COT 2000

Foundations of Computing

Spring 2024

Lecture 19 – part 1

Lab 9
Homework 6 – 07/19/24
HW 7, Exam 4

Lecture 19 – part 2

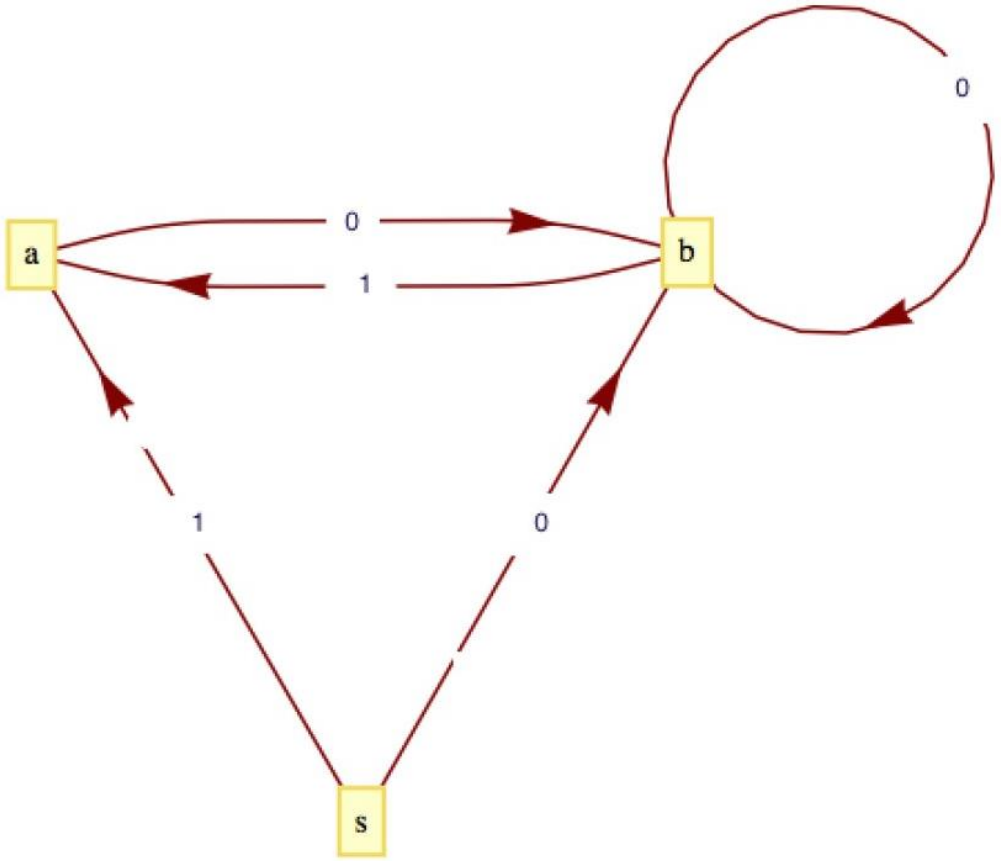
Review

Review

- Counting and Probability
- Graph Theory
- Simple Directed Graph
- Simple Undirected Graph
- Multigraph

Simple Directed Graph

- A simple directed graph consists of a nonempty **set of vertices**, V , and a **set of edges**, E, that is a subset of the set $V \times V$.
- Each edge is an ordered pair of elements from the vertex set.
- The first entry is the **initial vertex** of the edge and the second entry is the **terminal vertex**.

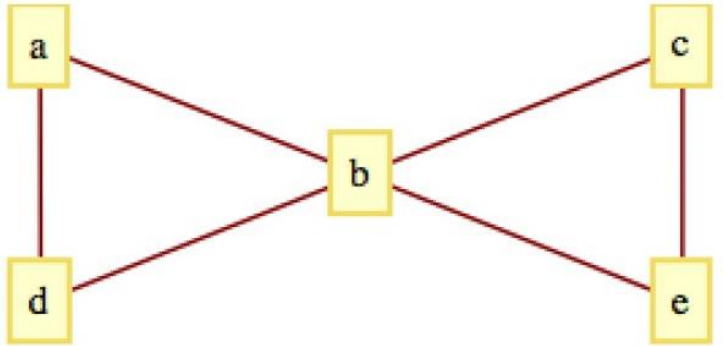


$V = \{s, a, b\}$

$E = \{(s, a), (s, b), (a, b), (b, a), (b, b)\}$.

Simple Undirected Graph

- The order of the vertices is not significant.
- A simple undirected graph consists of a nonempty set V , called a vertex set, and a set E of two-element subsets of V , called the edge set.
- When drawing an undirected graph, the two-element subsets are drawn as undirected lines or arcs connecting the vertices.
- It is customary to not allow “self loops” in undirected graphs since $\{v, v\}$ isn’t a two element subset of vertices.
- Both directed and undirected graphs have nonempty vertex sets.



