1. The famous Basque computational arborist Gorka Oihanean has a favorite 26-node binary tree, in which each node is labeled with a letter of the alphabet. Intorder and postorder traversals of his tree visits the nodes in the following orders:

   Inorder: `F E V I B H N X G W A Z O D J S R M U T C K Q P L Y`
   Postorder: `F V B I E N A Z W G X J S D M U R O H K C Q Y L P T`

   (a) List the nodes in Professor Oihanean's tree according to a preorder traversal.

   (b) Draw Professor Oihanean's tree.

   [Hint: It may be easier to write a short Python program than to figure this out by hand.]

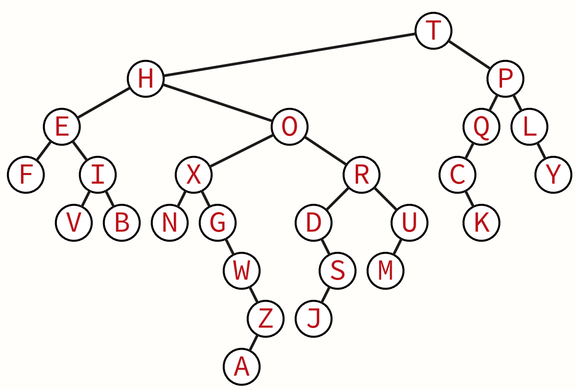
Solution:

(a) `T H E F I V B O X N G W Z A R D S J U M P Q C K L Y`

> **Rubric:** 5 points. −1 for each misplaced, missing, or repeated letter, but no negative scores. No proof is required. The solution is unique.

"The five boxing wizards jump quickly." is one of my favorite examples of a ***pangram****: a sentence that contains each letter of the alphabet at least once. Removing all duplicate letters from this pangram yielded a more readable string than either* `PACKMYBOXWITHFVEDZNLQURJGS` *or* `THEQUICKBROWNFXJMPSVLAZYDG`.

(b)



> **Rubric:** 5 points. −1 for each misplaced, missing, or repeated node, but no negative scores. No credit if the submission is not a binary tree. In particular, the distinction between left and right children matters, even for nodes with only one child. No proof is required. The solution is unique.

■

2. For any string $w \in \{0, 1\}^*$, let $swap(w)$ denote the string obtained from w by swapping the first and second symbols, the third and fourth symbols, and so on. The *swap* function can be formally defined as follows:

$$swap(w) := \begin{cases} \varepsilon & \text{if } w = \varepsilon \\ w & \text{if } w = 0 \text{ or } w = 1 \\ ba \bullet swap(x) & \text{if } w = abx \text{ for some } a, b \in \{0,1\} \text{ and } x \in \{0,1\}^* \end{cases}$$

(a) Prove by induction that $|swap(w)| = |w|$ for every string w.

Solution: Let w be an arbitrary string.

Assume $|swap(x)| = |x|$ for every string x that is shorter than w.

There are three cases to consider (mirroring the definition of *swap*):

- If $w = \varepsilon$, then

$$\begin{aligned} |swap(w)| &= |\varepsilon| && \text{by definition of } swap \\ &= |w| && \text{because } w = \varepsilon \end{aligned}$$

- If $w = 0$ or $w = 1$, then

$$|swap(w)| = |w| \qquad \text{by definition of } swap$$

- Finally, if $w = abx$ for some $a, b \in \{0, 1\}$ and $x \in \{0, 1\}^*$, then

$$\begin{aligned} |swap(w)| &= |ba \bullet swap(x)| && \text{by definition of } swap \\ &= |ba| + |swap(x)| && \text{because } |y \bullet z| = |y| + |z| \\ &= |ba| + |x| && \text{by the induction hypothesis} \\ &= 2 + |x| && \text{by definition of } |\cdot| \\ &= |ab| + |x| && \text{by definition of } |\cdot| \\ &= |ab \bullet x| && \text{because } |y \bullet z| = |y| + |z| \\ &= |abx| && \text{by definition of } \bullet \\ &= |w| && \text{because } w = abx \end{aligned}$$

In all cases, we conclude that $|swap(w)| = |w|$. ■

Rubric: 5 points: Standard induction rubric (scaled). This is more detail than necessary for full credit.

(b) Prove by induction that $swap(swap(w)) = w$ for every string w.

