

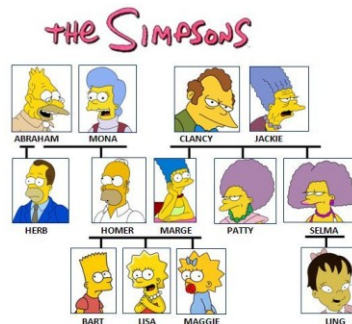
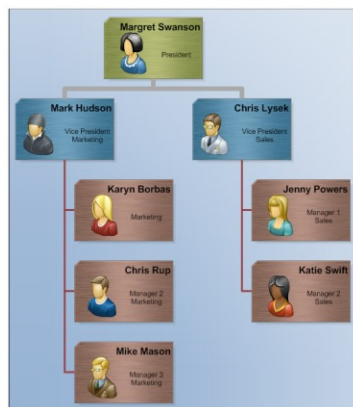
UNIT 6C

Organizing Data: Trees and Graphs

15110 Principles of Computing,
Carnegie Mellon University - CORTINA

1

Trees



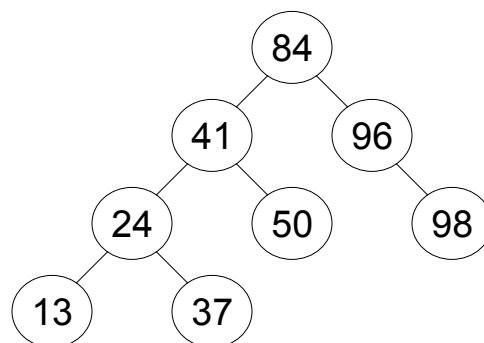
15110 Principles of Computing,
Carnegie Mellon University - CORTINA

2

Trees

- A **tree** is a hierarchical data structure.
 - Every tree has a **node** called the **root**.
 - Each node can have 1 or more nodes as **children**.
 - A node that has no children is called a **leaf**.
- A common tree in computing is a **binary tree**.
 - A binary tree consists of nodes that have at most 2 children.
 - A **complete binary tree** has the maximum number of nodes on each of its levels.
- Applications: data compression, file storage, game trees

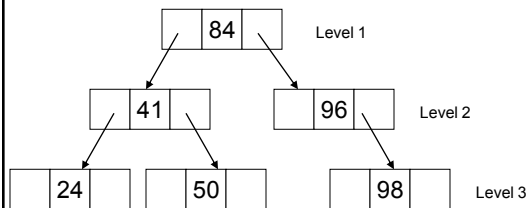
Binary Tree



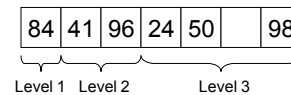
The root has the data value 84.
There are 4 leaves in this binary tree: 13, 37, 50, 98.
This binary tree is not complete.

Binary Trees: Implementation

- One common implementation of binary trees uses nodes like a linked list does.
 - Instead of having a “next” pointer, each node has a “left” pointer and a “right” pointer.
- We could also use an array.



Array representation:

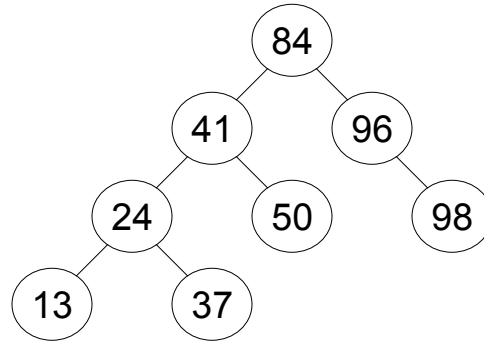


Binary Search Tree (BST)

- A binary search tree (BST) is a binary tree such that
 - All nodes to the left of any node have data values less than that node
 - All nodes to the right of any node have data values greater than that node

