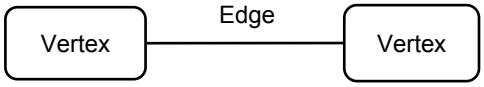
Graphs

You will learn basic principles of graphs as well as how they can be applied to problem solving.

What Is A Graph

- A collection of vertices and edges (sets and relations)

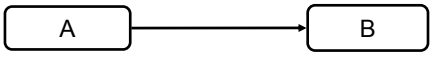


- As shown in the last section, visualizing the set of relations can be easier when shown in graphical rather than textual form.
- Types of graphs
  - Directed
  - Undirected

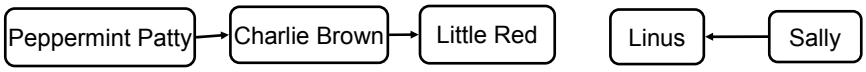
Directed Graphs

•Edges are one-way

- Direction of travel on streets



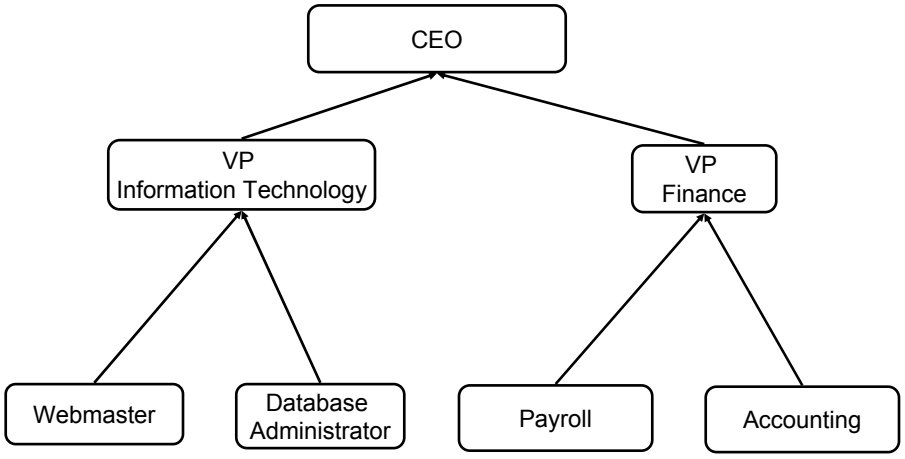
- Relationships ("likes")



V = {Charlie Brown, Peppermint Patty, Little Red, Linus, Sally}
E = {(Charlie Brown, Little Red),
(Peppermint Patty, Charlie Brown),
(Sally, Linus)}

Directed Graphs (2)

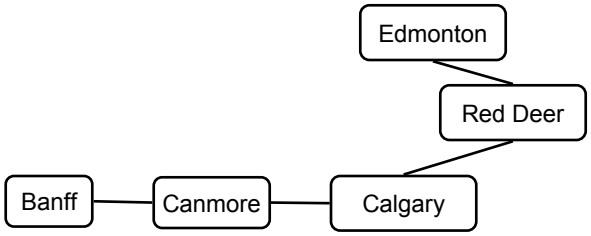
- Organizational structure ("reports to")



Undirected Graphs

•Edges are two-way

- Alberta towns, cities and the highways that connect them.

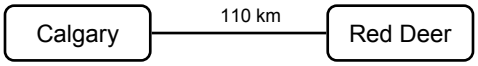


V = {Edmonton, Red Deer, Calgary, Canmore, Banff}
E = {(Banff, Canmore),
(Calgary, Canmore),
(Calgary, Red Deer),
(Edmonton, Red Deer)}

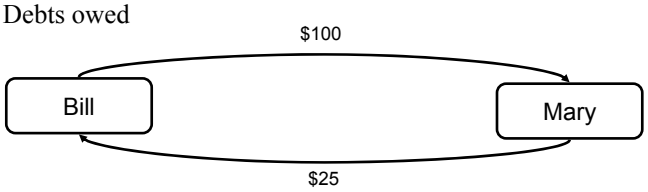
Graphs Can Be Labeled

•The annotations can provide additional information.