Solution: Let w be an arbitrary string.
Assume $swap(swap(x)) = x$ for every string x that is shorter than w.
There are three cases to consider (mirroring the definition of *swap*):

- If $w = \varepsilon$, then

$$\begin{aligned} swap(swap(w)) &= swap(\varepsilon) && \text{by definition of } swap \\ &= \varepsilon && \text{by definition of } swap \\ &= w && \text{because } w = \varepsilon \end{aligned}$$

- If $w = \mathtt{0}$ or $w = \mathtt{1}$, then

$$\begin{aligned} swap(swap(w)) &= swap(w) && \text{by definition of } swap \\ &= w && \text{by definition of } swap \end{aligned}$$

- Finally, if $w = abx$ for some $a, b \in \{\mathtt{0}, \mathtt{1}\}$ and $x \in \{\mathtt{0}, \mathtt{1}\}^*$, then

$$\begin{aligned} swap(swap(w)) &= swap(ba \bullet swap(x)) && \text{by definition of } swap \\ &= swap(ba \bullet z) && \text{where } z = swap(x) \\ &= swap(baz) && \text{by definition of } \bullet \\ &= ab \bullet swap(z) && \text{by definition of } swap \\ &= ab \bullet swap(swap(x)) && \text{because } z = swap(x) \\ &= ab \bullet x && \text{by the induction hypothesis} \\ &= abx && \text{by definition of } \bullet \\ &= w && \text{because } w = abx \end{aligned}$$

In all cases, we conclude that $swap(swap(w)) = w$. ■

Rubric: 5 points: Standard induction rubric (scaled). This is more detail than necessary for full credit.

3. Consider the set of strings $L \subseteq \{\texttt{0}, \texttt{1}\}^*$ defined recursively as follows:

   - The empty string ε is in L.
   - For any string x in L, the string $\texttt{0}x$ is also in L.
   - For any strings x and y in L, the string $\texttt{1}x\texttt{1}y$ is also in L.
   - These are the only strings in L.

   There was a discrepancy in the string to consider in part (a) between the homework PDF (1011101011101011) and Gradescope (1011101011110101). We will accept proofs that either of these strings is in L for full credit. Sorry about that.

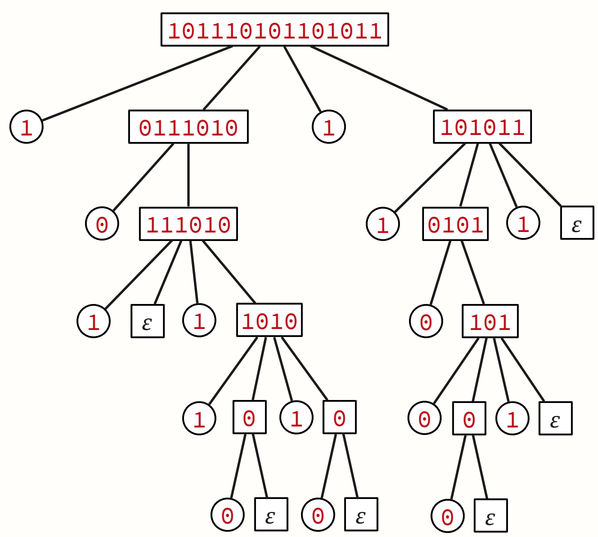
   (a) **[homework PDF]** Prove that the string 1011101011101011 is in L.

Solution: We derive a sequence of strings in L as follows:

$$
\begin{aligned}
o &= \texttt{0}\varepsilon = \texttt{0} \\
t &= \texttt{1}\varepsilon\texttt{1}\varepsilon = \texttt{11} \\
u &= \texttt{0}t = \texttt{011} \\
w &= \texttt{1}o\texttt{1}o = \texttt{1010} \\
x &= \texttt{1}o\texttt{1}u = \texttt{1 0 1 011} \\
y &= \texttt{1}w\texttt{1}x = \texttt{1 1010 1 101011} \\
z &= \texttt{1}o\texttt{1}w = \texttt{1 0 1 110101101011}
\end{aligned}
$$

■

Solution: The following parse tree provides a proof. Every string in a rectangle is in L, either because that string is empty, or because it can be broken into smaller strings (the children) as described in the definition of L. Circles represent individual symbols.



■

Solution (clever): The string $w = $ 1011101011110011 contains exactly ten 1s. Because 10 is even, part (c) implies that $w \in L$. ■

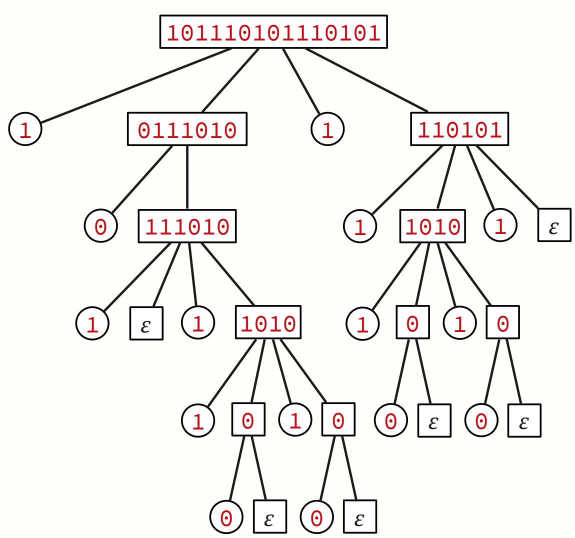
Rubric: 2 points. This is not the only derivation of this string, and these are not the only valid proof structures. The first proof is more detailed than necessary for full credit, but any similar proof must separately justify each of the component substrings (110101101011, 101011, 1010, 011, 11, and 0 in the first proof). The clever proof is worth full credit even without a solution to part (c).