Using a BST

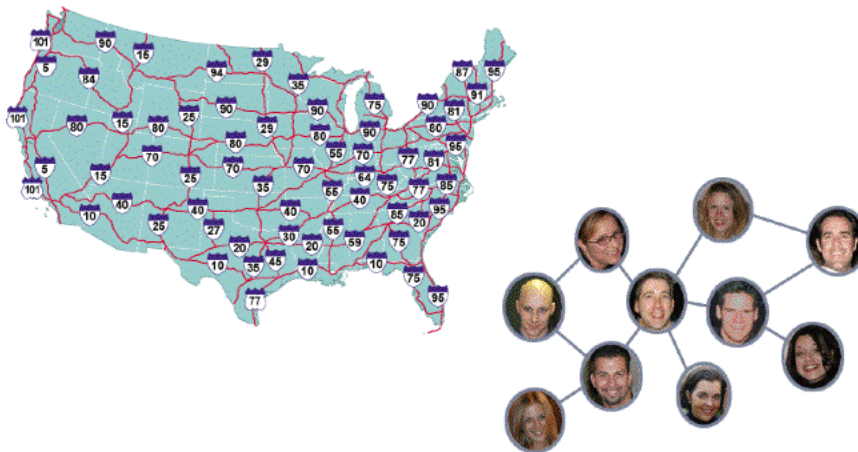
- How would you search for an element in a BST?



15110 Principles of Computing,
Carnegie Mellon University - CORTINA

7

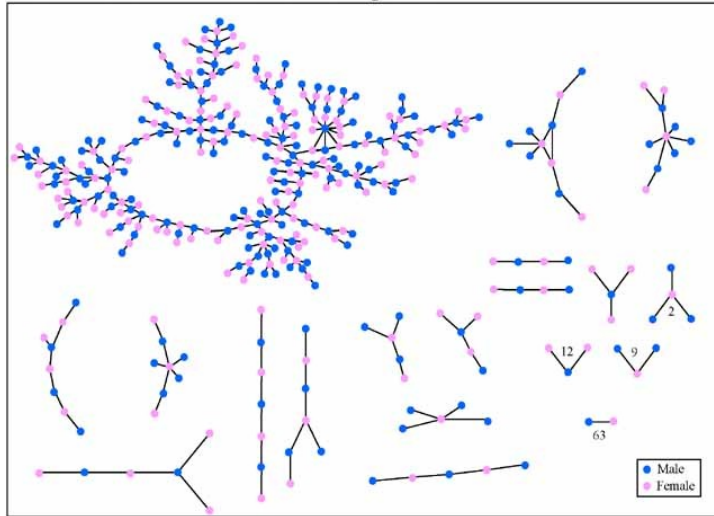
Graphs



15110 Principles of Computing,
Carnegie Mellon University - CORTINA

8

The Structure of Romantic Relations at "Jefferson High School"



Each circle represents a student and lines connecting students represent romantic relations occurring within the 6 months preceding the interview. Numbers under the figure count the number of times that pattern was observed (i.e. we found 63 pairs unconnected to anyone else).

15110 Principles of Computing,
Carnegie Mellon University - CORTINA

9

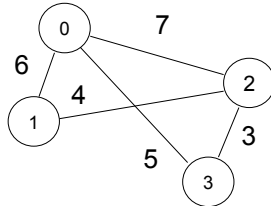
Graphs

- A **graph** is a data structure that consists of a set of vertices and a set of edges connecting pairs of the vertices.
 - A graph doesn't have a root, per se.
 - A vertex can be connected to any number of other vertices using edges.
 - An edge may be bidirectional or directed (one-way).
 - An edge may have a weight on it that indicates a cost for traveling over that edge in the graph.
- Applications: computer networks, transportation systems, social relationships

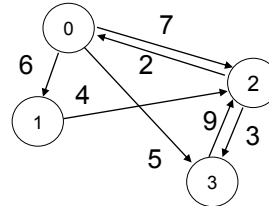
15110 Principles of Computing,
Carnegie Mellon University - CORTINA

