Induction

Wolfgang Schreiner
Engineering for Computer-based Learning
University of Applied Sciences at Hagenberg

Wolfgang.Schreiner@fh-hagenberg.at
http://cbl.fh-hagenberg.at/~schreine

Overview

- Inductive Definitions
- Induction as a Proof Technique
- Induction on Sets
  - Inductive set definitions.
  - Inductive function/predicate definitions.
  - Induction proofs on sets.

Inductive Definitions

Situation

- Recursive definition on $\mathbb{N}$:

$$
\begin{aligned}
&* : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N} \\
&x * y := \textbf{if } y = 0 \textbf{ then } 0 \textbf{ else } x + (x * y^{-})
\end{aligned}
$$

- Proposition:

$$
\begin{aligned}
x * 0 &= 0, \\
x * y' &= x + (x * y).
\end{aligned}
$$

- Usually written as:

$$
\begin{aligned}
x * 0 &= 0, \\
x * (y + 1) &= x + (x * y).
\end{aligned}
$$

Recursive function definition implies pair of equations.

Idea

Also converse is true:

- For each $a * b$, left hand side of only one equation "matches":
  - Either $b = 0$ or $b = y'$ for some y.
  - Consequence of first Peano axiom.
- Equality $b = y'$ determines unique y.
  - $(b = {y_0}' \wedge b = {y_1}') \Rightarrow y_0 = y_1$ for all y_0 and y_1.
  - Consequence of second Peano axiom.

Pair of equations uniquely determines a function.

Alternative Definition Format

Definition by pair of equations ("induction on second argument"):

$$
\begin{aligned}
&* : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}\\
&x * 0 := 0,\\
&x * (y + 1) := x + (x * y).
\end{aligned}
$$

By syntactic restriction of the equations, the function is well-defined.

Inductive Function Definitions

Definition: An inductive definition over $\mathbb{N}$ of an n-ary function f:

$$\begin{aligned} f(x_0, \ldots, 0, \ldots, x_{n-1}) &:= T_\mathrm{b}, \\ f(x_0, \ldots, x_i + 1, \ldots, x_{n-1}) &:= T_\mathrm{r}. \end{aligned}$$

- f does not occur in base term T_b.
- Every application of f in recursion term T_r has form

$$f(T_0, \ldots, x_i, \ldots, T_{n-1}).$$

- Free variables of terms must occur in definiendum.

Induction runs over x_i.

Induction with Larger Decrements

Example: Fibonacci Numbers

$$\begin{aligned} \mathrm{fib}(0) &:= 1, \\ \mathrm{fib}(1) &:= 1, \\ \mathrm{fib}(x+2) &:= \mathrm{fib}(x) + \mathrm{fib}(x+1). \end{aligned}$$

$$\mathrm{fib} = [1, 1, 2, 3, 5, 8, 13, 21, \ldots]$$

All possible base cases must be covered!

Induction over Multiple Arguments

Examples:

$$\begin{aligned} f(0,0) &:= 0, \\ f(x+1,0) &:= 1 + f(x,0), \\ f(x,y+1) &:= 1 + f(x,y). \end{aligned}$$

$$\begin{aligned} f(0,0) &:= 0, \\ f(x+1,0) &:= 1 + f(x,0), \\ f(0,y+1) &:= 1 + f(0,y), \\ f(x+1,y+1) &:= 2 + f(x,y), \end{aligned}$$

All possible base cases must be covered!

Example

- Ackerman's function.

$$\begin{aligned} A(0,y) &:= y+1, \\ A(x+1,0) &:= A(x,1), \\ A(x+1,y+1) &:= A(x, A(x+1,y)). \end{aligned}$$

- Well-founded lexicographic ordering:

$$x \prec y :\Leftrightarrow x_0 < y_0 \vee (x_0 = y_0 \wedge x_1 < y_1)$$

- Correctness:

  1. $\langle x, 1\rangle \prec \langle x+1, 0\rangle$,
  2. $\langle x+1, y\rangle \prec \langle x+1, y+1\rangle$,
  3. $\langle x, A(x+1,y)\rangle \prec \langle x+1, y+1\rangle$.

Inductive Predicate Definitions

Definition: An inductive definition over $\mathbb{N}$ of an n-ary predicate p:

$$\begin{aligned} p(x_0, \ldots, 0, \ldots, x_{n-1}) &:\Leftrightarrow F_{\mathrm{b}}, \\ p(x_0, \ldots, x_i + 1, \ldots, x_{n-1}) &:\Leftrightarrow F_{\mathrm{r}}. \end{aligned}$$

- p does not occur in base formula F_{b}.
- Every application of p in recursion formula F_{r} has form

$$p(T_0, \ldots, x_i, \ldots, T_{n-1}).$$

- Free variables of terms must occur in definiendum.