$V = \{a, b, c, d, e\}$

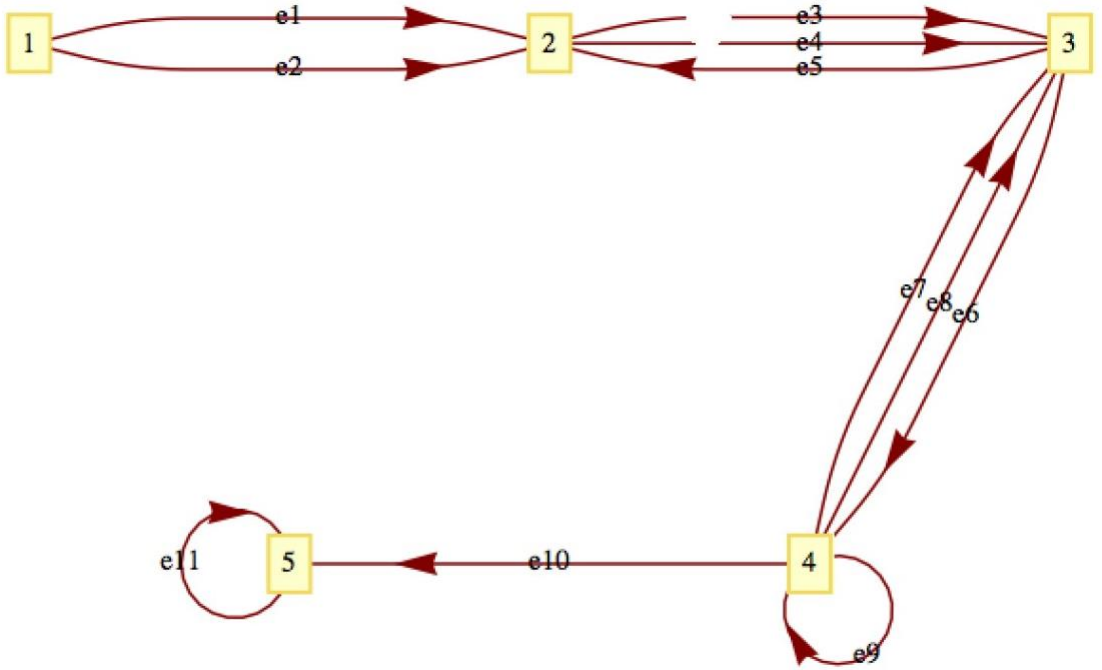
$E = \{\{a, b\}, \{a, d\}, \{b, c\}, \{b, d\}, \{c, e\}, \{b, e\}\}$

Complete Undirected Graph. A complete undirected graph on n vertices is an undirected graph with the property that each pair of distinct vertices are connected to one another. Such a graph is usually denoted by K_n.

