MAD 3305 - GRAPH THEORY FLORIDA INT'L UNIV

REVISION FOR TEST #1 REMEMBER TO BRING AN 8x11 BLUE EXAM BOOKLET

KEY DEFINITIONS AND MAIN CONCEPTS

 Digraph, multi-digraph, pseudo-digraph, multi-pseudo-digraph (digraph-like object), in-degree & out-degree, Graph, multi-graph, pseudo-graph, multi-pseudo-graph (graph-like object), degree of a vertex, degree sequence, (G), Δ(G), sub-graphs, , regular graphs, adjacency matrix; geometric, set-theoretic, and matrix representation; walk, trail, circuit, cycle, path, distances in weighted graphs; inaccessible vertices, eccentricity, diameter, center, & radius of a graph; connected graphs, the connected components of a graph, weakly & strongly connected digraphs, bridge (cut-edge), edge connectivity, vertex connectivity, cut-vertex, pendant vertices, trees, forests, non-identical trees, leaves, minimum spanning trees, rooted trees, levels, height of a tree, children, parent, ancestors, descendants, n-ary trees, binary trees, codes, coding, uniquely decipherable coding, optimal coding, weighted path-length of a coding, [~~pre-order & post-order traversals, prefix & postfix notations~~], Networks, source & sink, capacity of an edge, source & sink, capacity constraint, conservation of flow, legal flow, value of a flow, source separating set of vertices, capacity of a source separating set of vertices, *MAXFLOW (N), MINCUT(N)*.

MAIN ALGORITHMS & PROBLEM SOLVING TECHNIQUES

1. (a) Graphical sequence algorithm

 (b) Graph recovery algorithm,

2. (a) Dijkstra’s distance algorithm,

 (b) Dijkstra’s shortest-path algorithm [modification of 2(a)]

3. (a) Kruskal's minimum-weight spanning tree algorithm,

 (b) Prim's minimum-weight spanning tree algorithm,

4. (a) Prufer's tree-encoding algorithm,

 (b) Prufer's tree-decoding algorithm,

5. Huffman's optimal-coding algorithm.

6. Finding the number of leaves or the minimum number of vertices in certain trees.

MAIN THEOREMS

1. The decreasing sequence <a,d2,d3, ... ,dp > is graphical if and only if

 <d2-1,d3-1,...,da+1-1,da+2,...,dp> is graphical . (*Graphical Sequence Theorem*)

2. The number of walks of length n from vi to vj = (An) [i, j].

3. (a) A connected graph with p vertices has at least p-1 edges.

 (b) A graph with p vertices and more than (p-1)(p-2)/2 edges is always connected.

4. (a) If G is a disconnected graph, then Gc must be connected.

 (b) If G has p vertices and (G) > (p-1)/2, then G is connected.

5. (a) Any tree with p vertices has exactly p-1 edges.

 (b) G is a tree if and only if there is exactly one path between any two vertices.

6. In any n-ary tree T with p vertices we have logn [{p.(n-1) +1}/n] < h(T) < p-1.

7. There are pp-2 different (non-identical) trees on p distinct vertices. (*Cayley's theorem*)