Induction with Recursive Definitions in Superposition

Márton Hajdu[1], Petra Hozzová[1], Laura Kovács[1], Andrei Voronkov[1,2,3]

[1]TU Wien

[2]The University of Manchester

[3]EasyChair

fmcad.21

Motivation

```
datatype nat = 0 | s of nat
datatype list = nil | cons of nat list
datatype btree = leaf | node of btree nat btree

fun add 0 y = y
    add (s x) y = s (add x y)
fun mult 0 y = 0
    mult (s x) y = add (mult x y) y
fun even 0 = ⊤
    even (s 0) = ⊥
    even (s (s x)) = even x

fun app nil z = z
    app (cons x y) z = cons x (app y z)
fun flat leaf = nil
    flat (node x y z) = app (flat x) (cons y (flat z))
fun aflat leaf u = u
    aflat (node x y z) u = aflat x (cons y (aflat z u))

assert (∀x, y)(app (flat x) y = aflat x y)
assert (∀x, y)(even y → even (mult x y))
```

Motivation

```
datatype nat = 0 | s of nat
datatype list = nil | cons of nat list
datatype btree = leaf | node of btree nat btree

fun add 0 y = y
    add (s x) y = s (add x y)
fun mult 0 y = 0
    mult (s x) y = add (mult x y) y
fun even 0 = ⊤
    even (s 0) = ⊥
    even (s (s x)) = even x

fun app nil z = z
    app (cons x y) z = cons x (app y z)
fun flat leaf = nil
    flat (node x y z) = app (flat x) (cons y (flat z))
fun aflat leaf u = u
    aflat (node x y z) u = aflat x (cons y (aflat z u))

assert (∀x, y)(app (flat x) y = aflat x y)
assert (∀x, y)(even y → even (mult x y))
```

$\rightarrow$ Induction

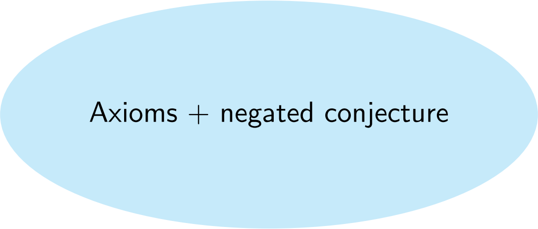
Motivation

```
datatype nat = 0 | s of nat
datatype list = nil | cons of nat list
datatype btree = leaf | node of btree nat btree

fun add 0 y = y
    add (s x) y = s (add x y)
fun mult 0 y = 0
    mult (s x) y = add (mult x y) y
fun even 0 = ⊤
    even (s 0) = ⊥
    even (s (s x)) = even x

fun app nil z = z
    app (cons x y) z = cons x (app y z)
fun flat leaf = nil
    flat (node x y z) = app (flat x) (cons y (flat z))
fun aflat leaf u = u
    aflat (node x y z) u = aflat x (cons y (aflat z u))

assert (∀x, y)(app (flat x) y = aflat x y)
assert (∀x, y)(even y → even (mult x y))
```

