Strings

- An **alphabet** $\Sigma$ is any finite set of characters

- The set $\Sigma^*$ of **strings** over the alphabet $\Sigma$ is defined by
  - **Basis:** $\varepsilon \in \Sigma^*$ (\(\varepsilon\) is the empty string)
  - **Recursive:** if $w \in \Sigma^*$, $a \in \Sigma$, then $wa \in \Sigma^*$

Palindromes

Palindromes are strings that are the same backwards and forwards (e.g. “abba”, “tth”, “neveroddoreven”).

**Basis**
- $\varepsilon$ is a palindrome
- $a$ is a palindrome for every $a \in \Sigma$

**Recursive Step**
- If $p$ is a palindrome and $a \in \Sigma$, then $apa$ is a palindrome.
All Binary Strings with no 1’s before 0’s...

Write a recursive definition for the set of binary strings in which all 0’s appear before any 1’s in the entire string.

---

Function Definitions on Recursively Defined Sets

Length:
\[ \text{len}(\varepsilon) = 0 \]
\[ \text{len}(wa) = 1 + \text{len}(w) \text{ for } w \in \Sigma^*, a \in \Sigma \]

Reversal:
\[ \varepsilon^R = \varepsilon \]
\[ (wa)^R = aw^R \text{ for } w \in \Sigma^*, a \in \Sigma \]

Concatenation:
\[ x \cdot \varepsilon = x \text{ for } x \in \Sigma^* \]
\[ x \cdot wa = (x \cdot w)a \text{ for } x \in \Sigma^*, a \in \Sigma \]

Number of c’s in a string:
\[ \#_c(\varepsilon) = 0 \]
\[ \#_c(wc) = \#_c(w) + 1 \text{ for } w \in \Sigma^* \]
\[ \#_c(wa) = \#_c(w) \text{ for } w \in \Sigma^*, a \in \Sigma, a \neq c \]

---

Rooted Binary Trees

• **Basis:** • is a rooted binary tree

• **Recursive step:**

If \( T_1 \) and \( T_2 \) are rooted binary trees,

then so is: \( T_1 \) \( T_2 \)

---

Functions Defined on Rooted Binary Trees

• **size(•) = 1**

• **size( ) = 1 + size(T_1) + size(T_2)**

• **height(•) = 0**

• **height( ) = 1 + \max\{\text{height}(T_1), \text{height}(T_2)\}**
Structural Induction

How to prove $\forall x \in S, \, P(x)$ is true:

**Base Case:** Show that $P(u)$ is true for all specific elements $u$ of $S$ mentioned in the **Basis step**

**Inductive Hypothesis:** Assume that $P$ is true for some arbitrary values of each of the existing named elements mentioned in the **Recursive step**

**Inductive Step:** Prove that $P(w)$ holds for each of the new elements $w$ constructed in the **Recursive step** using the named elements mentioned in the **Inductive Hypothesis**

**Conclude** that $\forall x \in S, \, P(x)$

---

Structural Induction vs. Ordinary Induction

**Ordinary induction is a special case of structural induction:**

**Recursive definition of $\mathbb{N}$**

**Basis:** $0 \in \mathbb{N}$

**Recursive Step:** If $k \in \mathbb{N}$ then $k+1 \in \mathbb{N}$

**Structural induction follows from ordinary induction:**

Let $Q(n)$ be true iff for all $x \in S$ that take $n$ recursive steps to be constructed, $P(x)$ is true.

---

Using Structural Induction

- Let $S$ be given by...
  - **Basis:** $6 \in S; \ 15 \in S$;
  - **Recursive:** if $x, \, y \in S$ then $x + y \in S$.

**Claim:** Every element of $S$ is divisible by $3$.

---

**Claim:** Every element of $S$ is divisible by $3$. 
Structural Induction for Strings

Let \( S \) be a set of strings over \( \{a,b\} \) defined as follows...

**Basis:** \( a \in S \)

**Recursive:**
- If \( w \in S \) then \( aw \in S \) and \( bw \in S \)
- If \( u \in S \) and \( v \in S \) then \( uv \in S \)

**Claim:** If \( w \in S \), then \( w \) has more \( a \)'s than \( b \)'s.

----

Function Definitions on Recursively Defined Sets

**Length:**
- \( \text{len}(\varepsilon) = 0 \)
- \( \text{len}(wa) = 1 + \text{len}(w) \) for \( w \in \Sigma^* \), \( a \in \Sigma \)

**Reversal:**
- \( \varepsilon^R = \varepsilon \)
- \( (wa)^R = aw^R \) for \( w \in \Sigma^* \), \( a \in \Sigma \)

**Concatenation:**
- \( x \cdot \varepsilon = x \) for \( x \in \Sigma^* \)
- \( x \cdot wa = (x \cdot w)a \) for \( x \in \Sigma^* \), \( a \in \Sigma \)

**Number of c's in a string:**
- \( \#_c(\varepsilon) = 0 \)
- \( \#_c(wa) = \#_c(w) + 1 \) for \( w \in \Sigma^* \)
- \( \#_c(wa) = \#_c(w) \) for \( w \in \Sigma^* \), \( a \in \Sigma \), \( a \neq c \)

Claim: If \( w \in S \), then \( \#_a(w) > \#_b(w) \)

**Basis:** \( a \in S \)

**Recursive:**
- If \( w \in S \) then \( aw \in S \) and \( bw \in S \)
- If \( u \in S \) and \( v \in S \) then \( uv \in S \)

Claim: \( \text{len}(x \cdot y) = \text{len}(x) + \text{len}(y) \) for all \( x,y \in \Sigma^* \)

Let \( P(y) \) be "\( \text{len}(x \cdot y) = \text{len}(x) + \text{len}(y) \) for all \( x \in \Sigma^* \)"
Functions Defined on Rooted Binary Trees

- \( \text{size}(\bullet) = 1 \)
- \( \text{size}(\text{T}_1, \text{T}_2) = 1 + \text{size}(\text{T}_1) + \text{size}(\text{T}_2) \)
- \( \text{height}(\bullet) = 0 \)
- \( \text{height}(\text{T}_1, \text{T}_2) = 1 + \max\{\text{height}(\text{T}_1), \text{height}(\text{T}_2)\} \)

Claim: For every rooted binary tree \( T \), \( \text{size}(T) \leq 2^{\text{height}(T)} + 1 - 1 \)