Review Guide: Chapter 10

Definitions: How are the following terms defined?

- graph, edge-endpoint function *(p. 626)*
- loop in a graph, parallel edges, adjacent edges, isolated vertex *(p. 626)*
- directed graph *(p. 629)*
- simple graph *(p. 632)*
- complete graph on n vertices *(p. 633)*
- complete bipartite graph on (m, n) vertices *(p. 633)*
- subgraph *(p. 634)*
- degree of a vertex in a graph, total degree of a graph *(p. 635)*
- walk, trail path, closed walk, circuit, simple circuit *(p. 644)*
- connected vertices, connected graph *(p. 646)*
- connected component of a graph *(p. 647)*
- Euler circuit in a graph *(p. 648)*
- Euler trail in a graph *(p. 652)*
- Hamiltonian circuit in a graph *(p. 654)*
- adjacency matrix of a directed (or undirected) graph *(p. 662)*
- symmetric matrix *(p. 664)*
- $n \times n$ identity matrix *(p. 669)*
- powers of a matrix *(p. 670)*
- isomorphic graphs *(p. 676)*
- isomorphic invariant for graphs *(p. 679)*
- circuit-free graph *(p. 683)*
- tree, forest, trivial tree *(p. 683)*
- parse tree, syntactic derivation tree *(p. 684)*
- terminal vertex (or leaf), internal vertex (or branch vertex) *(p. 688)*
- rooted tree, level of a vertex in a rooted tree, height of a rooted tree *(p. 694)*
- parents, children, siblings, descendants, and ancestors in a rooted tree *(p. 694)*
- binary tree, full binary tree, subtree *(p. 696)*
- spanning tree *(p. 702)*
- weighted graph, minimum spanning tree *(p. 704)*

Graphs

- How can you use a graph as a model to solve a problem? *(p. 631)*
- What does the handshake theorem say? In other words, how is the total degree of a graph related to the number of edges of the graph? *(p. 636)*
- How can you use the handshake theorem to determine whether graphs with specified properties exist? *(pp. 636, 638)*
- If an edge is removed from a circuit in a graph, does the graph remain connected? *(p. 647, 690)*
- A graph has an Euler circuit if, and only if, it satisfies what two conditions? *(p. 652)*
- A graph has a Hamiltonian circuit if, and only if, it satisfies what four conditions? *(p. 655)*
- What is the traveling salesman problem? *(p. 656)*
- How do you find the adjacency matrix of a directed (or undirected) graph? How do you find the graph that corresponds to a given adjacency matrix? *(p. 663)*

- How can you determine the connected components of a graph by examining the adjacency matrix of the graph? *(p. 666)*
- How do you multiply two matrices? *(p. 666)*
- How do you use matrix multiplication to compute the number of walks from one vertex to another in a graph? *(p. 672)*
- How do you show that two graphs are isomorphic? *(p. 677)*
- What are some invariants for graph isomorphisms? *(p. 679)*
- How do you establish that two simple graphs are isomorphic? *(p. 680)*

Trees

- How do you show that a saturated carbon molecule with k carbon atoms has $2k + 2$ hydrogen atoms? *(p. 686 and exercise 4 in Section 10.2)*
- If a tree has at least two vertices, how many vertices of degree 1 does it have? *(p. 687)*
- If a tree has n vertices, how many edges does it have? *(p. 688)*
- If a connected graph has n vertices, what additional property guarantees that it will be a tree? *(p. 692)*
- How can you represent an algebraic expression using a binary tree? *(p. 696)*
- Given a full binary tree, what is the relation among the number of its internal vertices, terminal vertices, and total number of vertices? *(p. 697)*
- Given a binary tree, what is the relation between the number of its terminal vertices and its height? *(p. 698)*
- What is the relation between the number of edges in two different spanning trees for a graph? *(p. 702)*
- How does Kruskal's algorithm work? *(p. 704)*
- How do you know that Kruskal's algorithm produces a minimum spanning tree? *(p. 706)*
- How does Prim's algorithm work? *(p. 707)*
- How do you know that Prim's algorithm produces a minimum spanning tree? *(p. 708)*
- How does Dijkstra's shortest path algorithm work? *(p. 711)*
- How do you know that Dijkstra's shortest path algorithm produces a shortest path? *(p. 713)*