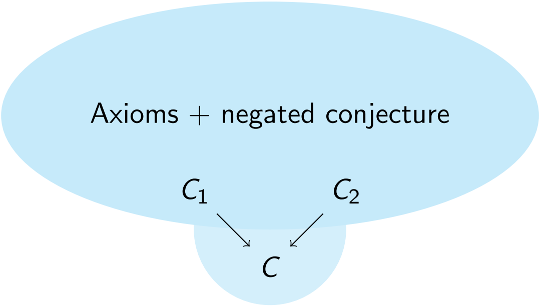
$\rightarrow$ Induction

$\rightarrow$ Saturation & Superposition

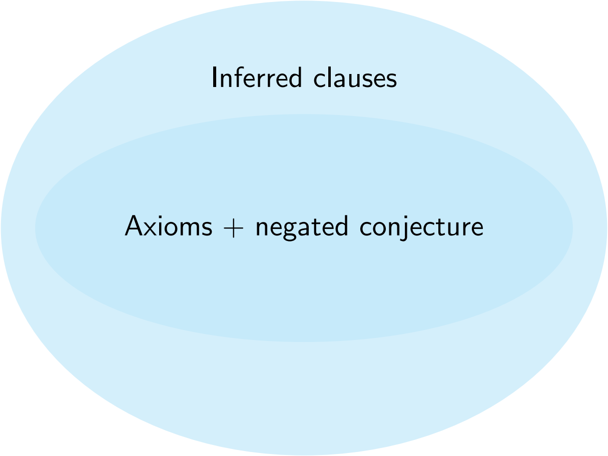
Saturation-based theorem proving



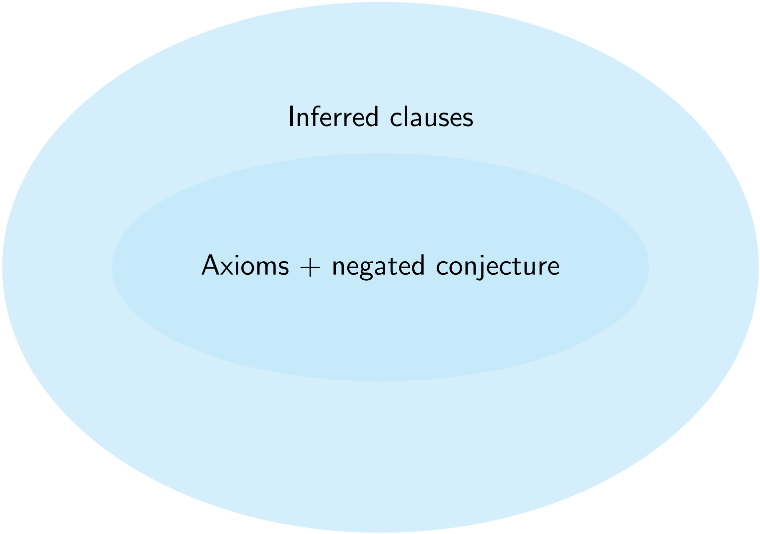
Saturation-based theorem proving



Saturation-based theorem proving

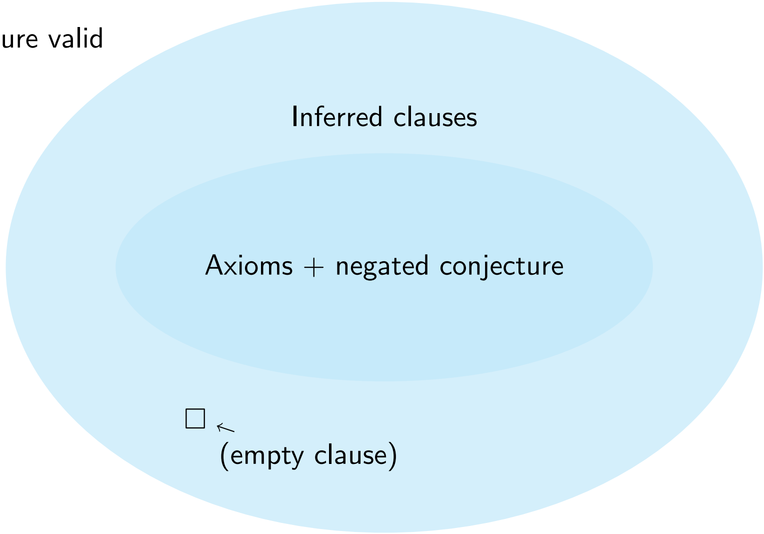


Saturation-based theorem proving

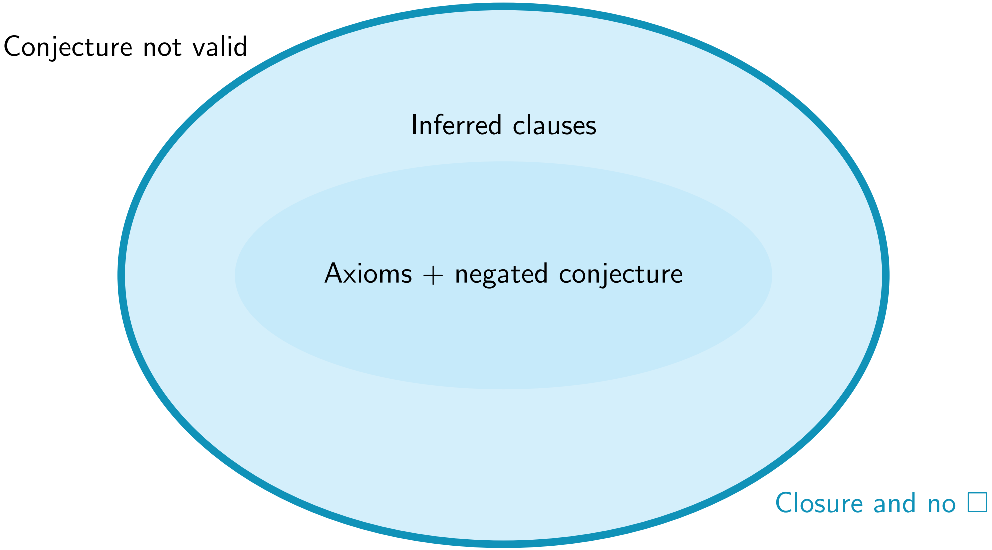


Saturation-based theorem proving

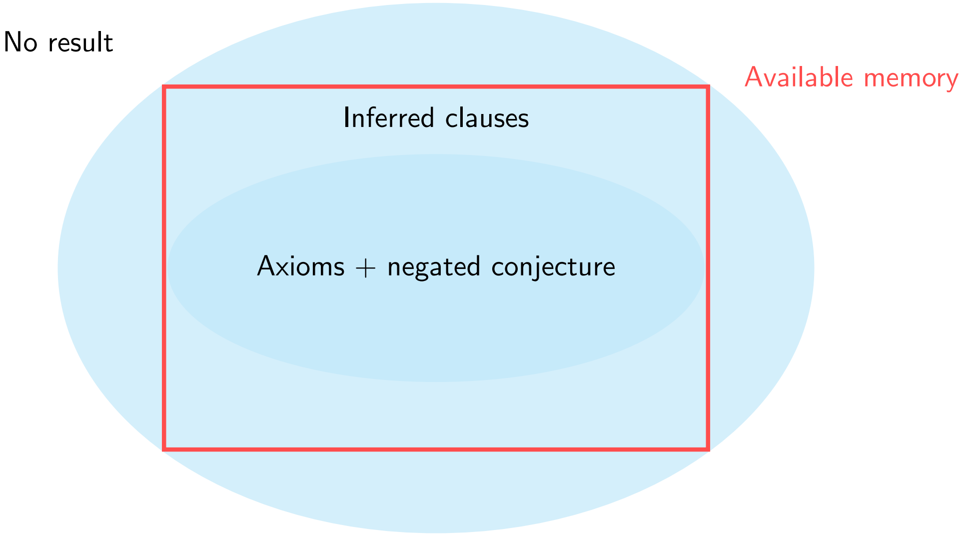
Conjecture valid



Saturation-based theorem proving



Saturation-based theorem proving



Saturation-based theorem proving

Generating Inferences

Example:

$$\frac{L \vee C \quad \neg L' \vee D}{(C \vee D)\theta}\ (BR)$$

where $\theta := \mathtt{mgu}(L, L')$.

Saturation-based theorem proving

Generating Inferences

Example:

$$\frac{L \vee C \quad \neg L' \vee D}{(C \vee D)\theta} \; (BR)$$

where $\theta := \mathtt{mgu}(L, L')$.

Simplifying inferences

Example:

$$\frac{l = r \quad \cancel{L[l\theta] \vee D}}{L[r\theta] \vee D} \; (Dem)$$

where $l\theta \succ r\theta$ and $L[l\theta] \vee D \succ l\theta = r\theta$.

Motivating example I.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

Motivating example I.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

---

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

Motivating example I.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

---

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

[CADE'19]

$$\frac{\overline{L}[\overline{t}] \vee C}{\texttt{cnf}(F \rightarrow \forall \overline{y}.L[\overline{y}])}\ (\texttt{Ind})$$

- L is ground
- $F \rightarrow \forall \overline{y}.L[\overline{y}]$ is valid induction formula

Motivating example I.

$$\underline{\textbf{Prove:}} \quad \forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$$

---

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

[CADE'19]

$$\frac{\overline{L}[\overline{t}] \vee C}{\texttt{cnf}(F \rightarrow \forall \overline{y}.L[\overline{y}])} \ (\texttt{Ind})$$

- L is ground
- $F \rightarrow \forall \overline{y}.L[\overline{y}]$ is valid induction formula

$$\left(\begin{array}{c} L[\texttt{leaf}] \wedge \\ \forall x, y, z. \left(\begin{array}{c} (L[x] \wedge L[z]) \rightarrow \\ L[\texttt{node}(x, y, z)] \end{array} \right) \end{array} \right) \rightarrow \forall u.L[u]$$

Motivating example I.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

[CADE'19]

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

2. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
 $\vee\ \texttt{app}(\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2)$
 $\neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [Ind 1]
 $\vee\ \texttt{app}(\texttt{flat}(x), \sigma_2) = \texttt{aflat}(x, \sigma_2)$

3. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
 $\vee\ \texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$ [Ind 1]
 $\vee\ \texttt{app}(\texttt{flat}(x), \sigma_2) = \texttt{aflat}(x, \sigma_2)$

4. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
 $\vee\ \texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$ [Ind 1]
 $\vee\ \texttt{app}(\texttt{flat}(x), \sigma_2) = \texttt{aflat}(x, \sigma_2)$

$$\frac{\overline{L}[\overline{t}] \vee C}{\texttt{cnf}(F \to \forall \overline{y}.L[\overline{y}])}\ (\texttt{Ind})$$

- L is ground
- $F \to \forall \overline{y}.L[\overline{y}]$ is valid induction formula

$$\left(\forall x, y, z.\ \begin{array}{c} L[\texttt{leaf}] \wedge \\ \left(\begin{array}{c}(L[x] \wedge L[z]) \to \\ L[\texttt{node}(x, y, z)]\end{array}\right)\end{array}\right) \to \forall u. L[u]$$

