1. (a) Write a procedure that takes a pointer to the root of an ordered tree represented as a binary tree, and returns the number of leaves in the tree.

(b) Write a procedure that takes a pointer to the root of an ordered tree represented as a binary tree, and returns the height of the tree.

(c) Write a procedure that takes a pointer to the root of an ordered tree represented as a binary tree, and returns the average depth of the leaves in the tree. Recall that the depth of a node is the length of the path from the root to the node. Your algorithm should compute the quantity
\[
\sum_{v \text{ is a leaf}} \text{depth of } v / \text{number of leaves}.
\]

2. (a) How many directed graphs are there on a given set of \(n\) vertices?

(b) How many undirected graphs are there on a given set of \(n\) vertices?

3. A path in a graph is in a graph is called simple if all vertices in the path are distinct. Prove that if a graph has a path between two vertices, then it has a simple path between the same two vertices.

4. When an adjacency-matrix representation is used, most graph algorithms require time \(\Omega(n^2)\), where \(n\) is the number of vertices. However, there are some exceptions. A universal sink in a directed graph is a vertex with in-degree \(n-1\) and out-degree 0. Give an \(O(n)\) algorithm that determines if a given graph, represented as an adjacency matrix, has a universal sink.