- Speed limits

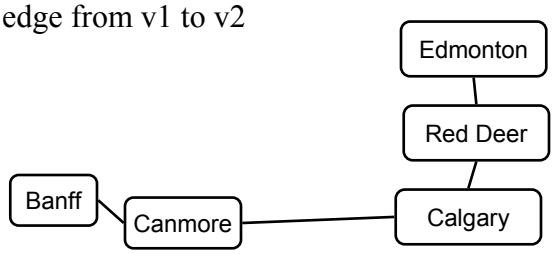


- Debts owed



Adjacent Vertices

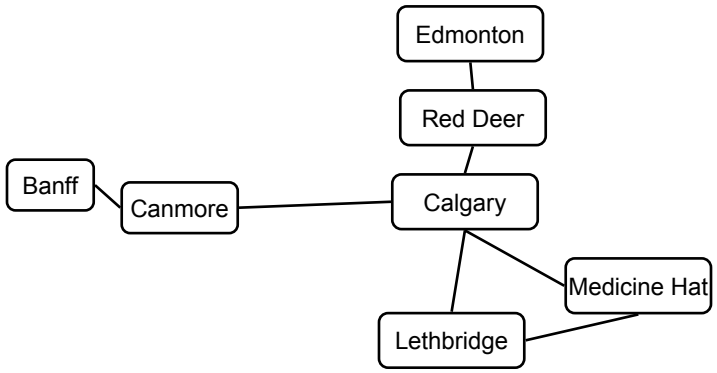
- In a directed graph vertices are adjacent if they are connected by an edge.
- In an undirected graph: v1 is adjacent to v2 if there is a direct edge from v1 to v2



- Adjacent: Banff-Canmore, Canmore-Calgary etc.
- Not adjacent: Red Deer and Canmore etc.

Paths

- There is a path between vertices if they are directly connected by an edge or indirectly connected.

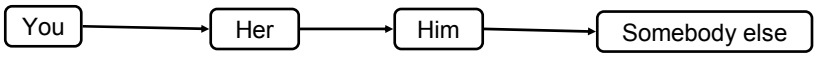


- Red Deer and Lethbridge are not adjacent but there is at least one path from Red Deer to Lethbridge.

Paths (Directed Graphs)

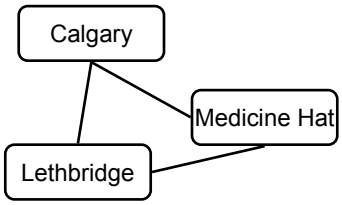
•There is a path from v^1 to v^n (v^1, v^2, v^3..v^{n-1}, v^n) if v^1 is adjacent to v^2, v^2 is adjacent to v^3...v^{n-1} is adjacent to v^n.

- The "Love" graph



Cyclical Paths

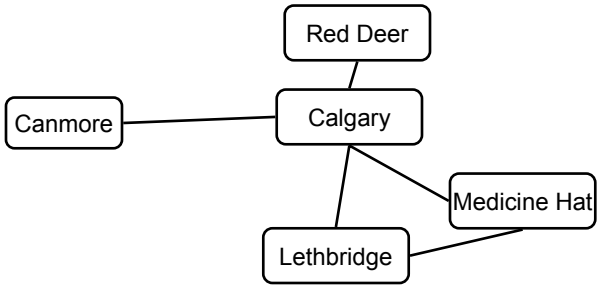
•A path that starts and ends at the same vertex.



One cycle = (Calgary, Medicine Hat, Lethbridge, Calgary)

Degree Of A Vertex

•It's the number of edges that are connected to it.

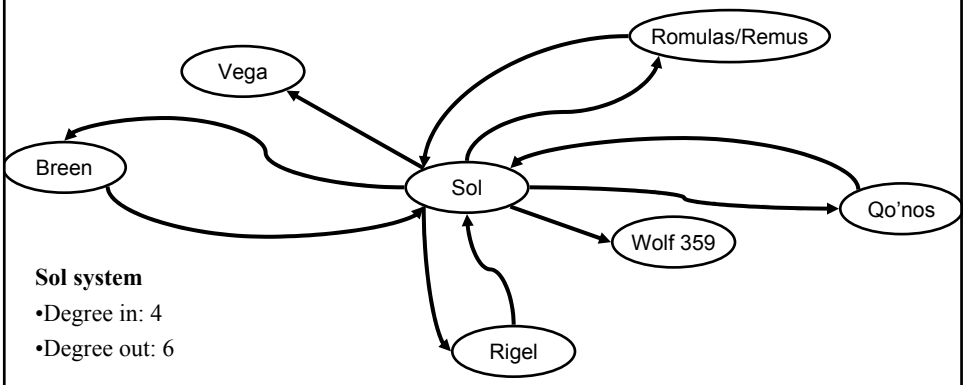


Degree of Calgary = 4
Degree of Medicine Hat, Lethbridge = 2
Degree of Canmore and Red Deer = 1

James Tam

Degree Of A Vertex (Directed Graphs)

•There are two measures of degree for directed graphs.
- In-degree of a vertex: The number of edges that are adjacent to the vertex (arrows in).
- Out-degree of a vertex: The number of edges that are adjacent from the vertex (arrows out).

