Structural Induction
Examples

1. Propositions (Later)
   a. Base Case: T, F, p, q, r, ...
   b. Recursive Step: \(~p, p \land q, p \lor q, p \rightarrow q\)

2. Polynomials
   a. Base Case: 1, x
   b. Recursive Step: \(p + q, p \ast q, c \cdot p\)

3. Binary Trees
   a. Base Case: Empty Tree, Tree with one node
   b. Recursive Step: Node with left and right subtrees

4. Strings (of Balanced Parentheses)
   a. Base Case: Empty string, ()
   b. Recursive Step: (S), S_1S_2
**Principle of Structural Induction**

Let $R$ be a recursive definition. 
Let $S$ be a statement about the elements defined by $R$. 

If the following hypotheses hold:

i. $S$ is True for every element $b_1,\ldots,b_m$ in the base case of the definition $R$. 

ii. For every element $E$ constructed by the recursive definition from some elements $e_1,\ldots,e_n$:

   $S$ is True for $e_1,\ldots,e_n \Rightarrow S$ is true for $E$

Then we can conclude that:

iii. $S$ is True for Every Element $E$ defined by the recursive definition $R$. 

Template for Proofs by Structural Induction

Prove

i. $S$ is True for $b_1,\ldots,b_n$  \hspace{2cm} \{Base Case\}

ii. $S$ is True for $e_1,\ldots,e_n \Rightarrow S$ is True for $E$  \hspace{2cm} \{Inductive Step\}

Conclude

iii. $S$ is True for Every Element defined by $R$  \hspace{2cm} \{Conclusion\}
Observations on Structural Induction

Proofs by Structural Induction

- Extends inductive proofs to discrete data structures -- lists, trees, …
- For every recursive definition there is a corresponding structural induction rule.
- The base case and the recursive step mirror the recursive definition.
  -- Prove Base Case
  -- Prove Recursive Step

Proof of Structural Induction

Let $T = \{ E \mid S \text{ is True for } E \}$.

- $T$ contains the base cases
- $T$ contains all structures that can be built from the base cases

Hence $T$ must contain the entire recursively defined set.
Binary Trees

1. Recursive Definition
   a. Base Case: Empty Tree \( \phi \)
   b. Recursive Step: Node with left and right subtrees

2. Structural Induction
   a. If \( P(\phi) \) and \( \forall T_1, T_2 \{ P(T_1) \land P(T_2) \Rightarrow P(T) \text{ with nodes } T_1, T_2 \} \)
   b. Then \( \forall T P(T) \)

3. Size of a Tree
   a. Base Case: \( s(\phi) = 0 \)
   b. Recursive Step: \( s(T) = 1 + s(T_1) + s(T_2) \)

4. Height of a Tree
   a. Base Case: \( h(\phi) = 0 \)
   b. Recursive Step: \( h(T) = 1 + \max(h(T_1), h(T_2)) \)
Theorem: \( s(T) \leq 2^{h(T)+1} - 1 \)

Proof: By Structural Induction.

Base Case: \( s(\phi) = 0 \) and \( h(\phi) = 0 \)
\[
s(\phi) = 0 < 1 = 2 - 1 = 2^{h(\phi)+1} - 1
\]

Recursive Step: Let \( T \) be the tree with nodes \( T_1, T_2 \)

Assume: \( s(T_1) \leq 2^{h(T_1)+1} - 1 \) and \( s(T_2) \leq 2^{h(T_2)+1} - 1 \)

Must Show: \( s(T) \leq 2^{h(T)+1} - 1 \)

Structural Induction: By definition

\[
h(T) = 1 + \max (h(T_1), h(T_2)) = \max (1 + h(T_1), 1 + h(T_2))
\]

\[
s(T) = 1 + s(T_1) + s(T_2)
\]
Induction continued

\[ s(T) = 1 + s(T_1) + s(T_2) \]

\[ \leq 1 + \left( 2^{h(T_1)+1} - 1 \right) + \left( 2^{h(T_2)+1} - 1 \right) \quad \text{\{Inductive Hypothesis\}} \]

\[ \leq 1 + 2 \left( 2^{\max(h(T_1)+1, h(T_2)+1)} - 1 \right) \]

\[ \leq 2 \left( 2^{h(T)} \right) - 1 \]

\[ = 2^{h(T)+1} - 1 \]
Fractals

Theorem

Every angle in a Sierpinski Triangle is 60 degrees.