Induction runs over x_i.

Example

We can introduce the predicate $\text{iseven}(x) :\Leftrightarrow 2|x$ also as

$$\begin{aligned} \text{iseven}(0) &:\Leftrightarrow \text{T}, \\ \text{iseven}(x+1) &:\Leftrightarrow \neg\text{iseven}(x). \end{aligned}$$

or as

$$\begin{aligned} \text{iseven}(0) &:\Leftrightarrow \text{T}, \\ \text{iseven}(1) &:\Leftrightarrow \text{F}, \\ \text{iseven}(x+2) &:\Leftrightarrow \text{iseven}(x). \end{aligned}$$

$$\text{iseven} = [\text{T}, \text{F}, \text{T}, \text{F}, \text{T}, \ldots]$$

Induction as a Proof Technique

Mathematical Induction

Third Peano Axiom:

$$(F[x \leftarrow 0] \wedge (\forall x \in \mathbb{N} : F \Rightarrow F[x \leftarrow x+1])) \Rightarrow \forall x \in \mathbb{N} : F.$$

Proposition: In order to prove

$$\forall x \in \mathbb{N} : F,$$

it suffices to prove

1. $F[x \leftarrow 0]$,
2. $\forall x \in \mathbb{N} : F \Rightarrow F[x \leftarrow x+1]$.

Typical Format

We want to prove

$$\forall x \in \mathbb{N} : F.$$

1. Induction Base: We show $F[x \leftarrow 0]$.
2. Induction Hypothesis: We take arbitrary $x \in \mathbb{N}$ and assume F.
3. Induction Step: We show $F[x \leftarrow x + 1]$.

Proof strategy for formulas that are universally quantified over $\mathbb{N}$.

Example

We prove by induction on n

$$\forall n \in \mathbb{N} : n < 2^n.$$

The induction base holds because $0 < 1 = 2^0$.
Now we take arbitrary $n \in \mathbb{N}$ and assume (induction hypothesis)

$$(1)\ n < 2^n.$$

We have to show (induction step)

$$(2)\ n + 1 < 2^{n+1}.$$

By (1) we have

$$n + 1 < 2^n + 1$$

and therefore

$$n + 1 < 2^n + 1 \leq 2^n + 2^n = 2 * 2^n = 2^{n+1}$$

which implies (2).

Example

We prove by induction on n

$$\forall n \in \mathbb{N} : 3 | n^3 + 2n$$

The induction base holds because $3|0$ and $0 = 0^3 + 2 * 0$.
We take arbitrary $n \in \mathbb{N}$ and assume

$$(1)\ 3 | n^3 + 2n.$$

We have to show

$$(2)\ 3 | (n+1)^3 + 2(n+1).$$

Example (Continued)

By (1) and definition of $|$ we have some $a \in \mathbb{N}$ such that

$$(3)\ 3a = n^3 + 2n.$$

We therefore have

$$\begin{aligned}
(n+1)^3 + 2(n+1) &= \\
(n^3 + 3n^2 + 3n + 1) + (2n + 2) &= \\
(n^3 + 2n) + (3n^2 + 3n + 3) &= (3) \\
3a + 3(n^2 + n + 1) &= \\
3(a + n^2 + n + 1) &
\end{aligned}$$

which implies (2) by definition of $|$.

Example

We prove by induction on n

$$\forall n \in \mathbb{N} : \sum_{1 \leq i \leq n} i = \frac{(n+1)n}{2}$$

The induction base holds because

$$\sum_{1 \leq i \leq 0} i = 0 = \frac{(0+1)*0}{2}.$$

We take arbitrary $n \in \mathbb{N}$ and assume

$$(1) \quad \sum_{1 \leq i \leq n} i = \frac{(n+1)n}{2}.$$

Example (Continued)

We have to show

$$\sum_{1 \leq i \leq n+1} i = \frac{((n+1)+1)(n+1)}{2}.$$

$$\begin{aligned}
\textstyle\sum_{1 \leq i \leq n+1} i &= (\text{definition } \textstyle\sum) \\
\textstyle\sum_{1 \leq i \leq n} i + (n+1) &= (1) \\
\frac{(n+1)n}{2} + (n+1) &= \\
\frac{(n+1)n+2(n+1)}{2} &= \\
\frac{(n+1)(n+2)}{2} &= \\
\frac{(n+1)((n+1)+1)}{2}.
\end{aligned}$$

Example

We can prove by induction the "computing laws" in $\mathbb{N}$:

We prove

$$\forall x \in \mathbb{N}, y \in \mathbb{N}, z \in \mathbb{N} : x + (y + z) = (x + y) + z.$$

We take arbitrary $x \in \mathbb{N}$ and $y \in \mathbb{N}$ and prove by induction on z

$$\forall z \in \mathbb{N} : x + (y + z) = (x + y) + z.$$

We have by definition of $+$

$$x + (y + 0) = x + y = (x + y) + 0$$

and thus the induction base.