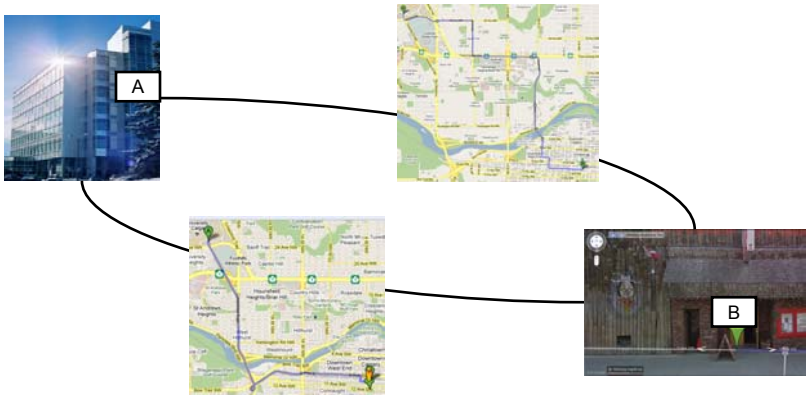


Sol system
•Degree in: 4
•Degree out: 6

James Tam

Multigraphs

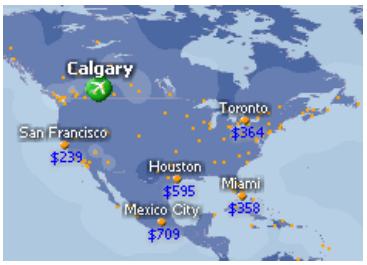
- Most graphs don't allow for multiple edges between a pair of vertices.
- Multigraphs allow multiple edges to connect a pair of vertices.
- Example: path finding when alternatives are possible.



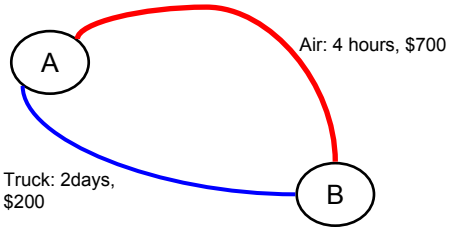
James Tam

Applications Of Graphs

- Logistics and supply (multigraph)



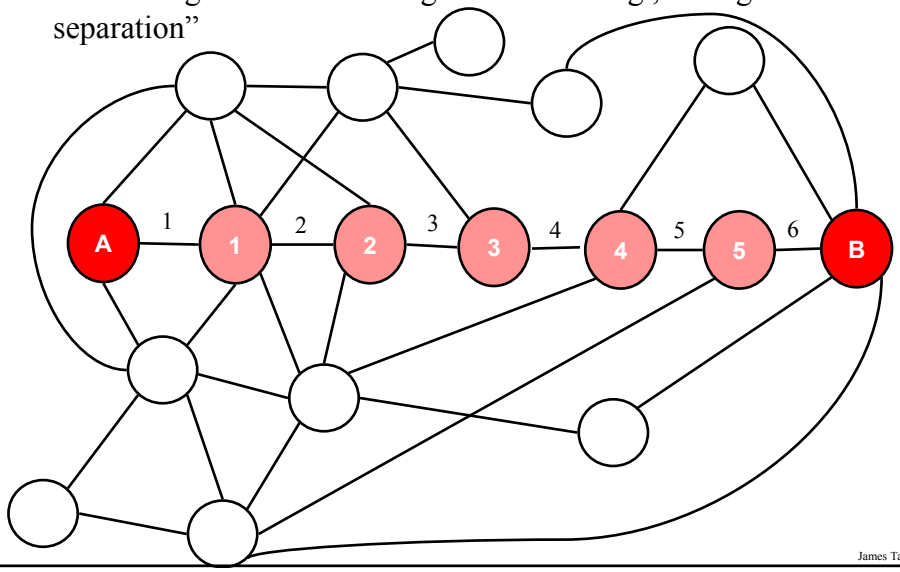
www.skyscanner.com



James Tam

Application Of Graphs (2)

