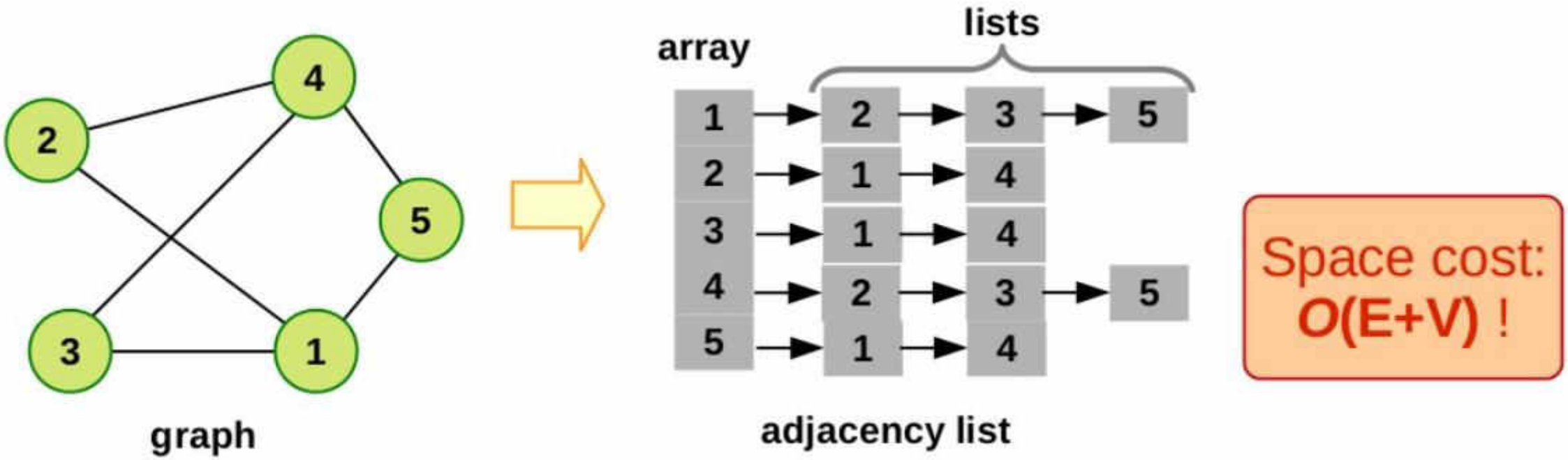


Linked-list representation



Adjacency List

- Lists are used to store the connectivity information



Comparisons

Comparison	Winner
Faster to test if (x,y) is in the graph?	Adjacency matrices.
Faster to find the degree of a vertex?	Adjacency lists.
Less memory on small graphs?	Adjacency list (E + V) vs. (V^2).
Less memory on big graphs?	Adjacency matrix (small win).
Edge insertion or deletion?	Adjacency matrix O(1) vs. O(L).
Faster to traverse the graph?	Adjacency list $\Theta(E + V)$ vs. $\Theta(V^2)$.
Better for most problems?	Adjacency lists.

Graphs and Their Data Structures

- **Stack:** A stack is one of the simplest data structures available.

- There are four main operations on a stack:
  - Push – Adds an element to the top of the stack
  - Pop – Removes the top element from the stack
  - Top – Returns the top element on the stack
  - Empty – Tests if the stack is empty or not

- In C++, this is done with the STL class stack:

```
#include <stack>
std::stack<int> myStack;
```

Graphs and Their Data Structures

- **Queue :**A queue is a simple extension of the stack data type. Whereas the stack is a FILO (first-in last-out) data structure the queue is a FIFO (first-in first-out) data structure.
- What this means is the first thing that you add to a queue will be the first thing that you get when you perform a pop().
- There are four main operations on a queue:
  - Push – Adds an element to the back of the queue
  - Pop – Removes the front element from the queue
  - Front – Returns the front element on the queue
  - Empty – Tests if the queue is empty or not
- In C++, this is done with the STL class queue:

```
#include <queue>
queue<int> myQueue;
```