Example (Continued)

We assume

$$(1)\ x + (y + z) = (x + y) + z$$

and show

$$(2)\ x + (y + (z + 1)) = (x + y) + (z + 1).$$

We have

$$\begin{aligned}
(x + y) + (z + 1) &= \text{(definition } +) \\
((x + y) + z) + 1 &= (1) \\
(x + (y + z)) + 1 &= \text{(definition } +) \\
x + ((y + z) + 1) &= \text{(definition } +) \\
x + (y + (z + 1))
\end{aligned}$$

which implies (2).

Example

Theorem: The number of permutations of length n is $n!$:

$$\forall n \in \mathbb{N} : |\{f : f \text{ is permutation of length } n\}| = n!.$$

Proof: We proceed by induction on n.

If n=0, then the only permutation is $p = [\,]$.

Assume $|\{f : f \text{ is permutation of length } n\}| = n!$.

Example (Continued)

We define a function which returns the sequence constructed from f by inserting element x at position i.

$$
\begin{aligned}
&\mathrm{insert}(x, i, f) := \\
&\quad \textbf{such } s : \\
&\qquad \mathrm{length}(s) = 1 + \mathrm{length}(f) \wedge \\
&\qquad \forall j \in \mathbb{N}_{n+1} : \\
&\qquad\quad j < i \Rightarrow s(j) = f(j) \wedge \\
&\qquad\quad j = i \Rightarrow s(j) = x \wedge \\
&\qquad\quad j > i \Rightarrow s(j) = f(j-1)
\end{aligned}
$$

Example (Continued)

Then we have:

$$
\begin{aligned}
|\{f : f \text{ is permutation of length } n+1\}| &= \\
|\{\mathrm{insert}(n+1, i, f) : i \in \mathbb{N}_{n+1} \wedge f \text{ is permutation of length } n\}| &= \\
(n+1) * |\{f : f \text{ is permutation of length } n\}| &= \\
(n+1) * n! &= \\
(n+1)!
\end{aligned}
$$

Induction on Sets

Example

- Let "List(T)" be the set of all finite lists whose elements are from set T.
- Let "nil" denote the empty list and let "cons(e, l)" be the list with first element $e \in T$ and rest list $l \in \mathrm{List}(T)$.
- Then every element $l \in \mathrm{List}(T)$ is either "nil" or "cons(e, l')" for some unique $e \in T$ and $l' \in \mathrm{List}(T)$.
- Therefore we may inductively define functions/predicates on List(T) and we may prove a property $\forall x \in \mathrm{List}(T) : F$ by induction.

Induction can be useful for sets other than $\mathbb{N}$.

Inductive Set Definition

Definition: An inductive definition of a set S is a collection of formulas

$$
\begin{array}{l}
(\forall x_1, \ldots, x_{m_1}, y_1 \in S, \ldots, y_{n_1} \in S : \\
\quad f_1(x_1, \ldots, x_{m_1}, y_1, \ldots, y_{n_1}) \in S) \\
, \;\ldots\; , \\
(\forall x_1, \ldots, x_{m_c}, y_1 \in S, \ldots, y_{n_c} \in S : \\
\quad f_c(x_1, \ldots, x_{m_c}, y_1, \ldots, y_{n_c}) \in S)
\end{array}
$$

We call the function constants $f_1, \ldots, f_c$ the constructors of S.

Defined Set

S is the smallest set on which the conjunction of these formulas holds, i.e., every element of S is described by a constructor term

$$f_i(T_1, \ldots, T_{m_i}, S_1, \ldots, S_{n_i})$$

for some terms $T_1, \ldots, T_{m_i}, S_1, \ldots, S_{n_i}$ where the $S_1, \ldots, S_{n_i}$ are also such constructor terms.

Example

The set $\mathbb{N}$ is inductively defined by

$$0 \in \mathbb{N},$$
$$\forall x \in \mathbb{N} : x' \in \mathbb{N}$$

with constructors 0 and '.