Motivating example I.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

[CADE'19]

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

2. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
   $\vee\ \texttt{app}(\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2)$
   $\neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [Ind 1]
   $\vee\ \texttt{app}(\texttt{flat}(x), \sigma_2) = \texttt{aflat}(x, \sigma_2)$

3. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
   $\vee\ \texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$ [Ind 1]
   $\vee\ \texttt{app}(\texttt{flat}(x), \sigma_2) = \texttt{aflat}(x, \sigma_2)$

4. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
   $\vee\ \texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$ [Ind 1]
   $\vee\ \texttt{app}(\texttt{flat}(x), \sigma_2) = \texttt{aflat}(x, \sigma_2)$

$$\frac{\overline{L}[\overline{t}] \vee C}{\texttt{cnf}(F \rightarrow \forall \overline{y}.L[\overline{y}])}\ (\texttt{Ind})$$

- L is ground
- $F \rightarrow \forall \overline{y}.L[\overline{y}]$ is valid induction formula

$$\left(\forall x, y, z. \begin{pmatrix} L[\texttt{leaf}] \wedge \\ (L[x] \wedge L[z]) \rightarrow \\ L[\texttt{node}(x, y, z)] \end{pmatrix}\right) \rightarrow \forall u.L[u]$$

Motivating example I.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

[CADE‘19]

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
   $\lor\ \texttt{app}(\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2)$
   $\neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [BR]

6. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
   $\lor\ \texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$ [BR]

7. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
   $\lor\ \texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$ [BR]

$$\frac{\overline{L}[\overline{t}] \lor C}{\texttt{cnf}(F \to \forall \overline{y}.L[\overline{y}])}\ (\texttt{Ind})$$

- L is ground
- $F \to \forall \overline{y}.L[\overline{y}]$ is valid induction formula

$$\left(\forall x, y, z. \begin{array}{c} L[\texttt{leaf}] \land \\ \big((L[x] \land L[z]) \to \\ L[\texttt{node}(x, y, z)]\big) \end{array}\right) \to \forall u.L[u]$$

Recursive function definitions

$$\textbf{fun } \texttt{app}(\texttt{nil}, z) = z$$
$$\mid \texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$$

Recursive function definitions

$$\textbf{fun}\ \mathtt{app}(\mathtt{nil}, z) = z$$
$$\mid\ \mathtt{app}(\mathtt{cons}(x, y), z) = \mathtt{cons}(x, \mathtt{app}(y, z))$$

Branches

Recursive calls

Inductive argument positions

Recursive function definitions

$$\textbf{fun}\ \mathtt{app}(\mathtt{nil}, z) = z$$
$$\mid\ \mathtt{app}(\mathtt{cons}(x, y), z) = \mathtt{cons}(x, \mathtt{app}(y, z))$$

Branches	$\mathcal{B}_{\mathrm{app}} := \{\mathtt{app}(\mathtt{nil}, z_0),\ \mathtt{app}(\mathtt{cons}(x, y), z_1)\}$
Recursive calls	
Inductive argument positions	

Recursive function definitions

$$\textbf{fun}\ \texttt{app}(\texttt{nil}, z) = z$$
$$\mid\ \texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$$

Branches	$\mathcal{B}_{\texttt{app}} := \{\texttt{app}(\texttt{nil}, z_0),\ \texttt{app}(\texttt{cons}(x, y), z_1)\}$
Recursive calls	$\mathcal{R}_{\texttt{app}(\texttt{nil}, z_0)} := \emptyset$ $\mathcal{R}_{\texttt{app}(\texttt{cons}(x,y), z_1)} := \{\texttt{app}(y, z_1)\}$
Inductive argument positions	

Recursive function definitions

$$\textbf{fun}\ \texttt{app}(\texttt{nil}, z) = z$$
$$\mid\ \texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$$

Branches	$\mathcal{B}_{\texttt{app}} := \{\texttt{app}(\texttt{nil}, z_0),\ \texttt{app}(\texttt{cons}(x, y), z_1)\}$
Recursive calls	$\mathcal{R}_{\texttt{app}(\texttt{nil}, z_0)} := \emptyset$ $\mathcal{R}_{\texttt{app}(\texttt{cons}(x,y), z_1)} := \{\texttt{app}(y, z_1)\}$
Inductive argument positions	$I_{\texttt{app}} := \{1\}$

Recursive function definitions - ctd.

$$\textbf{fun } \texttt{flat}(\texttt{leaf}) = \texttt{nil}$$
$$\mid \texttt{flat}(\texttt{node}(x, y, z)) = \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$$

Branches	
Recursive calls	
Inductive argument positions	

Recursive function definitions - ctd.

$$\textbf{fun } \texttt{flat}(\texttt{leaf}) = \texttt{nil}$$
$$\mid \texttt{flat}(\texttt{node}(x, y, z)) = \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$$

Branches	$\mathcal{B}_{\texttt{flat}} := \{\texttt{flat}(\texttt{leaf}),\ \texttt{flat}(\texttt{node}(x, y, z))\}$
Recursive calls	
Inductive argument positions	

Recursive function definitions - ctd.

$$\textbf{fun } \texttt{flat}(\texttt{leaf}) = \texttt{nil}$$
$$\mid \texttt{flat}(\texttt{node}(x, y, z)) = \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$$

Branches	$\mathcal{B}_{\texttt{flat}} := \{\texttt{flat}(\texttt{leaf}),\ \texttt{flat}(\texttt{node}(x, y, z))\}$
Recursive calls	$\mathcal{R}_{\texttt{flat}(\texttt{leaf})} := \emptyset$ $\mathcal{R}_{\texttt{flat}(\texttt{node}(x,y,z))} := \{\texttt{flat}(x),\ \texttt{flat}(z)\}$
Inductive argument positions	

Recursive function definitions - ctd.

$$\textbf{fun } \mathtt{flat}(\mathtt{leaf}) = \mathtt{nil}$$
$$\mid \mathtt{flat}(\mathtt{node}(x, y, z)) = \mathtt{app}(\mathtt{flat}(x), \mathtt{cons}(y, \mathtt{flat}(z)))$$

Branches	$\mathcal{B}_{\mathtt{flat}} := \{\mathtt{flat}(\mathtt{leaf}),\ \mathtt{flat}(\mathtt{node}(x, y, z))\}$
Recursive calls	$\mathcal{R}_{\mathtt{flat}(\mathtt{leaf})} := \emptyset$ $\mathcal{R}_{\mathtt{flat}(\mathtt{node}(x,y,z))} := \{\mathtt{flat}(x),\ \mathtt{flat}(z)\}$
Inductive argument positions	$I_{\mathtt{flat}} := \{1\}$

Recursive function definitions - ctd.

$$\textbf{fun}\ \texttt{aflat}(\texttt{leaf}, u) = u$$
$$\mid \texttt{aflat}(\texttt{node}(x, y, z), u) = \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$$

