CIS2520

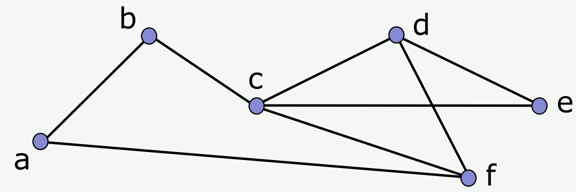
6. Graphs

UNDIRECTED GRAPHS 6.2

An ***undirected graph*** is a pair G=(V,E) where
✧ V is a finite nonempty set
✧ E is a set of unordered pairs of distinct elements of V

The elements of V are the ***vertices*** of the graph.
The elements of E are the (undirected) ***edges***.

V={a,b,c,d,e,f}, E={{a,b},{a,f},{b,c},{c,d},{c,e},{c,f},{d,e},{d,f}}

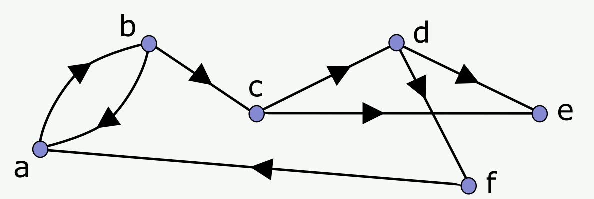


DIRECTED GRAPHS 6.3

A ***directed graph***, or ***digraph***, is a pair G=(V,E) where
- ✧ V is a finite nonempty set
- ✧ E is a set of ordered pairs of distinct elements of V

The elements of V are the ***vertices*** of the graph.
The elements of E are the (directed) ***edges***.

V={a,b,c,d,e,f}, E={(a,b),(b,a),(b,c),(c,d),(c,e),(d,e),(d,f),(f,a)}



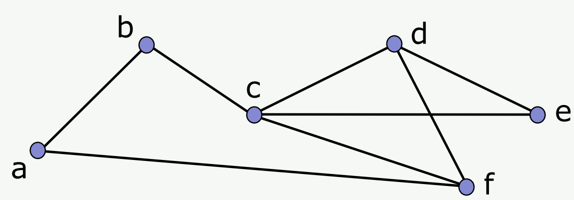
Reading suggestion: Chapter 10 of the textbook

BASIC TERMINOLOGY (1/2) 6.4

Consider a graph.

If {u,v} or (u,v) is an edge then:
- this edge is ***incident*** with u and v
- u and v are its ***endpoints***
- u and v are ***adjacent***

The ***degree*** of a vertex v is the number of vertices adjacent to v.



- ✧ a and b are adjacent
- ✧ f and e are not adjacent
- ✧ the degree of c is 4
- ✧ the degree of d is 3

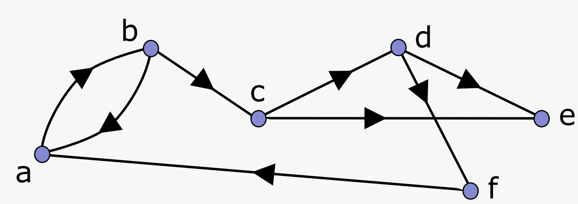
Reading suggestion: Chapter 10 of the textbook

BASIC TERMINOLOGY (2/2) 6.5

Consider a directed graph.

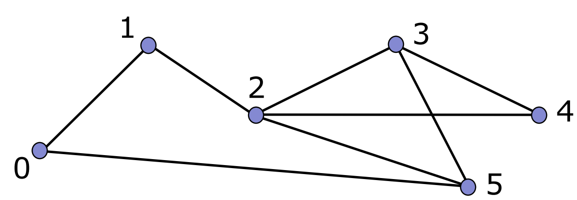
If (u,v) is an edge then:
- u is its ***origin***, v its ***destination***
- u is a ***predecessor*** of v, v a ***successor*** of u

The ***in-degree*** of a vertex v is the number of predecessors of v.
The ***out-degree*** of v is the number of successors of v.



- ✧ predecessors of a: b and f
- ✧ successors of e: none
- ✧ in-degree of b: 1
- ✧ out-degree of b: 2

IMPLEMENTATION (1/2) 6.6

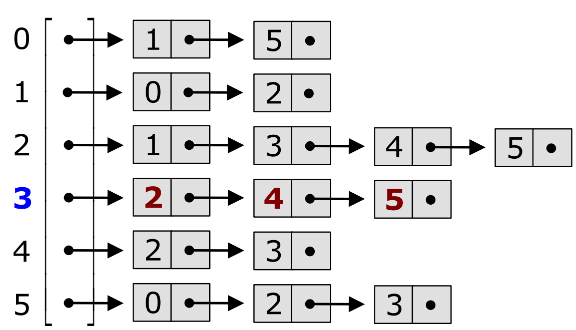


Adjacency matrix:

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

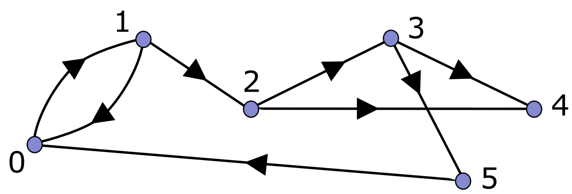
(e.g., **1** means that **3** and **4** are adjacent)