Proof

Base Case: Easy.

Inductive Step: By Structural Induction.
Balanced Parentheses

1. Definition
   a. Base Case: $\lambda$ (empty string)
   b. Recursive Step: $(S), S_1S_2$

2. Structural Induction
   a. $P(\lambda)$ and $\forall S_1, S_2 \{ P[S_1] \text{ and } P[S_2] \} \rightarrow P[(S)] \text{ and } P[S_1S_2]$
      then $\forall SP[S]$

3. Count Function
   a. $c[S] = \#\text{open parentheses} - \#\text{closed parentheses}$
      i. $c(\lambda) = 0$
      ii. $c[(S)] = c[S]$
      iii. $c[S_1S_2] = c[S_1] + c[S_2]$
Theorem: \( c[S] = 0 \)

Proof: By Structural Induction.

Base Case: \( c[\lambda] = 0 \)

Recursive Step;

\( c[(S)] = c[S] = 0 \)

\( c[S_1S_2] = c[S_1] + c[S_2] = 0 + 0 = 0 \)
More Strings

Recursive Definition

• Base Cases: $b$
• Recursive Step: $aSa$

Explicit Formula

• $a^n b a^n$  $n \geq 0$

Structural Induction

• If $P(b)$ and $(\forall S \ P(S) \rightarrow P(aSa))$, then $\forall S \ P(S)$
**Theorem:** Recursive Definition $\Leftrightarrow$ Explicit Definition

**Proof:** Recursive $\Rightarrow$ Explicit.

Every element constructed recursively is of the form $a^n b a^n$

By Structural Induction.

Base Case: $b = a^0 b a^0$.

Structural Induction:
- Suppose $S = a^n b a^n$
- Then $aSa = a(a^n b a^n) a = a^{n+1} b a^{n+1}$

Explicit $\Rightarrow$ Recursive.

Every element of the form $a^n b a^n$ can be constructed recursively.

By Weak Induction on $n$.

Base Case: $n = 0 \Rightarrow a^0 b a^0 = b$  Okay.
**Induction**

Assume: Every element of the form $a^n b a^n$ can be constructed recursively.

Must Show: Every element of the form $a^{n+1} b a^{n+1}$ can be constructed recursively.

Observe: $a^{n+1} b a^{n+1} = a^n b a^n a = a S a$.

By the inductive hypothesis: $a^n b a^n$ satisfies the recursive definition;

Hence by the recursive step, so does $a^{n+1} b a^{n+1}$. 
Polynomials

Recursive Definition

• Base Cases: $1, x$
• Recursive Step: $p + q, p \times q, c \, p$

Explicit Definition

• $p(x) = c_0 + c_1 x + \cdots + c_n x^n$

Structural Induction

• If $S(1), S(x)$ and $(\forall p, q \ S(p) \land S(q) \rightarrow S(p + q), S(p \times q), S(c \, p))$,
  then $\forall p \ S(p)$
Theorem: Recursive Definition ⇔ Explicit Definition

Proof: Recursive ⇒ Explicit.

Every element constructed recursively is of the form

\[ p(x) = c_0 + c_1 x + \cdots + c_n x^n. \]

By Structural Induction.

Base Case: 1, x. Okay.

Structural Induction:
- Suppose
  \[ p(x) = c_0 + c_1 x + \cdots + c_n x^n \]
  \[ q(x) = d_0 + d_1 x + \cdots + d_m x^m \]
- Then \( p + q, p \ast q, c \ p \) are also of the form
  \[ r(x) = e_0 + e_1 x + \cdots + e_k x^k \]
Explicit $\Rightarrow$ Recursive.

Every polynomial

$$p(x) = c_0 + c_1 x + \cdots + c_n x^n$$

can be constructed recursively.