Branches

Recursive calls

Inductive argument positions

Recursive function definitions - ctd.

$$\textbf{fun}\ \texttt{aflat}(\texttt{leaf}, u) = u$$
$$\mid\ \texttt{aflat}(\texttt{node}(x, y, z), u) = \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$$

Branches	$\mathcal{B}_{\texttt{aflat}} := \{\texttt{aflat}(\texttt{leaf}, u_0),\ \texttt{aflat}(\texttt{node}(x, y, z), u_1)\}$
Recursive calls	
Inductive argument positions	

Recursive function definitions - ctd.

$$\textbf{fun}\ \texttt{aflat}(\texttt{leaf}, u) = u$$
$$\mid\ \texttt{aflat}(\texttt{node}(x, y, z), u) = \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$$

Branches	$\mathcal{B}_{\texttt{aflat}} := \{\texttt{aflat}(\texttt{leaf}, u_0),\ \texttt{aflat}(\texttt{node}(x, y, z), u_1)\}$
Recursive calls	$\mathcal{R}_{\texttt{aflat}(\texttt{leaf}, u_0)} := \emptyset$ $\mathcal{R}_{\texttt{aflat}(\texttt{node}(x,y,z),u_1)} := \left\{ \begin{array}{c} \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u_1))), \\ \texttt{aflat}(z, u_1) \end{array} \right\}$
Inductive argument positions	$I_{\texttt{aflat}} := \{1, 2\}$

Generating induction formulas - motivating example I. ctd.

1. $\mathrm{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2)$ [*conj*.]

Generating induction formulas - motivating example I. ctd.

$$1. \quad \texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2) \quad [\textit{conj.}]$$

$$\left(\qquad \qquad \qquad \qquad \right) \rightarrow$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathrm{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2) \quad [\mathit{conj.}]$$

I. create generating term: replace arguments in $\texttt{flat}(\sigma_1)$ in argument positions $I_{\texttt{flat}} = \{1\}$ with fresh variables

$$\texttt{flat}(u)$$

$$\Bigg(\qquad\qquad \Bigg) \rightarrow$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathrm{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2) \quad [\mathit{conj.}]$$

I. create generating term: replace arguments in $\mathtt{flat}(\sigma_1)$ in argument positions $I_{\mathtt{flat}} = \{1\}$ with fresh variables

$$\mathtt{flat}(u)$$

$$\left(\qquad\qquad\qquad\qquad \right) \to \mathrm{app}(\mathtt{flat}(u), \sigma_2) = \mathtt{aflat}(u, \sigma_2)$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathrm{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2) \quad [\mathit{conj}.]$$

II. unify generating term with each of the branches
$\mathcal{B}_{\mathtt{flat}} = \{\mathtt{flat}(\mathtt{leaf}),\ \mathtt{flat}(\mathtt{node}(x, y, z))\}$

$$\left(\qquad \right) \rightarrow \mathrm{app}(\mathtt{flat}(u), \sigma_2) = \mathtt{aflat}(u, \sigma_2)$$

Generating induction formulas - motivating example I. ctd.

1. $\mathrm{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2)$ [*conj.*]

II. unify generating term with each of the branches

$\mathcal{B}_{\mathtt{flat}} = \{\mathtt{flat}(\mathtt{leaf}),\ \mathtt{flat}(\mathtt{node}(x, y, z))\}$

$$\mathtt{mgu}(\mathtt{flat}(u), \mathtt{flat}(\mathtt{leaf})) = \{u \mapsto \mathtt{leaf}\}$$

$$\left(\qquad \right) \rightarrow \mathtt{app}(\mathtt{flat}(u), \sigma_2) = \mathtt{aflat}(u, \sigma_2)$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathrm{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2) \quad [conj.]$$

II. unify generating term with each of the branches
$\mathcal{B}_{\mathtt{flat}} = \{\mathtt{flat}(\mathtt{leaf}),\ \mathtt{flat}(\mathtt{node}(x, y, z))\}$

$$\mathrm{mgu}(\mathtt{flat}(u), \mathtt{flat}(\mathtt{leaf})) = \{u \mapsto \mathtt{leaf}\}$$

$$\left(\quad \mathrm{app}(\mathtt{flat}(\mathtt{leaf}), \sigma_2) = \mathtt{aflat}(\mathtt{leaf}, \sigma_2) \quad \right) \to \mathrm{app}(\mathtt{flat}(u), \sigma_2) = \mathtt{aflat}(u, \sigma_2)$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathrm{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2) \quad [conj.]$$

II. unify generating term with each of the branches
$\mathcal{B}_{\mathtt{flat}} = \{\mathtt{flat}(\mathtt{leaf}),\ \mathtt{flat}(\mathtt{node}(x, y, z))\}$

$$\mathrm{mgu}(\mathtt{flat}(u), \mathtt{flat}(\mathtt{leaf})) = \{u \mapsto \mathtt{leaf}\}$$
$$\mathrm{mgu}(\mathtt{flat}(u), \mathtt{flat}(\mathtt{node}(x, y, z))) = \{u \mapsto \mathtt{node}(x, y, z)\}$$

$$\begin{pmatrix} \mathrm{app}(\mathtt{flat}(\mathtt{leaf}), \sigma_2) = \mathtt{aflat}(\mathtt{leaf}, \sigma_2) \\ \\ \\ \end{pmatrix} \rightarrow \mathrm{app}(\mathtt{flat}(u), \sigma_2) = \mathtt{aflat}(u, \sigma_2)$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathrm{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2) \quad [\mathit{conj.}]$$

II. unify generating term with each of the branches
$\mathcal{B}_{\mathtt{flat}} = \{\mathtt{flat}(\mathtt{leaf}),\ \mathtt{flat}(\mathtt{node}(x, y, z))\}$

$$\mathrm{mgu}(\mathtt{flat}(u), \mathtt{flat}(\mathtt{leaf})) = \{u \mapsto \mathtt{leaf}\}$$
$$\mathrm{mgu}(\mathtt{flat}(u), \mathtt{flat}(\mathtt{node}(x, y, z))) = \{u \mapsto \mathtt{node}(x, y, z)\}$$

$$\begin{pmatrix} \mathrm{app}(\mathtt{flat}(\mathtt{leaf}), \sigma_2) = \mathtt{aflat}(\mathtt{leaf}, \sigma_2) \ \wedge \\ \\ \mathrm{app}(\mathtt{flat}(\mathtt{node}(x, y, z)), \sigma_2) = \mathtt{aflat}(\mathtt{node}(x, y, z), \sigma_2) \end{pmatrix} \rightarrow \mathrm{app}(\mathtt{flat}(u), \sigma_2) = \mathtt{aflat}(u, \sigma_2)$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathrm{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2) \quad [\mathit{conj.}]$$