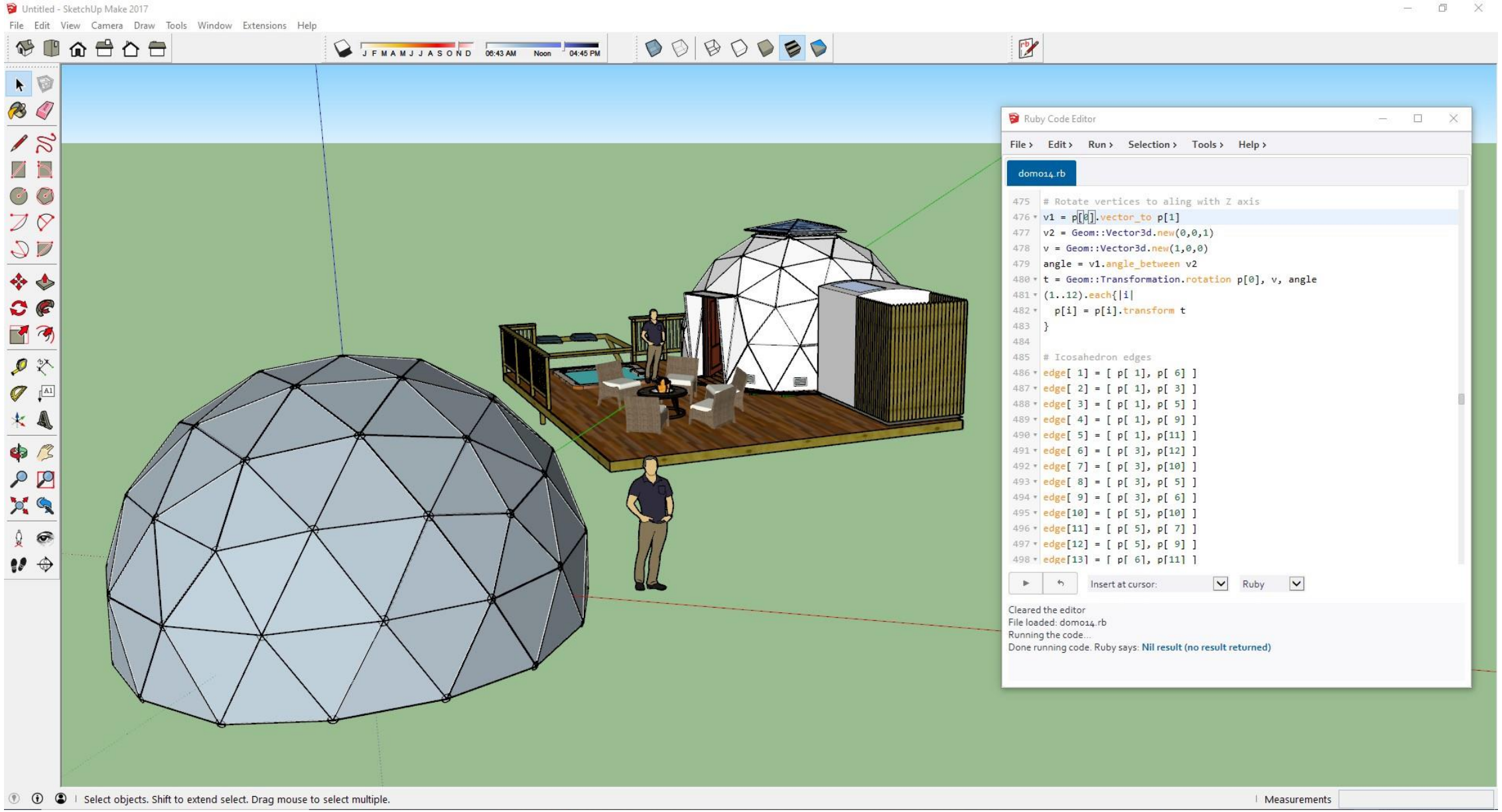
Multigraph

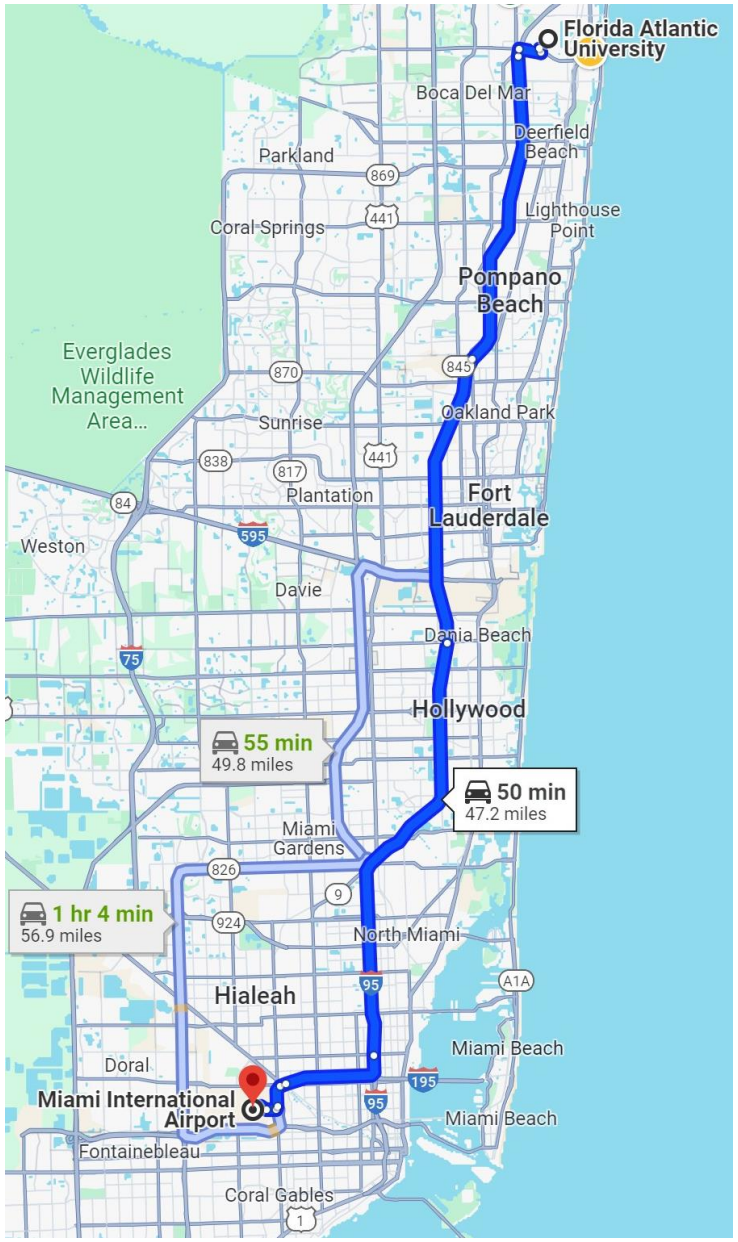
- A multigraph is a set of vertices V with a set of edges that can contain more than one edge between the vertices.