10

Undirected and Directed Graphs



from \ to	0	1	2	3
0	0	6	7	5
1	6	0	4	∞
2	7	4	0	3
3	5	∞	3	0

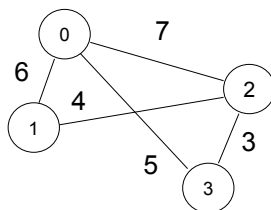


from \ to	0	1	2	3
0	0	6	7	5
1	∞	0	4	∞
2	2	∞	0	3
3	∞	∞	9	0

15110 Principles of Computing,
Carnegie Mellon University - CORTINA

11

Graphs in Ruby



from \ to	0	1	2	3
0	0	6	7	5
1	6	0	4	∞
2	7	4	0	3
3	5	∞	3	0

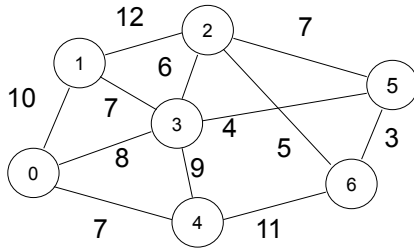
```
inf = 1.0/0.0
```

```
graph =
  [ [ 0, 6, 7, 5 ],
    [ 6, 0, 4, inf ],
    [ 7, 4, 0, 3 ],
    [ 5, inf, 3, 0 ] ]
```

15110 Principles of Computing,
Carnegie Mellon University - CORTINA

12

An Undirected Weighted Graph



0	1	2	3	4	5	6
Pitt.	Erie	Will.	S.C.	Harr.	Scr.	Phil.

vertices

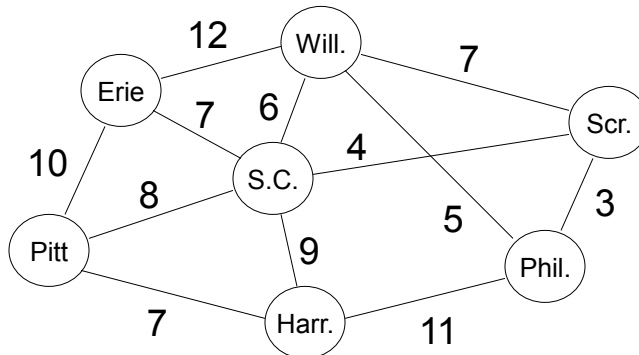
from \ to	0	1	2	3	4	5	6
	0	1	2	3	4	5	6
0	0	10	∞	8	7	∞	∞
1	10	0	12	7	∞	∞	∞
2	∞	12	0	6	∞	7	5
3	8	7	6	0	9	4	∞
4	7	∞	∞	9	0	∞	11
5	∞	∞	7	4	∞	0	3
6	∞	∞	5	∞	11	3	0

edges

15110 Principles of Computing,
Carnegie Mellon University - CORTINA

13

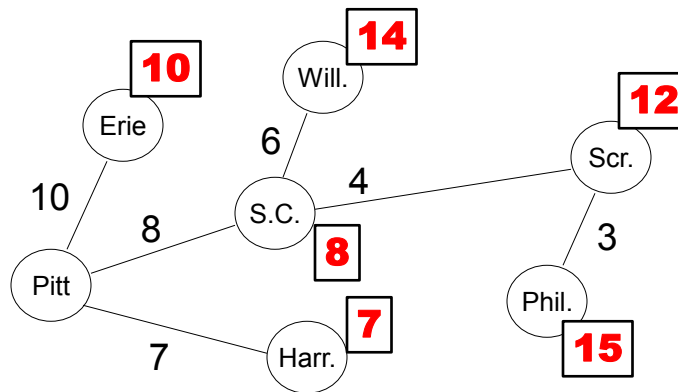
Original Graph



15110 Principles of Computing,
Carnegie Mellon University - CORTINA

14

Shortest Paths from Pittsburgh



15110 Principles of Computing,
Carnegie Mellon University - CORTINA

15

Graph Algorithms

- There are algorithms to compute the minimal spanning tree of a graph and the shortest paths for a graph.
 - We will see these later on in the semester.
- There are algorithms for other graph operations:
 - If a graph represents a set of pipes and the number represent the maximum flow through each pipe, then we can determine the maximum amount of water that can flow through the pipes assuming one vertex is a “source” (water coming into the system) and one vertex is a “sink” (water leaving the system)
 - Many more graph algorithms... very useful to solve real life problems.

15110 Principles of Computing,
Carnegie Mellon University - CORTINA

16