III. unify generating term with each of the recursive calls

$\mathcal{R}_{\mathtt{flat(leaf)}} \cup \mathcal{R}_{\mathtt{flat(node}(x,y,z))} = \{\mathtt{flat}(x),\ \mathtt{flat}(z)\}$

$$\begin{pmatrix} \mathrm{app}(\mathtt{flat}(\mathtt{leaf}), \sigma_2) = \mathtt{aflat}(\mathtt{leaf}, \sigma_2) \ \wedge \\ \\ \mathrm{app}(\mathtt{flat}(\mathtt{node}(x, y, z)), \sigma_2) = \mathtt{aflat}(\mathtt{node}(x, y, z), \sigma_2) \end{pmatrix} \rightarrow \mathrm{app}(\mathtt{flat}(u), \sigma_2) = \mathtt{aflat}(u, \sigma_2)$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathrm{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2) \quad [\mathit{conj.}]$$

III. unify generating term with each of the recursive calls

$\mathcal{R}_{\mathtt{flat(leaf)}} \cup \mathcal{R}_{\mathtt{flat(node}(x,y,z))} = \{\mathtt{flat}(x),\ \mathtt{flat}(z)\}$

$$\mathrm{mgu}(\mathtt{flat}(u), \mathtt{flat}(x)) = \{u \mapsto x\}$$

$$\mathrm{mgu}(\mathtt{flat}(u), \mathtt{flat}(z)) = \{u \mapsto z\}$$

$$\left(\begin{array}{c} \mathrm{app}(\mathtt{flat}(\mathtt{leaf}), \sigma_2) = \mathtt{aflat}(\mathtt{leaf}, \sigma_2) \ \wedge \\ \\ \mathrm{app}(\mathtt{flat}(\mathtt{node}(x,y,z)), \sigma_2) = \mathtt{aflat}(\mathtt{node}(x,y,z), \sigma_2) \end{array}\right) \rightarrow \mathrm{app}(\mathtt{flat}(u), \sigma_2) = \mathtt{aflat}(u, \sigma_2)$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathrm{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2) \quad [conj.]$$

III. unify generating term with each of the recursive calls
$\mathcal{R}_{\mathtt{flat(leaf)}} \cup \mathcal{R}_{\mathtt{flat(node}(x,y,z))} = \{\mathtt{flat}(x),\ \mathtt{flat}(z)\}$

$$\mathtt{mgu}(\mathtt{flat}(u), \mathtt{flat}(x)) = \{u \mapsto x\}$$
$$\mathtt{mgu}(\mathtt{flat}(u), \mathtt{flat}(z)) = \{u \mapsto z\}$$

$$\begin{pmatrix} \mathrm{app}(\mathtt{flat}(\mathtt{leaf}), \sigma_2) = \mathtt{aflat}(\mathtt{leaf}, \sigma_2) \ \wedge \\ (\mathrm{app}(\mathtt{flat}(x), \sigma_2) = \mathtt{aflat}(x, \sigma_2) \wedge \\ \mathrm{app}(\mathtt{flat}(z), \sigma_2) = \mathtt{aflat}(z, \sigma_2) \ \rightarrow \\ \mathrm{app}(\mathtt{flat}(\mathtt{node}(x, y, z)), \sigma_2) = \mathtt{aflat}(\mathtt{node}(x, y, z), \sigma_2)) \end{pmatrix} \rightarrow \mathrm{app}(\mathtt{flat}(u), \sigma_2) = \mathtt{aflat}(u, \sigma_2)$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathrm{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2) \quad [\mathit{conj}.]$$

IV. universally quantify formula

$$(\forall) \begin{pmatrix} \mathrm{app}(\mathtt{flat}(\mathtt{leaf}), \sigma_2) = \mathtt{aflat}(\mathtt{leaf}, \sigma_2) \ \wedge \\ (\mathrm{app}(\mathtt{flat}(x), \sigma_2) = \mathtt{aflat}(x, \sigma_2) \wedge \\ \mathrm{app}(\mathtt{flat}(z), \sigma_2) = \mathtt{aflat}(z, \sigma_2) \ \rightarrow \\ \mathrm{app}(\mathtt{flat}(\mathrm{node}(x, y, z)), \sigma_2) = \mathtt{aflat}(\mathrm{node}(x, y, z), \sigma_2)) \end{pmatrix} \rightarrow \mathrm{app}(\mathtt{flat}(u), \sigma_2) = \mathtt{aflat}(u, \sigma_2)$$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
   $\vee\ \texttt{app}(\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2)$
   $\neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [BR]

6. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
   $\vee\ \texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$ [BR]

7. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
   $\vee\ \texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$ [BR]

Function definition axioms:

$\texttt{app}(\texttt{nil}, z) = z$

$\texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$

$\texttt{flat}(\texttt{leaf}) = \texttt{nil}$

$\texttt{flat}(\texttt{node}(x, y, z)) =$
$\quad \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$

$\texttt{aflat}(\texttt{leaf}, u) = u$

$\texttt{aflat}(\texttt{node}(x, y, z), u) =$
$\quad \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
$\vee\ \texttt{app}(\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2)$
$\neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [BR]

6. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
$\vee\ \texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$ [BR]

7. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
$\vee\ \texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$ [BR]

Function definition axioms:

$\texttt{app}(\texttt{nil}, z) = z$

$\texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$

$\texttt{flat}(\texttt{leaf}) = \texttt{nil}$

$\texttt{flat}(\texttt{node}(x, y, z)) =$
$\texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$

$\texttt{aflat}(\texttt{leaf}, u) = u$

$\texttt{aflat}(\texttt{node}(x, y, z), u) =$
$\texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{nil}, \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
   $\vee\ \texttt{app}(\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2)$
   $\neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [BR]

6. $\texttt{app}(\texttt{nil}, \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
   $\vee\ \texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$ [BR]

7. $\texttt{app}(\texttt{nil}, \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
   $\vee\ \texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$ [BR]

Function definition axioms:

$\texttt{app}(\texttt{nil}, z) = z$

$\texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$

$\texttt{flat}(\texttt{leaf}) = \texttt{nil}$

$\texttt{flat}(\texttt{node}(x, y, z)) =$
$\quad \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$

$\texttt{aflat}(\texttt{leaf}, u) = u$

$\texttt{aflat}(\texttt{node}(x, y, z), u) =$
$\quad \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{nil}, \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
$\vee\ \texttt{app}(\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2)$
$\neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [BR]

6. $\texttt{app}(\texttt{nil}, \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
$\vee\ \texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$ [BR]

7. $\texttt{app}(\texttt{nil}, \sigma_2) \neq \texttt{aflat}(\texttt{leaf}, \sigma_2)$
$\vee\ \texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$ [BR]

Function definition axioms:

$\texttt{app}(\texttt{nil}, z) = z$

$\texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$

$\texttt{flat}(\texttt{leaf}) = \texttt{nil}$

$\texttt{flat}(\texttt{node}(x, y, z)) =$
$\quad \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$

$\texttt{aflat}(\texttt{leaf}, u) = u$

$\texttt{aflat}(\texttt{node}(x, y, z), u) =$
$\quad \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{nil}, \sigma_2) \neq \sigma_2$
   $\vee\ \texttt{app}(\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2)$
   $\neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [BR]

6. $\texttt{app}(\texttt{nil}, \sigma_2) \neq \sigma_2$
   $\vee\ \texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$ [BR]