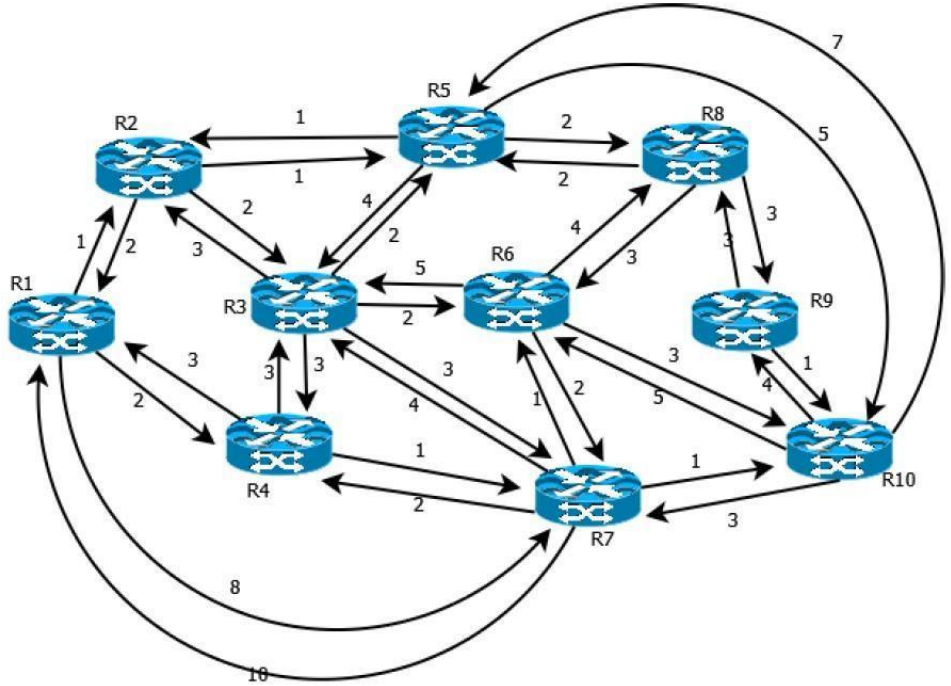
- Example: A road map: The cities and towns on the map can be thought of as vertices, while the roads are the edges. It is not uncommon to have more than one road connecting two cities.



Labels like e3 are used to avoid ambiguity, edge (2, 3) would be ambiguous.


Untitled - SketchUp Make 2017
File Edit View Camera Draw Tools Window Extensions Help
J F M A M J J A S O N D
06:43 AM
Noon
04:45 PM
Ruby Code Editor
File › Edit › Run › Selection › Tools › Help ›
domo14.rb
475 # Rotate vertices to aling with Z axis
476 v1 = p[0].vector_to p[1]
477 v2 = Geom::Vector3d.new(0,0,1)
478 v = Geom::Vector3d.new(1,0,0)
479 angle = v1.angle_between v2
480 t = Geom::Transformation.rotation p[0], v, angle
481 (1..12).each{|i|
482 p[i] = p[i].transform t
483 }
484
485 # Icosahedron edges
486 edge[1] = [p[1], p[6]]
487 edge[2] = [p[1], p[3]]
488 edge[3] = [p[1], p[5]]
489 edge[4] = [p[1], p[9]]
490 edge[5] = [p[1], p[11]]
491 edge[6] = [p[3], p[12]]
492 edge[7] = [p[3], p[10]]
493 edge[8] = [p[3], p[5]]
494 edge[9] = [p[3], p[6]]
495 edge[10] = [p[5], p[10]]
496 edge[11] = [p[5], p[7]]
497 edge[12] = [p[5], p[9]]
498 edge[13] = [p[6], p[11]]
Insert at cursor:
Ruby
Cleared the editor
File loaded: domo14.rb
Running the code...
Done running code. Ruby says: Nil result (no result returned)
Select objects. Shift to extend select. Drag mouse to select multiple.
Measurements