Every element of $\mathbb{N}$ is of the form

$$0^{'\cdots'},$$

e.g. the number 4 in $\mathbb{N}$ is denoted by $0''''$.

Example

For every set T, the set $\mathrm{List}(T)$ is defined by

$$\begin{aligned}&\mathrm{nil} \in \mathrm{List}(T),\\ &\forall e \in T, l \in \mathrm{List}(T) : \mathrm{cons}(e, l) \in \mathrm{List}(T).\end{aligned}$$

with constructors nil and cons.

Every element of $\mathrm{List}(T)$ is of the form

$$\mathrm{cons}(e_0, \ldots, \mathrm{cons}(e_{n-1}, \mathrm{nil})),$$

e.g. the list $[2, 3]$ in $\mathrm{List}(\mathbb{N})$ is denoted by $\mathrm{cons}(2, \mathrm{cons}(3, \mathrm{nil}))$.

Example

For every set T, the set $\mathrm{Tree}(T)$ is defined by

$$\mathrm{empty} \in \mathrm{Tree}(T),$$
$$\forall e \in T, l \in \mathrm{Tree}(T), r \in \mathrm{Tree}(T) : \mathrm{node}(e, l, r) \in \mathrm{Tree}(T).$$

with constructors empty and node.
Every element of $\mathrm{Tree}(T)$ is of the form

$$\mathrm{node}(n_0, \mathrm{node}(n_{11}, \ldots), \mathrm{node}(n_{21}, \ldots)),$$

1

2 5

3 4

$\mathrm{node}(1, \mathrm{node}(2, \mathrm{node}(3, \mathrm{empty}, \mathrm{empty}), \mathrm{node}(4, \mathrm{empty}, \mathrm{empty})), \mathrm{node}(5, \mathrm{empty}, \mathrm{empty}))$

Term

The set Term is defined by

$$\begin{aligned}
&0 \in \mathrm{Term}, \\
&1 \in \mathrm{Term}, \\
&\forall x \in \mathrm{Term} : -x \in \mathrm{Term}, \\
&\forall x \in \mathrm{Term}, y \in \mathrm{Term} : x + y \in \mathrm{Term}, \\
&\forall x \in \mathrm{Term}, y \in \mathrm{Term} : x * y \in \mathrm{Term}
\end{aligned}$$

with constructors $0, 1, -, +, *$.

An element of Term is $1 + (1 + 0) * 1$.

Formula

The set $\mathrm{Formula}$ is defined by

$$\begin{aligned}
&\mathrm{T} \in \mathrm{Formula},\\
&\forall x \in \mathrm{Formula} : \mathrm{not}(x) \in \mathrm{Formula},\\
&\forall x \in \mathrm{Formula}, y \in \mathrm{Formula} : \mathrm{and}(x, y) \in \mathrm{Formula},\\
&\forall x \in \mathrm{Variable}, y \in \mathrm{Formula} : \mathrm{forall}(x, y) \in \mathrm{Formula}
\end{aligned}$$

with constructors "T", "not", "and", "forall".

An element of $\mathrm{Formula}$ is $\mathrm{forall}(\mathrm{X}, \mathrm{and}(\mathrm{T}, \mathrm{not}(\mathrm{T})))$ (assuming $\mathrm{X} \in \mathrm{Variable}$).

Term Algebra

An inductively defined set is a term algebra if we have for every constructor f of this set

$$\forall x, y : f(x) = f(y) \Rightarrow x = y$$

i.e., different arguments are mapped to different results.

Furthermore, for all constructors f and g

$$\forall x, y : f(x) \neq g(y)$$

i.e., different constructors yield different results.

Consequence

- Every element of a term algebra is denoted by exactly one constructor term
$$f_i(T_1, \ldots, T_{m_i}, S_1, \ldots, S_{n_i})$$
for some terms $T_1, \ldots, T_{m_i}, S_1, \ldots, S_{n_i}$ where the $S_1, \ldots, S_{n_i}$ are also constructor terms.
- One to one correspondence between terms and set elements.

We may define functions and predicates in term algebras inductively.

Example

Take the set $\mathrm{List}(T)$ defined in the previous example and assume that it is a term algebra. We define the length of a list as

$$\begin{aligned}
&\mathrm{length} : \mathrm{List}(T) \rightarrow \mathbb{N} \\
&\mathrm{length}(\mathrm{nil}) := 0 \\
&\mathrm{length}(\mathrm{cons}(e, l)) := 1 + \mathrm{length}(l).
\end{aligned}$$

Then we have $\mathrm{length}(\mathrm{cons}(1, \mathrm{cons}(2, \mathrm{nil}))) = 2$.

