CSCI-C 241 HW #11
Instructors: Wennstrom & LaRacuente Due Tuesday, April 16th, 2024

Note that this assignment **is not eligible for resubmission**. You may turn it in up to a week late (for the usual late penalty), but make sure you've done it correctly the first time.

For the first two questions on this assignment, we are using the graph theory definition of a tree:

Definition. A **tree** is a connected graph with no cycles.

1. Read the following proof of the claim that for any two vertices a and b in a tree, there is one and only one path between them.

   Proof. Let T be a tree and choose two vertices a and b. Since all trees are connected, there must exist at least one path from a to b. (If $a = b$, then the only such path is the empty path because any other path would create a cycle, and trees don't have cycles. So for the rest of this proof, assume $a = b$.)

   Suppose towards a contradiction that there is a different path from a to b. Again, all the vertices are distinct from each other.

   We don't necessarily know that the vertices in the first path are *all* different from the vertices in the second path, but at least *some* of them must be different. If they're not all different, shrink down the two paths to find a pair of subpaths that start and end with the same vertices but don't have any other vertices in common. (It's possible these paths might just have one vertex that is different, but that's okay.) Name the vertices in these two paths $(v_0, v_1, v_2, \ldots, v_n)$ and $(w_0, w_1, w_2, \ldots, w_m)$, where $a = v_0 = w_0$ and $b = v_n = w_m$, and where all of the other vertices are distinct.

   The trail $(a, v_1, v_2, \ldots, b, w_{m-1}, w_{m-2}, \ldots, w_2, w_1, a)$ is a cycle because $v_0 = w_0$ are the only vertices that are the same and there's at least one edge in this trail because $v_0 \neq v_n$. But T is a tree, so there can't be any cycles in it.

   Therefore there cannot exist more than one path from a to b. □

   Give a similar proof of the converse claim:

   Claim. For any graph G, if every pair of vertices in G has one and only one path between them, then G is a tree.

2. Give a proof of the following claim, using induction on the number of vertices in the graph:

   Claim. For any tree T, if you remove one edge from the tree, the resulting graph will not be connected.

3. For the remaining questions, we will be using the recursive definition of a tree:

   Definition. A **full binary tree** (FBT) is one of:

   - [] (a single node), or

- $[T_1, T_2]$, where T_1 and T_2 are FBT's (two trees joined together with one new root and two new edges from the new root to the old roots of T_1 and T_2).

We can define a recursive function $\mathsf{nd} : \text{FBT} \to \mathbb{N}$ that counts the number of nodes in a FBT as follows:

$$\mathsf{nd}(T) = \begin{cases} 1 & \text{if } T = [\,] \\ \mathsf{nd}(T_1) + \mathsf{nd}(T_2) + 1 & \text{if } T = [T_1, T_2] \text{ for FBT's } T_1 \text{ and } T_2 \end{cases}$$

(a) Write a recursive definition of the function $\mathsf{ed} : \text{FBT} \to \mathbb{N}$ that calculates the number of edges in a tree.

(b) Using your definition of ed, prove the following claim using structural induction:

Claim. For every full binary tree t, $\mathsf{nd}(t) = \mathsf{ed}(t) + 1$.

(c) If you have a full binary tree with n leaves, what's the smallest possible depth that tree can have?

Remember that the **depth** of a tree is the length of the longest path (number of edges) from the root to a leaf.

(d) If you have a full binary tree with depth d, what's the largest possible number of leaves that tree can have?

(e) If you have a tree where each node has at most c children with n leaves, what's the largest possible number of leaves that tree can have?

(f) If you have a tree where each node has at most c children with depth d, what's the largest possible number of leaves that tree can have?

(g) Prove that your answer to the previous question is true.