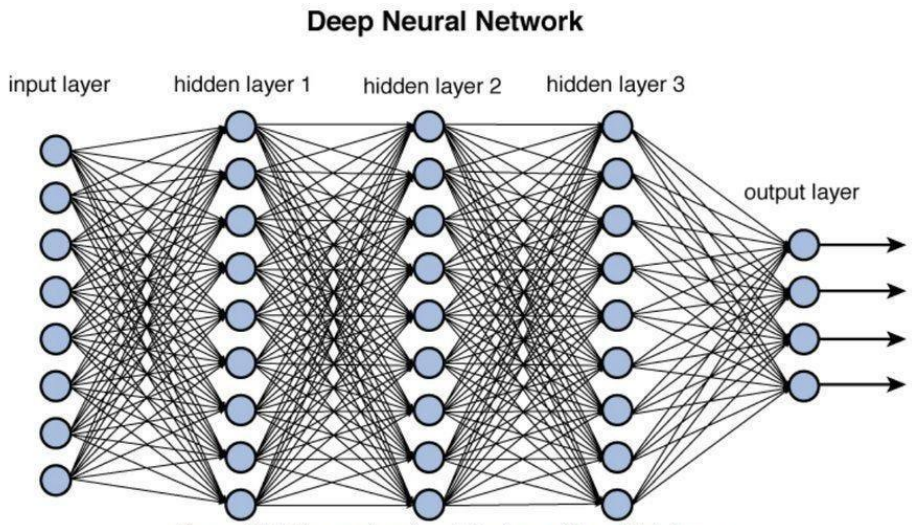
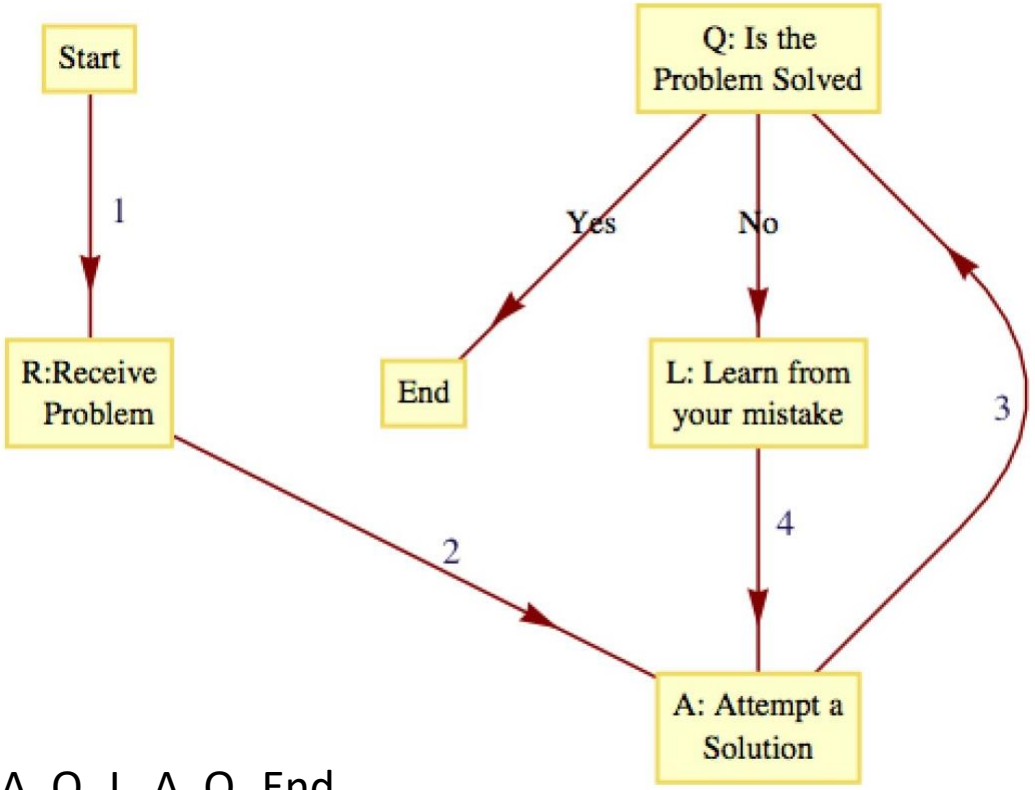


Figure 12.2 Deep network architecture with multiple layers.

Lecture 19 – part 3

More on Graph Theory

Example:
A Labeled Graph. A flowchart is a common example of a simple graph that requires labels for its vertices and some of its edges.



The sequence of vertices from "Start" to "End" is called a **path**.
The "Start" vertex is called the initial vertex of the path, while the "End" is called the final, or terminal, vertex.

