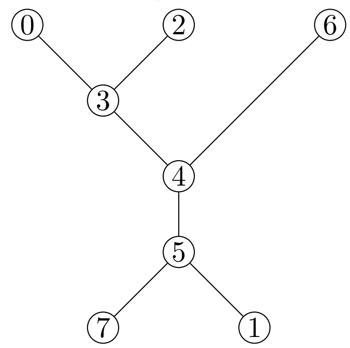
MATH 113: DISCRETE STRUCTURES
FRIDAY WEEK 8 HANDOUT

A *leaf* of a tree is a vertex of degree 1. Suppose T is a tree with vertex set $\{0, 1, 2, \ldots, n-1\}$. The *Prüfer code* of T is the sequence of length $n-2$ with entries in $\{0, 1, \ldots, n-1\}$ generated by the following algorithm: At step i, remove the leaf with the smallest label not equal to 0 and set the i-th entry of the Prüfer code equal to the label of the leaf's neighbor. After step $n-2$, the end of the algorithm, one is left with a single edge joining some node to 0.

For instance, the Prüfer code of the following graph is 534543.



In your reading, you learned how to turn a Prüfer code into a tree by writing down its extended Prüfer code, a $2 \times n$ array with entries in $\{0, 1, \ldots, n-1\}$ with columns corresponding to edges. To quote,

> Each entry in the first row of the extended Prüfer code is the smallest integer that does not occur in the first row before it, nor in the second row below or after it.

One applies this procedure with initial data the second row consisting of the Prüfer code with a 0 tacked on the end.

Problem 1. Draw a tree on vertex set $\{0, 1, \ldots, n-1\}$ with $n = 6$, 7, 8, or 9. Determine its Prüfer code and write the Prüfer code on the whiteboard. Then trade Prüfer codes with another group and decode into a tree. Draw the tree next to its Prüfer code and check your work with the group that made the Prüfer code.

Problem 2. Which trees have Prüfer codes that contain only one value?

Problem 3. Which trees have Prüfer codes with distinct values in all positions?