Algorithms & Complexity 4/24/17

0145-344-001

Note Taker: Jai Punjwani

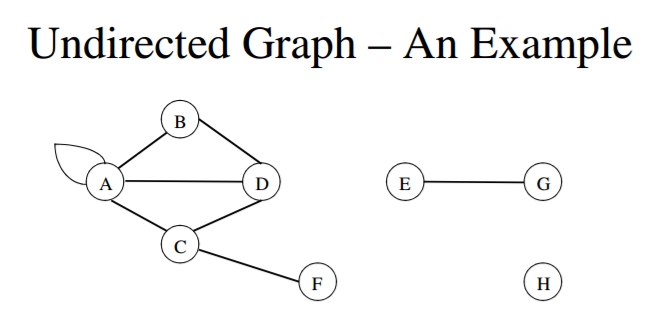
ANNOUNCEMENTS

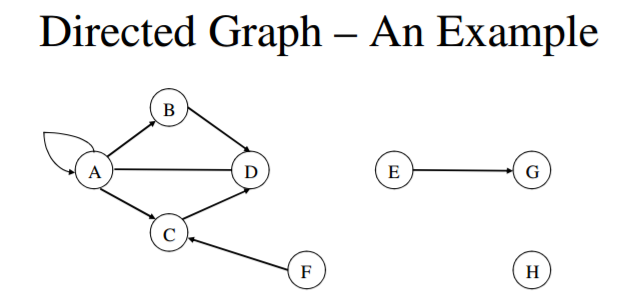
Topic: Graphs

PowerPoint: <http://home.adelphi.edu/~siegfried/cs344/344l12.pdf>

Graph – consists of a set of nodes and a set of arcs (edges). A tree and a linkedList are examples of graphs.

Digraph/Directed Graph – graph in which pair of nodes that make up arcs are **ordered** pairs.





Notice that the directed graph has arrows, or an **order**.

A common example of a digraph is a tree. Note that all trees are graphs, but not all graphs are trees.

Incident – a node *n* is incident to an arc *x* if *n* is one of the two nodes in the ordered pair of nodes constituting *x*.

Degree of a node – number of arcs incident to it

Weighted Graphs – a graph in which a number value is associated with each arc of a graph. Examples include the distance of an arc, or the cost of a trip from destination A to destination B.

Paths/Cycles

Path of length *k* from node *a* to node *b* consists of a sequence of *k+1* nodes.

Cycle – a path from one node to itself

Adjacency Matrix – describes the graph in terms of where a node can travel to in *k* hops.

Adj1 is the matrix of paths from a node with one arc in between

Adj2 is the matrix of paths from a node with two arcs in between

Adji is the matrix of paths from a node with i arcs in between

If you want to find a path from A to B, we simply look for it in each adjacency matrix until we find it.

path = Adj1 | Adj2 | Adj3 | … | Adji

Note that matrix multiplication is order ***O***(n3), and in the graph implementation in the code in the PowerPoint, and is performed (n-1) times, meaning it has order ***O***(n4), meaning it is VERY inefficient for large matrices and/or many matrices.

**Dijkstra’s Algorithm** – finds shortest path in a weighted graph

Algorithm can be found here: <http://www.blackwasp.co.uk/Dijkstra.aspx>