By Weak Induction on $n$.

Base Case: $\text{degree}(p) = 0 \Rightarrow p = c1$. Okay.

Recursive Step: Suppose every polynomial

$$q(x) = c_0 + c_1 x + \cdots + c_n x^n$$

of degree $n$ can be constructed recursively.

Must Show: Every polynomial

$$p(x) = c_0 + c_1 x + \cdots + c_n x^n + c_{n+1} x^{n+1} = q(x) + c_{n+1} x^{n+1}$$

of degree $n + 1$ can be constructed recursively.

By the inductive hypothesis: $q(x)$ and $x^n$ are can both be constructed recursively

Hence by the recursive definition so can

$$c_{n+1} x^{n+1} = c_{n+1} (x x^n) \text{ and } p(x) = q(x) + c_{n+1} x^{n+1}.$$
Structural Induction on the Natural Numbers

Recursive Definition

- Base Case: 0 is in \( N \)
- Recursive Step: if \( n \) is in \( N \), the \( s(n) = n + 1 \) is in \( N \)

Observation

- Structural Induction \( \iff \) Weak Induction

Theorem: Structural Induction on Recursive Schemes \( \iff \) Weak Induction

Proof: \( \Rightarrow \): Weak Induction follows from Structural Induction because weak induction is structural induction on the natural numbers.

\( \Leftarrow \): Structural Induction follows from weak induction by induction on the number of operations = number of recursive steps.
Structural Induction on Pairs of Natural Numbers

Lexicographic Order on $N \times N$

- Think order on two letter words
  - at, in, it, an
  - (2,3), (9,7), (2,7), (7,7)

Well Ordering on $N \times N$

- Every non-empty subset of $N \times N$ has a smallest element.
- But there are infinitely many elements smaller than any element in $N \times N$
  - List all elements less than (4,7)
**Well Ordering on** $N \times N$

*Theorem:* Every non-empty subset of $N \times N$ has a smallest element.

*Proof:* Let

$$S = \text{a nonempty subset of } N \times N.$$  

$$S_1 = \{ s \in N \mid \text{there is a number } t \text{ such that } (s, t) \in S \}$$

$s^* = \text{smallest element in } S_1$

$$S_2 = \{ t \in N \mid (s^*, t) \in S_1 \}$$

$t^* = \text{smallest element in } S_2$

*Claim:* $(s^*, t^*) = \text{smallest element in } S$

*Proof:* $s^* = \text{smallest } s$, and for this smallest $s$, $t^* = \text{the smallest } t.$
**Strong Induction on Pairs of Natural Numbers**

Let $P(m,n)$ be a statement about the pair of integers $(m,n)$.

If the following hypotheses hold

i. Base Case: $P(0,0)$

ii. Recursive Step: $P(a,b)$ for all $(a,b) < (c,d) \Rightarrow P(c,d)$

Then we can conclude that

iii. $P(m,n)$ is True for every pair of integers $(m,n)$

Proof: By Well Ordering Principle:

There is no smallest element where $P(m,n)$ is False.
Recursive Definition

- $a_{0,0} = 0$
- $a_{m,0} = a_{m-1,0} + 1$
- $a_{m,n} = a_{m,n-1} + n \quad n > 0$

Theorem: $a_{m,n} = m + n(n+1)/2$

Proof: By Strong Induction on $N \times N$.

Base Case: Obvious. ($0 = 0$)

Recursion: Two cases:

Case 1: $a_{m,0} = a_{m-1,0} + 1 = (m-1) + 1 = m$.

Case 2: $a_{m,n} = a_{m,n-1} + n = m + (n-1)n/2 + n = m + n(n+1)/2$. 
Bad Recursive Definitions

Legal Definitions

• New objects must be built from objects already in the set

Incorrect Example: Strings with more 0’s than 1’s

• Base Case: 0
• Recursive Step: 0S, S0, where S has same number of 0’s and 1’s.

Observation

• This recursive definition is not legal, since S is not in the set!
• Need to tell how S is constructed!