Session 2: Trees in Graphs

Shayan Shahrabi

October 2024

A Quick Review

(In the following text, the symbol T denotes an arbitrary tree.)
Definition of a tree: A connected graph without cycles!

Theorem 1 *In every graph resembling T, any two nodes, such as x and y, are connected by **a unique** path.*
Solution: *Prove by contradiction!*

Theorem 2 *If T has p nodes and q edges, it follows that $q = p - 1$.*
Solution: *Use induction on p.*

Theorem 3 *If T is a binary tree with height h and p nodes, then the following holds:*

$$h + 1 \leq p \leq 2^{(h+1)} - 1.$$

Solution: *Refer to page 103 of your textbook!*

Theorem 4 *An m-tree (a tree in which each node has either n children or is a leaf) with l leaves has a height h that satisfies the following inequality:*

$$h \geq \log_m(l).$$

Algorithm 1 ***Breadth First Search (BFS)***

Algorithm 2 ***Depth First Search (DFS)***

Algorithm 3 ***Prim's Algorithm***
For an explanation, view the following video: Prim's Algorithm in 2 Minutes.

Algorithm 4 ***Kruskal's Algorithm***
For an explanation, view the following video: Kruskal's Algorithm in 2 Minutes.

Problems

Problem 1: Bye Bye Node!

Suppose we remove a vertex v and all edges connected to it from the tree T. Prove that if $\deg(v) = w$, then T is divided into w connected components.

Problem 2: Rather Simple

Prove that every connected graph contains a minimum spanning tree!

Problem 3: A Classic Problem

Let G be a graph with $p-1$ edges. Prove that the following statements are equivalent (TFAE):

- G is connected.
- G has no cycles.
- G is a tree.

Problem 4

Prove that a simple connected graph with exactly two intersecting edges forms a path.

Problem 5: One from Bondy!

Prove that every tree with more than one node has at least two leaves.

Problem 6: Prove or Disprove

Prove or disprove the following statement: If $d_1, d_2, \ldots, d_p$ are the degrees of vertices in a graph G, then the set of degrees $\{1, d_1+1, d_2, \ldots, d_p\}$ is a valid set of degrees for a tree.

Problem 7: Tree of Cycles?!

Suppose T has n vertices. Prove that T is similar to a subgraph of $\bar{C}_{n+2}$.

Problem 8: Draw and Explore

Draw a tree of your own choice and execute the BFS and DFS algorithms on it!

Problem 9: Let's Do Some Coding!

You can take a look at this GitHub repository for further insights on the topic.