Path:
Start, R, A, Q, L, A, Q, End.
1, 2, 3, No, 4, 3,Yes. (is unambiguous)

Notation and Terminology

- If x and y are two vertices of a graph, then a **path** between x and y describes a motion from x to y along edges of the graph.
- Vertex x is called the initial vertex of the path and y is called the terminal vertex.
- A path between x and y can always be described by its edge list, the list of edges that were used: $(e_1, e_2, \ldots, e_n)$, where:

  (1) the initial vertex of e_1 is x;

  (2) the terminal vertex of e_i is the initial vertex of e_{i+1}, $i = 1, 2, \ldots, n - 1$; and

  (3) the terminal vertex of e_n is y.
- The number of edges in the edge list is the **path length**.
- A path on a simple graph can also be described by a vertex list.
- A path of length n will have a list of $n+1$ vertices $v_0 = x, v_1, v_2, \ldots, v_n = y$, where, for $k = 0, 1, 2, \ldots, n-1$, (v_k, v_{k+1}) is an edge on the graph.
- A **circuit** is a path that terminates at its initial vertex.

Notation and Terminology

- A **circuit** is a path that terminates at its initial vertex.
- A **subpath** of this graph is any portion of the path described by one or more consecutive edges in the edge list.
- Any path is its own subpath; however, we call it an **improper subpath** of itself.
- All other nonempty subpaths are called **proper subpaths**.
- A path or circuit is simple if it contains no proper subpath that is a circuit.
- In other words, a path or circuit is simple if it does not visit any vertex more than once except for the common initial and terminal vertex in the circuit.

Subgraphs

Let G = (V, E) be a graph. This includes all types: directed, directed multigraph, or undirected.

Subgraph Conditions: A subgraph G' = (V', E') of G must satisfy:

- $V' \neq \emptyset$ (V' is not empty).
- V' is a subset of V ($V' \subseteq V$).
- An edge e is in E' if and only if e is in E and the endpoints of e are within V'.

Creating a Subgraph:

- Remove zero or more vertices from V, and all edges connected to those vertices.
- Optionally, remove additional edges not connected to the removed vertices.

Induced Subgraph:

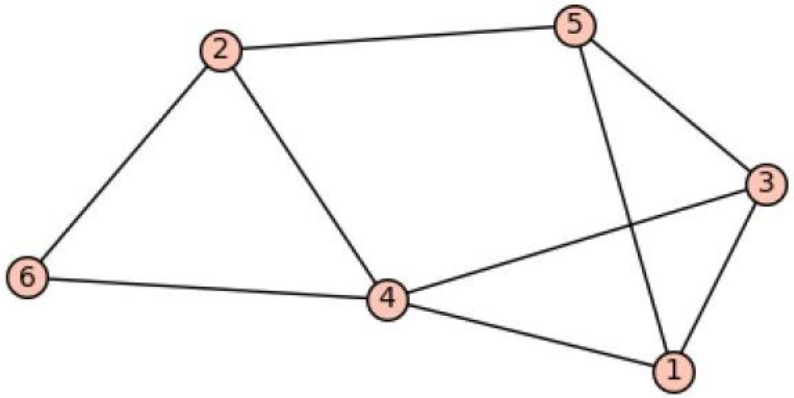
If G' is formed by removing only vertices and their connected edges (no other edges), then G' is an induced subgraph of G.

Spanning Subgraph:

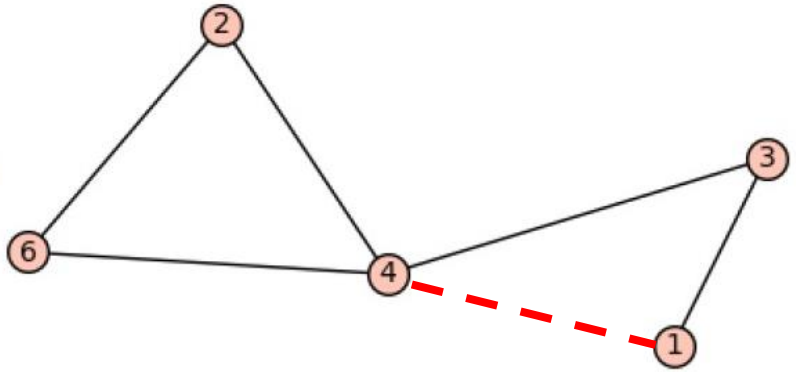
G' is a spanning subgraph of G if V' equals V (V' = V), meaning no vertices are removed, only edges can be removed.

Example:

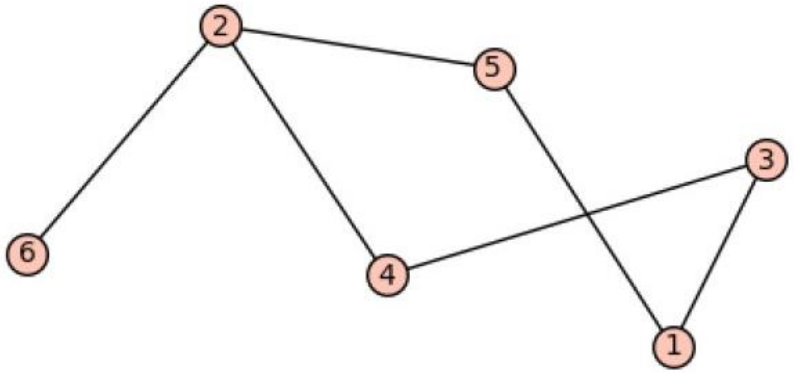
Graph



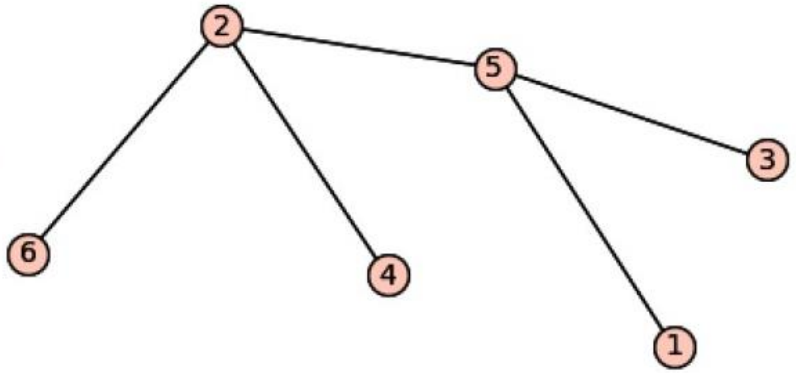
Subgraph



Induced subgraph



Spanning subgraph



Spanning subgraph
Spanning subtree

Connected Component

- You can think of a connected component as a little island of points that are all reachable from one another without having to use a bridge to another island.
- So if we had a large graph, the connected components would be the separate little islands of connectivity within it.

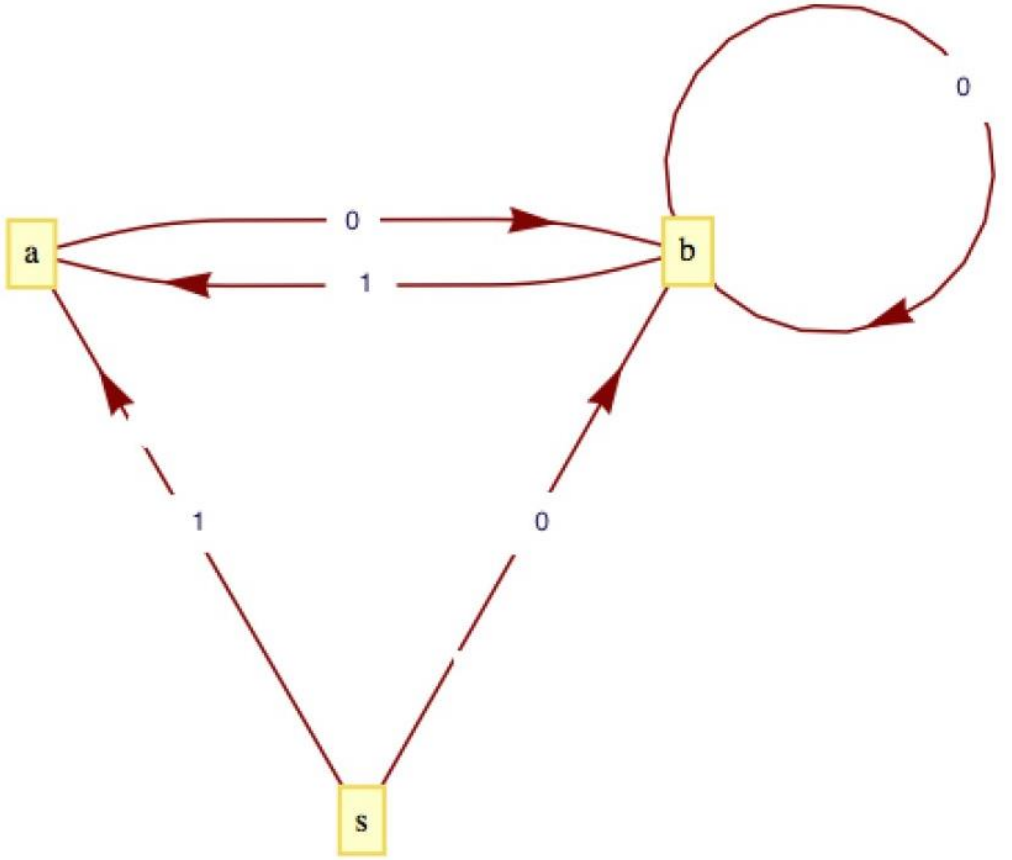
Example: A Graph as a Model for a Set of Strings.
Suppose that you would like to mechanically describe the set of strings of 0's and 1's having no consecutive 1's.

One way to visualize a string of this kind is with the graph in the Figure.

Consider any path starting at vertex s. If the label on each graph is considered to be the output to a printer, then the output will have no consecutive 1's.

