CHAPTER 6

Section 6.3 & 6.4

SECTION 6.3

Shortest Path and Minimal Spanning Tree

Shortest Path

- Given a simple, connected, weighted graph (positive weights)
  - How do we find a path between two given nodes x and y with minimum weight?
  - Know a path exists, because the graph is connected.
  - Example: Information in a communications network must be routed from one node to another in the most efficient way possible (weight could be a function of network traffic)
- Dijkstra's Algorithm
  - Apply to Figure 6.10 (page 504)

Minimal Spanning Tree

- Spanning tree for a connected graph
  - A nonrooted tree whose set of nodes coincides with the set of nodes for the graph and whose arcs are (some of) the arcs of the graph.
  - Connects all nodes of a graph with no excess arcs
- Minimal spanning tree for a simple, weighted, connected graph
  - A spanning tree with minimal weight
  - Example: Connect network nodes efficiently (least cost), with least amount of connecting medium (such as cable)
  - Prim's Algorithm

SECTION 6.4

Traversal Algorithms

Graph Traversal Algorithms

- Depth First Search
  - Travel deep into the graph first, then back up.
- Breadth First Search
  - Fan out from top nodes first, then move deeper