Four weeks left: What happens now?

The class speeds up a bit.
Homework problems get more conceptual.

We will cover:

- Recursively defined sets and functions
- Structural induction
- Regular expressions and context free grammars
- Relations and graphs
- Finite state machines and automata
- Turing machines and undecidability

---

**Recursive definition of sets**

- **Basis step**: Some specific elements are in $S$
- **Recursive step**: Given some existing named elements in $S$ some new objects constructed from these named elements are also in $S$.
- **Exclusion rule**: Every element in $S$ follows from basis steps and a finite number of recursive steps

**Recursive definition**

- Basis step: $0 \in S$
- Recursive step: if $x \in S$, then $x + 2 \in S$
- Exclusion rule: Every element in $S$ follows from basis steps and a finite number of recursive steps

**Powers of 3:**

Basis: $6 \in S; 15 \in S$
Recursive: if $x, y \in S$, then $x + y \in S$

Basis: $[1, 1, 0] \in S; [0, 1, 1] \in S$
Recursive:
if $[x, y, z] \in S$, $a \in \mathbb{R}$, then $[ax, ay, az] \in S$
if $[x_1, y_1, z_1], [x_2, y_2, z_2] \in S$
then $[x_1 + x_2, y_1 + y_2, z_1 + z_2] \in S$

---

**Strings**

- An alphabet $\Sigma$ is any finite set of characters.
  e.g. $\Sigma = \{0, 1\}$ or $\Sigma = \{A, B, C, \ldots, X, Y, Z\}$ or $\Sigma$

- 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.

**Basis:**

ε is a palindrome and any α ∈ Σ is a palindrome

**Recursive step:**

If p is a palindrome then ap α is a palindrome for every α ∈ Σ.

**binary strings such that...**

First digit cannot be a 1.

* No occurrence of the substring 11.

**function definitions on recursively defined sets**

**Length:**

len(ε) = 0; len(wα) = 1 + len(w); for w ∈ Σ*, α ∈ Σ

**Reversal:**

εR = ε, (wa)R = awR for w ∈ Σ*, a ∈ Σ

**Concatenation:**

x ε = x for x ∈ Σ*

x wα = (x w)α for x, w ∈ Σ*, α ∈ Σ

**Number of vowels in a string:**

Σ = {a, b, c, ..., z}

V = {a, e, i, o, u}

**rooted binary trees**

**Basis:**

• is a rooted binary tree

**Recursive step:**

If T1 and T2 are rooted binary trees, then so is:

 Gibraltar

**rooted binary trees**

**Basis:**

• is a rooted binary tree

**Recursive step:**

If T1 and T2 are rooted binary trees, then so is:
defining a function on rooted binary trees

- size(•) = 1
- size(T_1 + T_2) = 1 + size(T_1) + size(T_2)
- height(•) = 0
- height(T_1 + T_2) = 1 + max(height(T_1), height(T_2))

structural induction

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

**Base Case:** Show that \( P(\alpha) \) is true for all specific elements \( \alpha \) 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 \( Q(x) \) 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.