7. $\texttt{app}(\texttt{nil}, \sigma_2) \neq \sigma_2$
   $\vee\ \texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$ [BR]

Function definition axioms:

$\texttt{app}(\texttt{nil}, z) = z$

$\texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$

$\texttt{flat}(\texttt{leaf}) = \texttt{nil}$

$\texttt{flat}(\texttt{node}(x, y, z)) =$
$\quad \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$

$\texttt{aflat}(\texttt{leaf}, u) = u$

$\texttt{aflat}(\texttt{node}(x, y, z), u) =$
$\quad \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

---

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{nil}, \sigma_2) \neq \sigma_2$
   $\vee\ \texttt{app}(\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2)$
   $\neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [BR]

6. $\texttt{app}(\texttt{nil}, \sigma_2) \neq \sigma_2$
   $\vee\ \texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$ [BR]

7. $\texttt{app}(\texttt{nil}, \sigma_2) \neq \sigma_2$
   $\vee\ \texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$ [BR]

Function definition axioms:

$\texttt{app}(\texttt{nil}, z) = z$

$\texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$

$\texttt{flat}(\texttt{leaf}) = \texttt{nil}$

$\texttt{flat}(\texttt{node}(x, y, z)) =$
$\quad \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$

$\texttt{aflat}(\texttt{leaf}, u) = u$

$\texttt{aflat}(\texttt{node}(x, y, z), u) =$
$\quad \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

---

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\sigma_2 \neq \sigma_2 \lor \texttt{app}(\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2) \neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [BR]

6. $\sigma_2 \neq \sigma_2 \lor \texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$ [BR]

7. $\sigma_2 \neq \sigma_2 \lor \texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$ [BR]

Function definition axioms:

$\texttt{app}(\texttt{nil}, z) = z$

$\texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$

$\texttt{flat}(\texttt{leaf}) = \texttt{nil}$

$\texttt{flat}(\texttt{node}(x, y, z)) = \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$

$\texttt{aflat}(\texttt{leaf}, u) = u$

$\texttt{aflat}(\texttt{node}(x, y, z), u) = \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1.	$\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$	[conj.]
5.	$\sigma_2 \neq \sigma_2 \vee \texttt{app}(\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2) \neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$	[BR]
6.	$\sigma_2 \neq \sigma_2 \vee \texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$	[BR]
7.	$\sigma_2 \neq \sigma_2 \vee \texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$	[BR]

Function definition axioms:

$$\texttt{app}(\texttt{nil}, z) = z$$
$$\texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$$

$$\texttt{flat}(\texttt{leaf}) = \texttt{nil}$$
$$\texttt{flat}(\texttt{node}(x, y, z)) = \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$$

$$\texttt{aflat}(\texttt{leaf}, u) = u$$
$$\texttt{aflat}(\texttt{node}(x, y, z), u) = \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $$\texttt{app}(\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2) \neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$$ [BR]

6. $$\texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$$ [BR]

7. $$\texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$$ [BR]

Function definition axioms:

$$\texttt{app}(\texttt{nil}, z) = z$$
$$\texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$$
$$\texttt{flat}(\texttt{leaf}) = \texttt{nil}$$
$$\texttt{flat}(\texttt{node}(x, y, z)) = \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$$
$$\texttt{aflat}(\texttt{leaf}, u) = u$$
$$\texttt{aflat}(\texttt{node}(x, y, z), u) = \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \mathtt{btree}.\ \mathtt{app}(\mathtt{flat}(x), y) = \mathtt{aflat}(x, y)$

1. $\mathtt{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\mathtt{app}(\mathtt{flat}(\mathtt{node}(\sigma_3, \sigma_4, \sigma_5)), \sigma_2) \neq \mathtt{aflat}(\mathtt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [BR]

6. $\mathtt{app}(\mathtt{flat}(\sigma_3), \sigma_2) = \mathtt{aflat}(\sigma_3, \sigma_2)$ [BR]

7. $\mathtt{app}(\mathtt{flat}(\sigma_5), \sigma_2) = \mathtt{aflat}(\sigma_5, \sigma_2)$ [BR]

Function definition axioms:

$\mathtt{app}(\mathtt{nil}, z) = z$

$\mathtt{app}(\mathtt{cons}(x, y), z) = \mathtt{cons}(x, \mathtt{app}(y, z))$

$\mathtt{flat}(\mathtt{leaf}) = \mathtt{nil}$

$\mathtt{flat}(\mathtt{node}(x, y, z)) = \mathtt{app}(\mathtt{flat}(x), \mathtt{cons}(y, \mathtt{flat}(z)))$

$\mathtt{aflat}(\mathtt{leaf}, u) = u$

$\mathtt{aflat}(\mathtt{node}(x, y, z), u) = \mathtt{aflat}(x, \mathtt{cons}(y, \mathtt{aflat}(z, u)))$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \mathtt{btree}.\ \mathtt{app}(\mathtt{flat}(x), y) = \mathtt{aflat}(x, y)$

1. $\mathtt{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\mathtt{app}(\mathtt{app}(\mathtt{flat}(\sigma_3), \mathtt{cons}(\sigma_4, \mathtt{flat}(\sigma_5))), \sigma_2) \neq \mathtt{aflat}(\mathtt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [BR]

6. $\mathtt{app}(\mathtt{flat}(\sigma_3), \sigma_2) = \mathtt{aflat}(\sigma_3, \sigma_2)$ [BR]

7. $\mathtt{app}(\mathtt{flat}(\sigma_5), \sigma_2) = \mathtt{aflat}(\sigma_5, \sigma_2)$ [BR]

Function definition axioms:

$$\mathtt{app}(\mathtt{nil}, z) = z$$
$$\mathtt{app}(\mathtt{cons}(x, y), z) = \mathtt{cons}(x, \mathtt{app}(y, z))$$
$$\mathtt{flat}(\mathtt{leaf}) = \mathtt{nil}$$
$$\mathtt{flat}(\mathtt{node}(x, y, z)) = \mathtt{app}(\mathtt{flat}(x), \mathtt{cons}(y, \mathtt{flat}(z)))$$
$$\mathtt{aflat}(\mathtt{leaf}, u) = u$$
$$\mathtt{aflat}(\mathtt{node}(x, y, z), u) = \mathtt{aflat}(x, \mathtt{cons}(y, \mathtt{aflat}(z, u)))$$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{app}(\texttt{flat}(\sigma_3), \texttt{cons}(\sigma_4, \texttt{flat}(\sigma_5))), \sigma_2) \neq \texttt{aflat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5), \sigma_2)$ [BR]

6. $\texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$ [BR]

7. $\texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$ [BR]

Function definition axioms:

$\texttt{app}(\texttt{nil}, z) = z$

$\texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$

$\texttt{flat}(\texttt{leaf}) = \texttt{nil}$

$\texttt{flat}(\texttt{node}(x, y, z)) = \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$

$\texttt{aflat}(\texttt{leaf}, u) = u$

$\texttt{aflat}(\texttt{node}(x, y, z), u) = \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \mathtt{btree}.\ \mathtt{app}(\mathtt{flat}(x), y) = \mathtt{aflat}(x, y)$