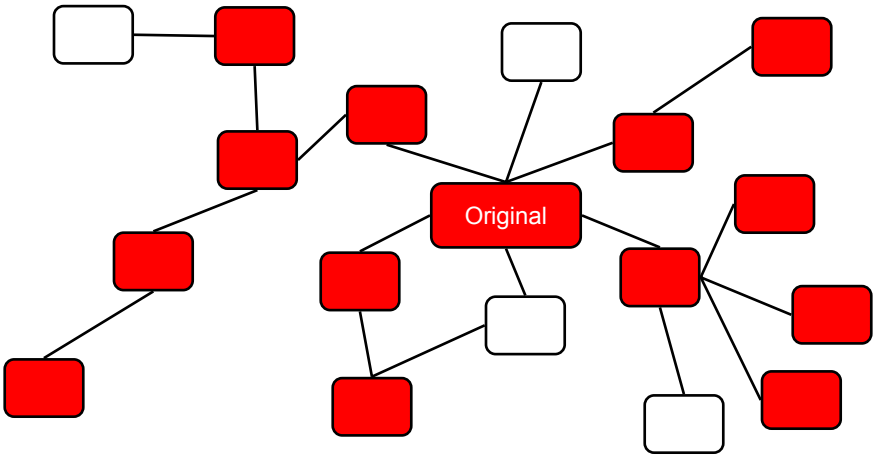
•Visualizing social networking connections e.g., “6 degrees of separation”



James Tam

Application Of Graphs (3)

•Disease transmission: examining which people had intimate contact in order to determine who may have become infected.

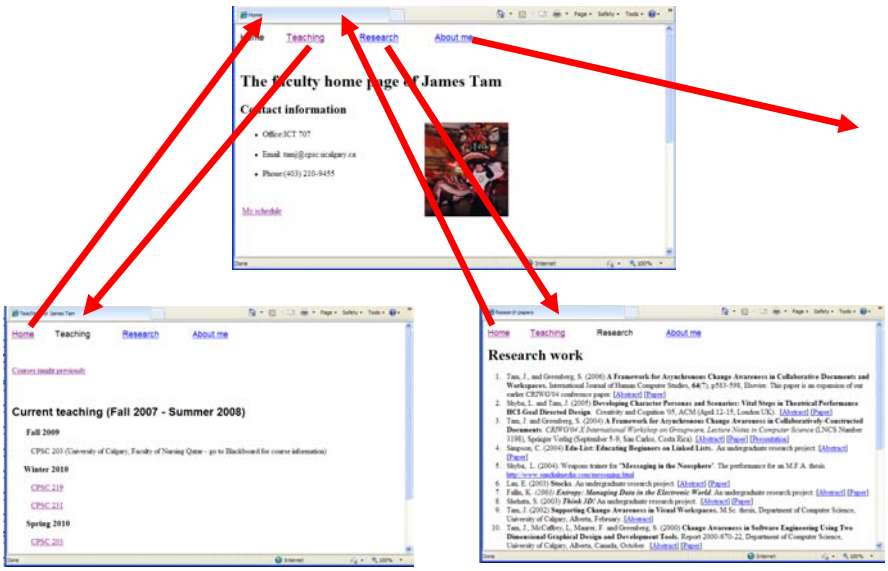


James Tam

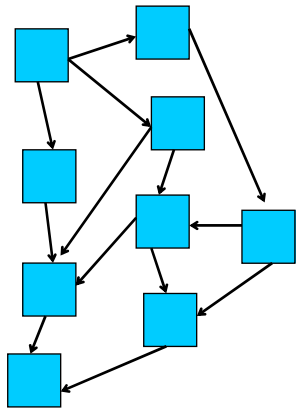
Application Of Graphs (4)

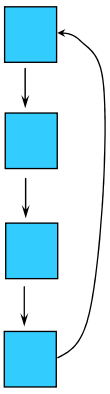
- The World Wide Web itself can be visualized as a directed graph.
  - Vertex = web page, Edge = link between pages.
- Visualizing the layout of a page as a graph can be useful in web design.
  - Are there sufficient connections between pages?
  - Do the connections (links) make sense to visitor?
  - Although it should be possible to reach any page without too much clicking (rule of thumb is 3), there are some pages that should always be accessible e.g., home page, contact page etc.