(a) **[Gradescope]** Prove that the string 1011101011110101 is in L.

Solution: We derive a sequence of strings in L as follows:

$$u = 0\varepsilon = 0$$
$$v = 1u1u = 1010$$
$$w = 1v1\varepsilon = 110101$$
$$x = 1\varepsilon 1v = 111010$$
$$y = 0x = 0\,111010$$
$$z = 1y1w = 1\,0111010\,1\,110101$$
■

Solution: The following parse tree provides a proof. Every string in a rectangle is in L, either because that string is empty, or because it can be broken into smaller strings (the children) as described in the definition of L. Circles represent individual symbols.



■

Solution (clever): The string $w = $ 1011101011110101 contains exactly ten 1s. Because 10 is even, part (c) implies that $w \in L$. ■

Rubric: 2 points. This is not the only derivation of this string, and these are not the only valid proof structures. The first proof is more detailed than necessary for full credit, but any similar proof must separately justify each of the component substrings (`0111010`, `111010`, `110101`, `1010`, and `0` in the first proof). The clever proof is worth full credit even without a solution to part (c).

(b) Prove that every string $w \in L$ contains an even number of `1`s.

Solution: Let w be an arbitrary string in L.

Assume that every string $x \in L$ that is shorter than w has an even number of `1`s.

Recall that $\#(1, w)$ denotes the number of `1`s in w.

There are three cases to consider (reflecting the definition of L).

- If $w = \varepsilon$, then w contains zero `1`s, and zero is even.
- If $w = 0x$ for some string $x \in L$, then

$$\begin{aligned} \#(1, w) &= \#(1, x) && \text{by definition of } \# \\ &\text{is even} && \text{by the induction hypothesis, because } x \in L \end{aligned}$$

- Finally, if $w = 1x1y$, for some strings $x, y \in L$, then

$$\begin{aligned} &\#(1, w) \\ &\quad = 1 + \#(1, x1y) && \text{by definition of } \# \\ &\quad = 1 + \#(1, x) + \#(1, 1y) && \text{because } \#(1, uv) = \#(1, u) + \#(1, v) \\ &\quad = 2 + \#(1, x) + \#(1, y) && \text{by definition of } \# \end{aligned}$$

  Because both $x \in L$ and $y \in L$, the inductive hypothesis implies that both $\#(1, x)$ and $\#(1, y)$ are even. It follows that $\#(1, w)$ is the sum of three even numbers and so must be even.

In all cases, we conclude that w contains an even number of `1`s. ■

Rubric: 4 points: standard induction rubric (scaled)

(c) Prove that every string $w \in \{0, 1\}^*$ with an even number of `1`s is a member of L.

Solution: Let w be an arbitrary string in $\{0, 1\}^*$ with an even number of `1`s.

Assume that every string $x \in \{0, 1\}^*$ that is shorter than w and has an even number of `1`s is a member of L.

Recall that $\#(1, w)$ denotes the number of `1`s in w.

There are three cases to consider:

- If $w = \varepsilon$, then $w \in L$ by definition of L.
- Suppose $w = 0x$ for some string $x \in \{0, 1\}^*$
  The definition of $\#$ implies that $\#(1, w) = \#(1, x)$.
  It follows that $\#(1, x)$ is even.
  Because x is shorter than w, the inductive hypothesis implies $x \in L$.
- Finally, suppose $w = 1x$ for some string $x \in \{0, 1\}^*$.

The definition of # implies that $\#(1, w) = \#(1, x) + 1$.
It follows that $\#(1, x)$ is odd.

Thus, x must contain at least one 1.
By breaking x at the ***first*** 1, we can write $x = y1z$, where y contains no 1s.
(This is the only non-mechanical part of the proof.)

We now have $\#(1, w) = 2 + \#(1, y) + \#(1, z)$, as in the last case of part (b).
By definition $\#(1, y) = 0$, which is even.
Because $\#(1, w)$ is even, $\#(1, z)$ must also be even.
The induction hypothesis now implies that $y \in L$ and $z \in L$.
Thus, the definition of L implies that $w = 1y1z \in L$.

In all cases, we conclude that $w \in L$. ■

Rubric: 4 points = standard induction rubric (scaled)