For example, the path that is described by the vertex list (s, a, b, b, a, b, b, a, b) would result in an output of 10010010.

Conversely, any string with no consecutive 1's determines a path starting at s.

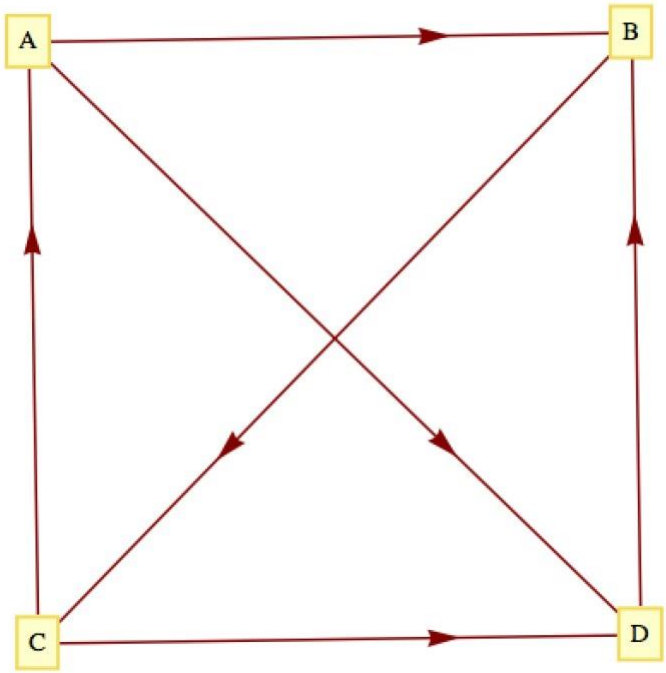


Example: A Tournament Graph.

Suppose that four teams compete in a round-robin sporting event; that is, each team meets every other team once, and each game is played until a winner is determined.

If the teams are named A, B, C, and D, we can define the relation β on the set of teams by $X\beta Y$ if X beat Y .

For one set of results, the graph of β might look like the figure.



There are many types of tournaments and they all can be modeled by different types of graphs.

Tournament Graph

(a) A tournament graph is a directed graph with the property that no edge connects a vertex to itself, and between any two vertices there is at most one edge.

(b) A complete (or round-robin) tournament graph is a tournament graph with the property that between any two distinct vertices there is exactly one edge.

(c) A single-elimination tournament graph is a tournament graph with the properties that:

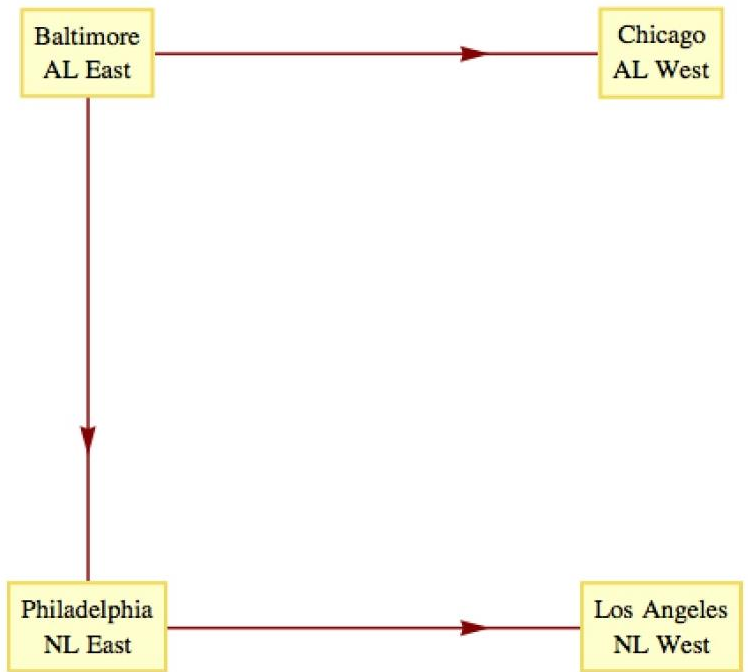
(i) one vertex (the champion) has no edge terminating at it and at least one edge initiating from it;

(ii) every other vertex is the terminal vertex of exactly one edge; and

(iii) there is a path from the champion vertex to every other vertex.

Example:

The major league baseball championship is decided with a single-elimination tournament, where each “game” is actually a series of games.
From 1969 to 1994, the two divisional champions in the American League (East and West) competed in a series of games.
The loser is eliminated, and the winner competed against the winner of the National League series (which is decided as in the American League).
The figure shows the tournament graph of the 1983 championship.



Graph Isomorphisms

- In simpler terms, if you can rename the vertices of one graph to get the other graph without changing which vertices are connected by edges, then the two graphs are isomorphic.