Example

Take the set Term defined in the previous example and assume that it is a term algebra. We define the value of a term as

$$
\begin{aligned}
&\mathrm{value} : \mathrm{Term} \to \mathbb{N}\\
&\mathrm{value}(0) := 0_{\mathbb{N}}\\
&\mathrm{value}(1) := 1_{\mathbb{N}}\\
&\mathrm{value}(-x) := -_{\mathbb{N}}\mathrm{value}(x)\\
&\mathrm{value}(x + y) := \mathrm{value}(x) +_{\mathbb{N}} \mathrm{value}(y)\\
&\mathrm{value}(x * y) := \mathrm{value}(x) *_{\mathbb{N}} \mathrm{value}(y)
\end{aligned}
$$

Then we have $\mathrm{value}(1 + (1 + 0) * 1) = 2$.

Generalized Induction Principle

We want to prove

$$\forall x \in S : F.$$

Idea: every element x in S is denoted by some term

$$f_i(x_1, \ldots, x_{m_i}, y_1, \ldots, y_{n_i}).$$

Let the induction run over the structure of every such term:

- assume that F holds for every "S-component" y_j of x, and
- show that F is propagated to x itself.

Structural Induction

Proposition: In order to prove a property

$$\forall x \in S : F$$

for an inductively defined set S, it suffices to prove

$$\begin{aligned}&\forall x_1, \ldots, x_{m_i}, y_1 \in S, \ldots, y_{n_i} \in S : \\ &\quad (F[x := y_1] \wedge \ldots \wedge F[x := y_{n_i}]) \Rightarrow \\ &\quad\quad F[x := f_i(x_1, \ldots, x_{m_i}, y_1, \ldots, y_{n_i})]\end{aligned}$$

for every constructor f_i of S.

Example

Take the set $\text{List}(T)$ defined inductively as

$$
\begin{aligned}
&\text{nil} \in \text{List}(T),\\
&\forall e \in T, l \in \text{List}(T) : \text{cons}(e, l) \in \text{List}(T).
\end{aligned}
$$

We define

$$
\begin{aligned}
&\text{append} : \text{List}(T) \times \text{List}(T) \to \text{List}(T)\\
&\text{append}(\text{nil}, y) := y\\
&\text{append}(\text{cons}(e, x), y) := \text{cons}(e, \text{append}(x, y))
\end{aligned}
$$

and claim that the following holds:

$$
\begin{aligned}
&\forall x \in \text{List}(T), y \in \text{List}(T) :\\
&\quad \text{length}(\text{append}(x, y)) = \text{length}(x) + \text{length}(y).
\end{aligned}
$$

Example (Continued)

We proceed by structural induction on x:

Case $x = \text{nil}$**:** We have to show

$$\forall y \in \text{List}(T) : \\ \quad \text{length}(\text{append}(\text{nil}, y)) = \text{length}(\text{nil}) + \text{length}(y).$$

Take arbitrary $y \in \text{List}(T)$. We have

$$\begin{aligned} \text{length}(\text{append}(\text{nil}, y)) &= \text{(definition append)} \\ \text{length}(y) &= \\ 0 + \text{length}(y) &= \text{(definition length)} \\ \text{length}(\text{nil}) + \text{length}(y). \end{aligned}$$

Example (Continued)

Case $x = \text{cons}(e, l)$**:** Take arbitrary $e \in T$ and $l \in \text{List}(T)$.

We assume (induction hypothesis)

$$\forall y \in \text{List}(T) : \\ \text{length}(\text{append}(l, y)) = \text{length}(l) + \text{length}(y)$$

and have to show

$$\forall y \in \text{List}(T) : \\ \text{length}(\text{append}(\text{cons}(e, l), y)) = \text{length}(\text{cons}(e, l)) + \text{length}(y).$$

Example (Continued)

Take arbitrary $y \in \mathrm{List}(T)$. We have

$$\begin{aligned}
\mathrm{length}(\mathrm{append}(\mathrm{cons}(e,l),y)) &= \text{(definition append)}\\
\mathrm{length}(\mathrm{cons}(e,\mathrm{append}(l,y))) &= \text{(definition length)}\\
1+\mathrm{length}(\mathrm{append}(l,y)) &= \text{(induction hypothesis)}\\
1+(\mathrm{length}(l)+\mathrm{length}(y)) &= \\
(1+\mathrm{length}(l))+\mathrm{length}(y) &= \text{(definition length)}\\
\mathrm{length}(\mathrm{cons}(e,l))+\mathrm{length}(y). &
\end{aligned}$$