Session 49: Structural Induction

- Principle of structural induction
- Examples

Structural Induction

To prove a property of the elements of a recursively defined set, we use **structural induction**.

BASIS STEP: Show that the result holds for all elements specified in the basis step of the recursive definition.

RECURSIVE STEP: Show that if the statement is true for each of the elements used to construct new elements in the recursive step of the definition, the result holds for these new elements.

- The validity of structural induction can be shown to follow from the principle of mathematical induction.

Example

Theorem: *l*(*xy*) = *l*(*x*) + *l*(*y*), where *x* and *y* belong to Σ*, the set of strings over the alphabet Σ.

Proof: $P(y)$ states $\ell(xy) = \ell(x) + \ell(y)$, structural induction on y for $x \in \Sigma^*$

Base Step: $P(\lambda)$: $\ell(x\lambda) = \ell(x) = \ell(x) + 0 = \ell(x) + \ell(\lambda)$ for $x \in \Sigma^*$

Recursive Step $P(y)$ is true, show that $P(ya)$ is true for $a \in \Sigma$.

$$\ell(x(ya)) \underset{\substack{\uparrow \\ \text{recursive definition} \\ \text{of string concatenation}}}{=} \ell((xy)a) \underset{\substack{\uparrow \\ \text{rec. def. of } \ell}}{=} \ell(xy) + 1 \underset{\substack{\uparrow \\ \text{IH}}}{=} \ell(x) + \ell(y) + 1 \underset{\substack{\uparrow \\ \text{rec. def of } \ell}}{=} \ell(x) + \ell(ya)$$

Thus $P(ya)$ done

Example

Theorem: Every well-formed formula for compound propositions contains an equal number of left and right parentheses.

Proof:

BASIS STEP*:* Each of the formula **T**, **F**, and *s* contains no parentheses, so they contain an equal number of left and right parentheses.

RECURSIVE STEP*:* Assume *p* and *q* are well-formed formulae, each containing an equal number of left and right parentheses.

For the two propositions *p* and *q*, let l_p and l_q be the number of left parenthesis and r_p and r_q the number of right parenthesis. The inductive hypothesis is that $l_p = r_p$ and $l_q = r_q$

We need to show that each of $(\neg p)$, $(p \vee q)$, $(p \wedge q)$, $(p \rightarrow q)$, and $(p \leftrightarrow q)$ also contains an equal number of left and right parentheses.

The number of left parentheses in $(\neg p)$ equals $l_p + 1$ and the number of right parentheses $r_p + 1$.

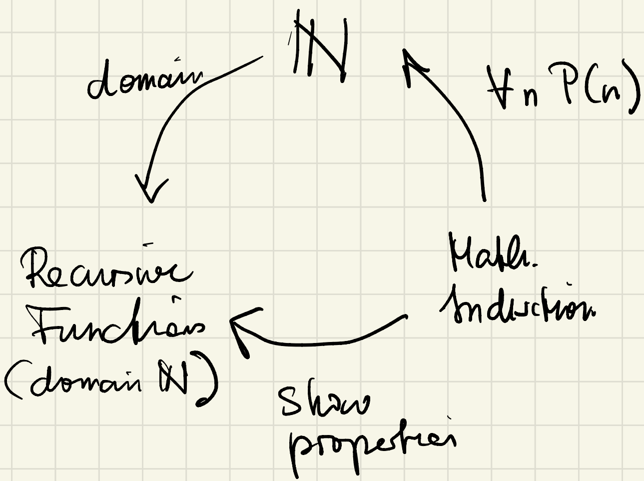
For the other compound propositions the number of left parentheses equals $l_p + l_q + 1$ and the number of right parentheses equals $r_p + r_q + 1$

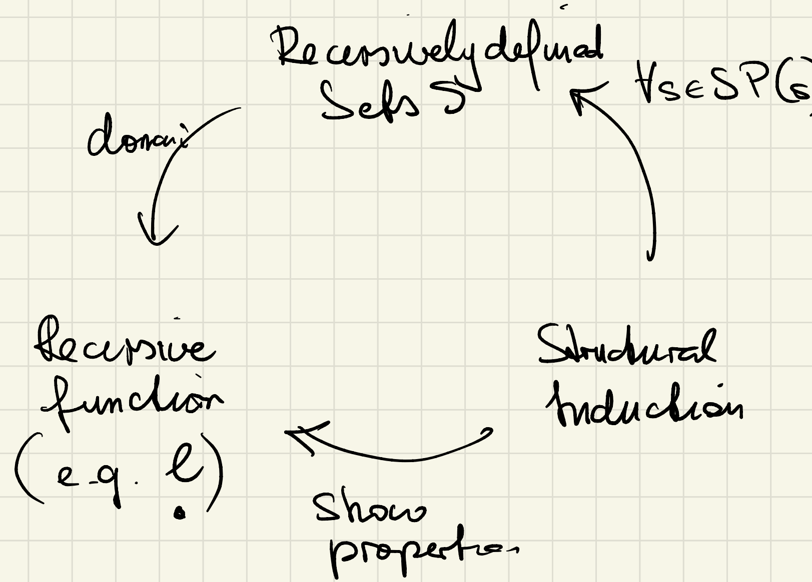
Since $l_p = r_p$ and $l_q = r_q$ in all cases the these compound expressions contains the same number of left and right parentheses.

This completes the proof by structural induction. ◀

Summary

- Principle of structural induction
- Structural induction on strings
- Structural induction on well-formed formulae


$\mathbb{N}$
domain
Recursive Functions (domain $\mathbb{N}$)
Math. Induction
$\forall n\, P(n)$
Show properties


Recursively defined Sets S
$\forall s \in S\, P(s)$
domain
Recursive function (e.g. ℓ)
Structural Induction
Show properties