Adjacency lists:



(e.g., **2**, **4** and **5** are adjacent to **3**)

IMPLEMENTATION (2/2) 6.7

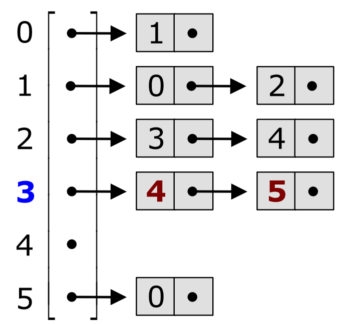


Adjacency matrix:

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

(e.g., **1** means that **4** is a successor of **3**)

Adjacency lists:



(e.g., **4** and **5** are the successors of **3**)

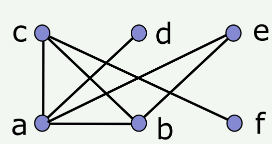
Reading suggestion: Chapter 10 of the textbook

PATHS (1/2) 6.8

A ***path*** is a tuple $(v_0,v_1,\ldots,v_n)$ of vertices, with n a positive integer, such that for any i in 1..n the pair $\{v_{i-1},v_i\}$, or (v_{i-1},v_i), is an edge.

The path $(v_0,v_1,\ldots,v_n)$ is a path of ***length*** n ***from*** its ***origin*** v_0 ***to*** its ***destination*** v_n.

It ***passes*** through the vertices v_0, v_1, ..., v_n and ***traverses*** the edges $\{v_{i-1},v_i\}$, or (v_{i-1},v_i).



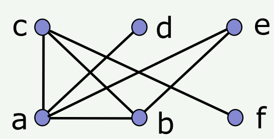
(b,c): path of length 1
(a,d,a): path of length 2
(a,b,e,a): path of length 3
(a,b,c,b,e): path of length 4
(c,a,e,b,a,d): path of length 5

Reading suggestion: Chapter 10 of the textbook

PATHS (2/2) 6.9

Consider a path $(v_0,v_1,\ldots,v_n)$.

- ✧ If all its edges and all its vertices are distinct, except maybe v_0 and v_n, then the path is ***simple***.
- ✧ If $v_0=v_n$ then the path is a ***cycle***.

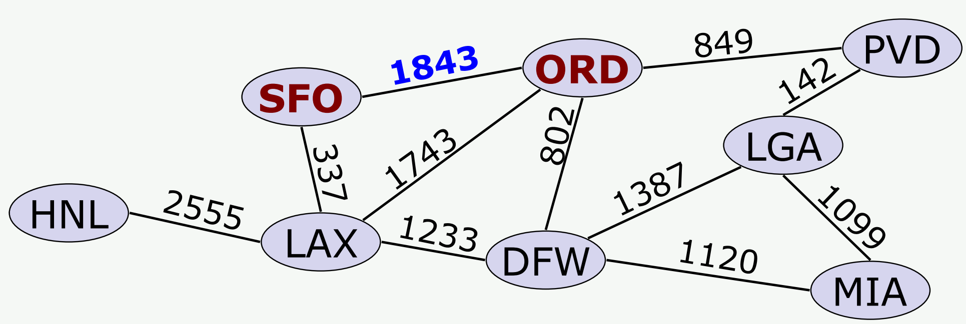


(b,c): simple path
(a,d,a): cycle
(a,b,e,a): simple cycle
(a,b,c,b,e): path
(c,a,e,b,a,d): path

WEIGHTED GRAPHS (1/2) 6.10

Let (V,E) be a graph,
$\mathbb{R}$ the set of real numbers,
w a total function from E to $\mathbb{R} \cup \{-\infty,+\infty\}$.

(V,E,w) is a ***weighted graph***.

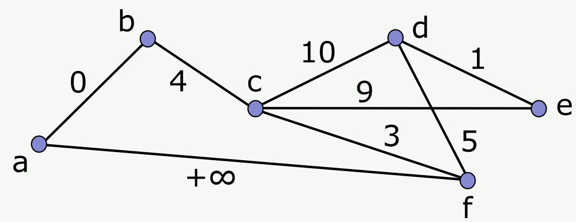


(e.g., **1843** miles between **San Francisco** and **Chicago** airports)

WEIGHTED GRAPHS (2/2) 6.11

Let (V,E,w) be a weighted graph.

- ✧ The ***weight*** of the edge e is w(e).
- ✧ The ***weight*** of the path $(v_0,v_1,\ldots,v_n)$ is the sum of the weights of its edges.
- ✧ A ***shortest path*** from u to v is a path from u to v with minimum weight.