Application Of Graphs (5)



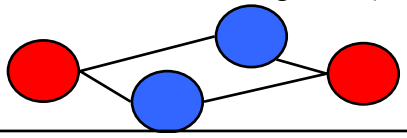
Application Of Graphs (6)





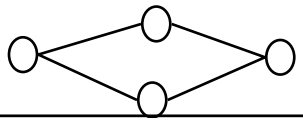
Graph Coloring

- Used for scheduling problems e.g., setting up interview schedules for a group of job applicants among a group of interviewers, setting exam schedules to avoid conflicts.
- Events are drawn as vertices.
- When events occur at the same time, the vertices are made adjacent.
- The idea is to color the vertices of the graph so that adjacent nodes aren't the same color. (Color's represent time slots or some other resource that may be needed simultaneously).
- Applying the algorithm correctly will result in the minimum number of colors being used (minimizes the use of resources).



Graph Coloring Algorithm

- Input: Undirected graph
- Output: Colored graph
- Steps:
  1. Select an uncolored vertex v and color it with a new color c.
  2. For each uncolored vertex:
     a. If the vertex is adjacent to a vertex that is already colored with color c skip v with color c.
     b. If the vertex is not adjacent to a vertex that is already colored with color c, color vertex v with color c.
     c. Repeat step 2 until all uncolored vertices have been checked.
  3. Check if there are any uncolored vertices:
     a. If there are any uncolored vertices then go to step #1
     b. If there aren't any uncolored vertices then the algorithm is finished (stop coloring).



James Tam

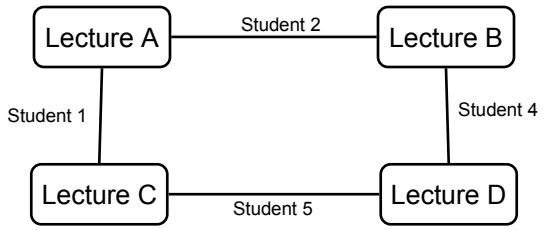
Graph Coloring Example 1: Final Exams

- You are to schedule final exams so that a student will not have two final exams scheduled at the same time.
- Although you could schedule each exam in it's own individual time slot (i.e., no two exams run simultaneously) this would be highly inefficient.
  - Another constraint is to schedule exams with the minimum number of time slots.
- Examinations for a lecture will be represented with vertices.
- A pair of vertices will be connected if there is at least one student who is registered in both.

James Tam

Graph Coloring Example 1: Final Exams (2)

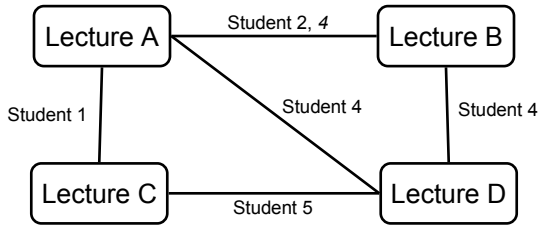
- Lecture A:
  - Student 1, student 2
- Lecture B:
  - Student 2, student 4, student 6
- Lecture C:
  - Student 1, student 3, student 5
- Lecture D:
  - Student 4, student 5



James Tam

Graph Coloring Example 2: Final Exams

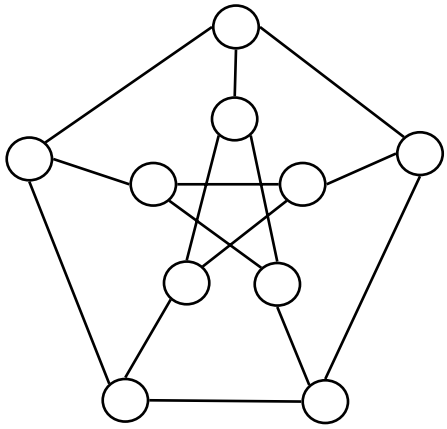
- Lecture A:
  - Student 1, student 2, *student 4*
- Lecture B:
  - Student 2, student 4, student 6
- Lecture C:
  - Student 1, student 3, student 5
- Lecture D:
  - Student 4, student 5



James Tam

Graph Coloring Example 3

- Apply the graph coloring algorithm to the following Petersen graph[1] so that the minimum number of colors is used.



1 Developed by Julius Peterson 1898

You Should Now Know

- What is a graph, vertex and edge
- The difference between a directed and an undirected graph
- What does it mean for vertices to be adjacent
- What is a path (directed and undirected graphs)
- What is a cycle
- The degree of a vertex (directed and undirected graphs)
- What is a multi-graph (how it differs from a regular graph)
- How can graphs can be applied to some everyday situations
- How to apply the graph coloring algorithm to solve scheduling problems