Problem 1.

Definition 1.1. The recursive data type, binary-2PTG, of *binary trees* with leaf labels, L is defined recursively as follows:

- **Base case:** $\langle \texttt{leaf}, l\rangle \in$ binary-2PTG, for all $l \in L$.
- **Constructor case:** If $G_1, G_2 \in$ binary-2PTG, then

$$\langle \texttt{bintree}, G_1, G_2\rangle \in \text{binary-2PTG}.$$

The *size*, $|G|$, of $G \in$ binary-2PTG is defined recursively on this definition by:

- **Base case:**

$$|\langle \texttt{leaf}, l\rangle| ::= ::= l, \quad \text{for all} l \in L.$$

- **Constructor case:**

$$|\langle \texttt{bintree}, G_1, G_2\rangle| ::= |G_1| + |G_2| + 1.$$

For example, for the size of the binary-2PTG, G, pictured in Figure 1, is 7.

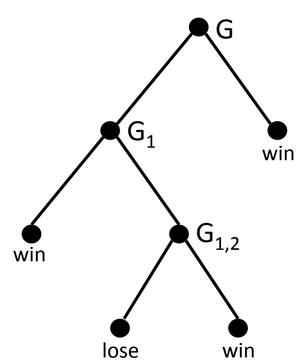

Figure 1: *A picture of a binary tree* w.

(a) Write out (using angle brackets and labels `bintree`, `leaf`, etc.) the binary-2PTG, G, pictured in Figure 1.

The value of flatten(G) for $G \in$ binary-2PTG is the sequence of labels in L of the leaves of G. For example, for the binary-2PTG, G, pictured in Figure 1,

$$\text{flatten}(G) = (\texttt{win}, \texttt{lose}, \texttt{win}, \texttt{win}).$$

(b) Give a recursive definition of flatten. (You may use the operation of *concatenation* (append) of two sequences.)

(c) Prove by structural induction on the definitions of flatten and size that

$$2 \cdot \text{length}(\text{flatten}(G)) = |G| + 1. \tag{1}$$