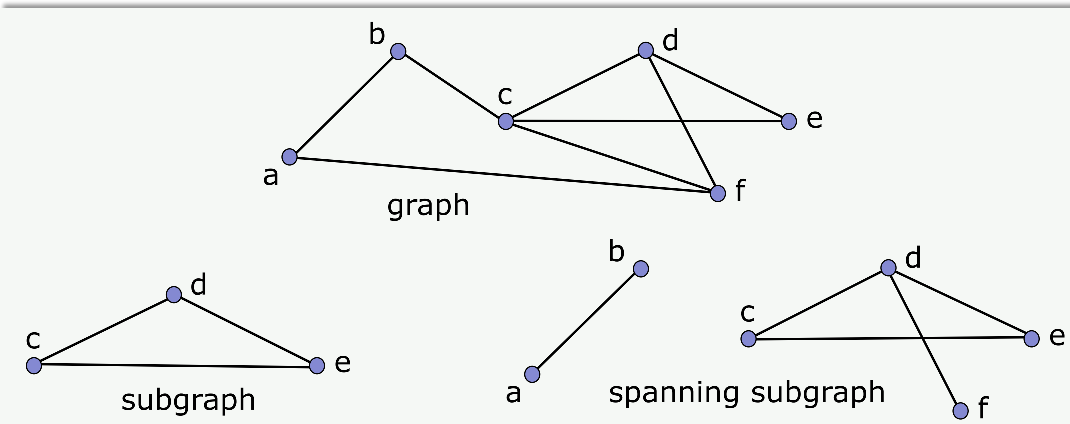


(e.g., (c,e) and (c,f,d,e) are shortest paths from c to e with weight 9)

OTHER TERMS (1/2) 6.12

Let (V,E) be a graph, V′ a nonempty subset of V, and E′ a subset of E.

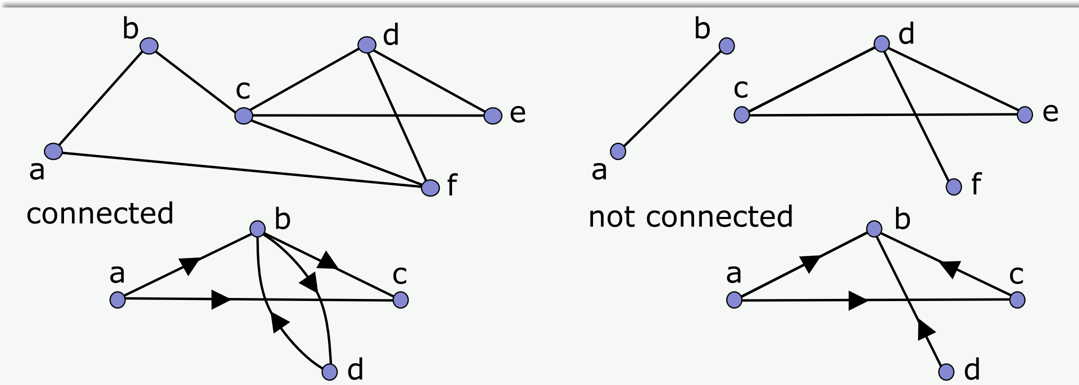
(V′,E′) is a ***subgraph*** of (V,E).
If V′=V then (V′,E′) is a ***spanning subgraph*** of (V,E).



OTHER TERMS (2/2) 6.13

Consider a graph G.

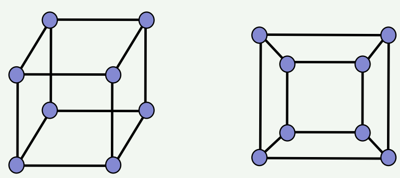
- ✧ Let u and v be two distinct vertices.
  u is ***connected*** to v if there exists a path from u to v.
- ✧ G is ***connected*** if, for any distinct vertices u and v,
  u is connected to v or v is connected to u.



APPLICATIONS: Example 1 6.14

Consider a digital circuit.
Is it possible to embed the circuit on a chip,
so that none of the wires cross?

The circuit is represented by a graph: each edge of the graph represents a wire. *Find out whether the graph is planar.*

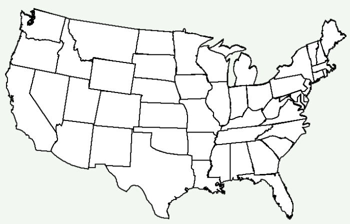


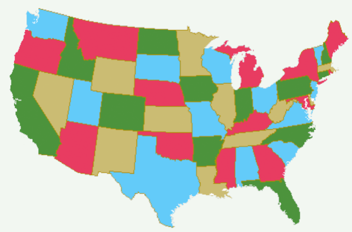
APPLICATIONS: Example 2 6.15

Two adjacent cell phone transmitters should be assigned different frequencies. What is the minimum number of frequencies that must be used?

In a map, two regions sharing a common border should be given different colors. What is the minimum number of colors that must be used?

The map is represented by a graph: each vertex represents a region; two vertices are adjacent if the corresponding regions share a common border. *Calculate the chromatic number of the graph.*





APPLICATIONS: Example 3 6.16

Draw a picture without retracing any line or picking the pencil up off the paper.

Find a path around a collection of bridges that crosses each bridge exactly once.

The problem is modeled by a *multigraph*: each vertex represents one of the regions the bridges give access to; two vertices are adjacent if the corresponding regions are linked by a bridge. *Find an Euler cycle.*