1. $\mathtt{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\mathtt{app}(\mathtt{app}(\mathtt{flat}(\sigma_3), \mathtt{cons}(\sigma_4, \mathtt{flat}(\sigma_5))), \sigma_2) \neq \mathtt{aflat}(\sigma_3, \mathtt{cons}(\sigma_4, \mathtt{aflat}(\sigma_5, \sigma_2)))$ [BR]

6. $\mathtt{app}(\mathtt{flat}(\sigma_3), \sigma_2) = \mathtt{aflat}(\sigma_3, \sigma_2)$ [BR]

7. $\mathtt{app}(\mathtt{flat}(\sigma_5), \sigma_2) = \mathtt{aflat}(\sigma_5, \sigma_2)$ [BR]

Function definition axioms:

$$\mathtt{app}(\mathtt{nil}, z) = z$$
$$\mathtt{app}(\mathtt{cons}(x, y), z) = \mathtt{cons}(x, \mathtt{app}(y, z))$$
$$\mathtt{flat}(\mathtt{leaf}) = \mathtt{nil}$$
$$\mathtt{flat}(\mathtt{node}(x, y, z)) = \mathtt{app}(\mathtt{flat}(x), \mathtt{cons}(y, \mathtt{flat}(z)))$$
$$\mathtt{aflat}(\mathtt{leaf}, u) = u$$
$$\mathtt{aflat}(\mathtt{node}(x, y, z), u) = \mathtt{aflat}(x, \mathtt{cons}(y, \mathtt{aflat}(z, u)))$$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5.
$$\texttt{app}(\texttt{app}(\texttt{flat}(\sigma_3), \texttt{cons}(\sigma_4, \texttt{flat}(\sigma_5))), \sigma_2) \neq \texttt{aflat}(\sigma_3, \texttt{cons}(\sigma_4, \texttt{aflat}(\sigma_5, \sigma_2)))$$ [BR]

6.
$$\texttt{app}(\texttt{flat}(\sigma_3), \sigma_2) = \texttt{aflat}(\sigma_3, \sigma_2)$$ [BR]

7.
$$\texttt{app}(\texttt{flat}(\sigma_5), \sigma_2) = \texttt{aflat}(\sigma_5, \sigma_2)$$ [BR]

Function definition axioms:

$$\texttt{app}(\texttt{nil}, z) = z$$
$$\texttt{app}(\texttt{cons}(x, y), z) = \texttt{cons}(x, \texttt{app}(y, z))$$

$$\texttt{flat}(\texttt{leaf}) = \texttt{nil}$$
$$\texttt{flat}(\texttt{node}(x, y, z)) = \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z)))$$

$$\texttt{aflat}(\texttt{leaf}, u) = u$$
$$\texttt{aflat}(\texttt{node}(x, y, z), u) = \texttt{aflat}(x, \texttt{cons}(y, \texttt{aflat}(z, u)))$$

Challenge: use induction hypotheses to get rid of aflat or flat in 5.

Generating induction formulas - motivating example I. ctd.

1. $\mathtt{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2)$ [*conj*.]

Generating induction formulas - motivating example I. ctd.

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [*conj*.]

$$(\forall)\left(\vphantom{\begin{array}{c}\\\\\\\\\end{array}}\qquad\qquad\qquad\qquad\qquad\qquad\right) \rightarrow$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathtt{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2) \quad [conj.]$$

I. create generating term: replace arguments in $\mathtt{aflat}(\sigma_1, \sigma_2)$ in argument positions $I_{\mathtt{aflat}} = \{1, 2\}$ with fresh variables

$$\mathtt{aflat}(x_0, y_0)$$

$$(\forall)\left(\qquad \right) \rightarrow \mathtt{app}(\mathtt{flat}(x_0), y_0) = \mathtt{aflat}(x_0, y_0)$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathtt{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2) \quad [conj.]$$

II. unify generating term with each of the branches
$\mathcal{B}_{\mathtt{aflat}} = \{\mathtt{aflat}(\mathtt{leaf}, u_0),\ \mathtt{aflat}(\mathtt{node}(x, y, z), u_1)\}$

$$\mathtt{mgu}(\mathtt{aflat}(x_0, y_0), \mathtt{aflat}(\mathtt{leaf}, u_0)) = \{x_0 \mapsto \mathtt{leaf}, y_0 \mapsto u_0\}$$
$$\mathtt{mgu}(\mathtt{aflat}(x_0, y_0), \mathtt{aflat}(\mathtt{node}(x, y, z), u_1)) = \{x_0 \mapsto \mathtt{node}(x, y, z), y_0 \mapsto u_1\}$$

$$(\forall) \left(\begin{array}{c} \mathtt{app}(\mathtt{flat}(\mathtt{leaf}), u_0) = \mathtt{aflat}(\mathtt{leaf}, u_0) \land \\ \\ \mathtt{app}(\mathtt{flat}(\mathtt{node}(x, y, z)), u_1) = \mathtt{aflat}(\mathtt{node}(x, y, z), u_1) \end{array} \right) \to \mathtt{app}(\mathtt{flat}(x_0), y_0) = \mathtt{aflat}(x_0, y_0)$$

Generating induction formulas - motivating example I. ctd.

$$1. \quad \mathtt{app}(\mathtt{flat}(\sigma_1), \sigma_2) \neq \mathtt{aflat}(\sigma_1, \sigma_2) \quad [conj.]$$

III. unify generating term with each of the recursive calls

$$\mathcal{R}_{\mathtt{aflat}(\mathtt{leaf}, u_0)} \cup \mathcal{R}_{\mathtt{aflat}(\mathtt{node}(x,y,z), u_1)} = \left\{ \begin{array}{c} \mathtt{aflat}(x, \mathtt{cons}(y, \mathtt{aflat}(z, u_1))), \\ \mathtt{aflat}(z, u_1) \end{array} \right\}$$

$$\mathtt{mgu}(\mathtt{aflat}(x_0, y_0), \mathtt{aflat}(x, \mathtt{cons}(y, \mathtt{aflat}(z, u_1)))) = \left\{ \begin{array}{c} x_0 \mapsto x, \\ y_0 \mapsto \mathtt{cons}(y, \mathtt{aflat}(z, u_1)) \end{array} \right\}$$

$$\mathtt{mgu}(\mathtt{aflat}(x_0, y_0), \mathtt{aflat}(z, u_1)) = \{x_0 \mapsto z, y_0 \mapsto u_1\}$$

$$(\forall) \left(\begin{array}{c} \mathtt{app}(\mathtt{flat}(\mathtt{leaf}), u_0) = \mathtt{aflat}(\mathtt{leaf}, u_0) \wedge \\ (\mathtt{app}(\mathtt{flat}(x), \mathtt{cons}(y, \mathtt{aflat}(z, u_1))) = \\ \mathtt{aflat}(x, \mathtt{cons}(y, \mathtt{aflat}(z, u_1))) \wedge \\ \mathtt{app}(\mathtt{flat}(z), u_1) = \mathtt{aflat}(z, u_1) \rightarrow \\ \mathtt{app}(\mathtt{flat}(\mathtt{node}(x, y, z)), u_1) = \mathtt{aflat}(\mathtt{node}(x, y, z), u_1)) \end{array} \right) \rightarrow \mathtt{app}(\mathtt{flat}(x_0), y_0) = \mathtt{aflat}(x_0, y_0)$$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_6) \neq \texttt{aflat}(\texttt{leaf}, \sigma_6)$
   $\vee\ \texttt{app}(\texttt{flat}(\texttt{node}(\sigma_7, \sigma_8, \sigma_9)), \sigma_{10}) \neq \texttt{aflat}(\texttt{node}(\sigma_7, \sigma_8, \sigma_9), \sigma_{10})$ [Ind 1]
   $\vee\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

6. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_6) \neq \texttt{aflat}(\texttt{leaf}, \sigma_6)$
   $\vee\ \texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10}))) = \texttt{aflat}(\sigma_7, \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$ [Ind 1]
   $\vee\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

7. $\texttt{app}(\texttt{flat}(\texttt{leaf}), \sigma_6) \neq \texttt{aflat}(\texttt{leaf}, \sigma_6)$
   $\vee\ \texttt{app}(\texttt{flat}(\sigma_9), \sigma_{10}) = \texttt{aflat}(\sigma_9, \sigma_{10})$ [Ind 1]
   $\vee\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{flat}(\sigma_9))), \sigma_{10}) \neq \texttt{aflat}(\sigma_7, \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$ [BR]

6. $\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10}))) = \texttt{aflat}(\sigma_7, \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$ [BR]

7. $\texttt{app}(\texttt{flat}(\sigma_9), \sigma_{10}) = \texttt{aflat}(\sigma_9, \sigma_{10})$ [BR]

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

---

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{flat}(\sigma_9))), \sigma_{10}) \neq \texttt{aflat}(\sigma_7, \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$ [BR]

6. $\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10}))) = \texttt{aflat}(\sigma_7, \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$ [BR]

7. $\texttt{app}(\texttt{flat}(\sigma_9), \sigma_{10}) = \texttt{aflat}(\sigma_9, \sigma_{10})$ [BR]

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{flat}(\sigma_9))), \sigma_{10}) \neq \texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$ [BR]

6. $\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10}))) = \texttt{aflat}(\sigma_7, \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$ [BR]

7. $\texttt{app}(\texttt{flat}(\sigma_9), \sigma_{10}) = \texttt{aflat}(\sigma_9, \sigma_{10})$ [BR]

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{flat}(\sigma_9))), \sigma_{10}) \neq \texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$ [BR]

6. $\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10}))) = \texttt{aflat}(\sigma_7, \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$ [BR]

7. $\texttt{app}(\texttt{flat}(\sigma_9), \sigma_{10}) = \texttt{aflat}(\sigma_9, \sigma_{10})$ [BR]

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{flat}(\sigma_9))), \sigma_{10}) \neq \texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{app}(\texttt{flat}(\sigma_9), \sigma_{10})))$ [BR]

6. $\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10}))) = \texttt{aflat}(\sigma_7, \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$ [BR]

7. $\texttt{app}(\texttt{flat}(\sigma_9), \sigma_{10}) = \texttt{aflat}(\sigma_9, \sigma_{10})$ [BR]

Generating induction formulas - motivating example I. ctd.

Prove: $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$

1. $\texttt{app}(\texttt{flat}(\sigma_1), \sigma_2) \neq \texttt{aflat}(\sigma_1, \sigma_2)$ [conj.]

5. $\texttt{app}(\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{flat}(\sigma_9))), \sigma_{10}) \neq \texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{app}(\texttt{flat}(\sigma_9), \sigma_{10})))$ [BR]

6. $\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10}))) = \texttt{aflat}(\sigma_7, \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$ [BR]

7. $\texttt{app}(\texttt{flat}(\sigma_9), \sigma_{10}) = \texttt{aflat}(\sigma_9, \sigma_{10})$ [BR]

Clause 5 can be refuted with on more induction on $\texttt{flat}(\sigma_7)$

Function definition rewriting

$$\frac{\texttt{flat}(\texttt{node}(x, y, z)) = \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z))) \qquad \texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)) \neq t \lor D}{\texttt{app}(\texttt{flat}(\sigma_3), \texttt{cons}(\sigma_4, \texttt{flat}(\sigma_5))) \neq t \lor D}$$

Function definition rewriting

$$\frac{\texttt{flat}(\texttt{node}(x, y, z)) = \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z))) \qquad \texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)) \neq t \vee D}{\texttt{app}(\texttt{flat}(\sigma_3), \texttt{cons}(\sigma_4, \texttt{flat}(\sigma_5))) \neq t \vee D}$$

does not happen if $\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)) \prec \texttt{app}(\texttt{flat}(\sigma_3), \texttt{cons}(\sigma_4, \texttt{flat}(\sigma_5)))$

Function definition rewriting

$$\frac{\texttt{flat}(\texttt{node}(x, y, z)) \doteq \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z))) \qquad \texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)) \neq t \lor D}{\texttt{app}(\texttt{flat}(\sigma_3), \texttt{cons}(\sigma_4, \texttt{flat}(\sigma_5))) \neq t \lor D}$$

does not happen if $\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)) \prec \texttt{app}(\texttt{flat}(\sigma_3), \texttt{cons}(\sigma_4, \texttt{flat}(\sigma_5)))$

$$\frac{f(\overline{s}) \doteq t \lor C \qquad L[f(\overline{s})\theta] \lor D}{L[t\theta] \lor C\theta \lor D}\ (\texttt{ParF})$$

Function definition rewriting

$$\frac{\texttt{flat}(\texttt{node}(x, y, z)) \doteq \texttt{app}(\texttt{flat}(x), \texttt{cons}(y, \texttt{flat}(z))) \qquad \texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)) \neq t \vee D}{\texttt{app}(\texttt{flat}(\sigma_3), \texttt{cons}(\sigma_4, \texttt{flat}(\sigma_5))) \neq t \vee D}$$

does not happen if $\texttt{flat}(\texttt{node}(\sigma_3, \sigma_4, \sigma_5)) \prec \texttt{app}(\texttt{flat}(\sigma_3), \texttt{cons}(\sigma_4, \texttt{flat}(\sigma_5)))$

$$\frac{f(\overline{s}) \doteq t \vee C \qquad L[f(\overline{s})\theta] \vee D}{L[t\theta] \vee C\theta \vee D} \; (\texttt{ParF})$$

$$\frac{f(\overline{s}) \doteq t \qquad \cancel{L[f(\overline{s})\theta] \vee D}}{L[t\theta] \vee D} \; (\texttt{DemF})$$

- $f(\overline{s})\theta \succ t\theta$
- $L[f(\overline{s})\theta] \vee D \succ f(\overline{s})\theta = t\theta$

Rewriting with induction hypotheses

Conclusion:

$$\texttt{app}(\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{flat}(\sigma_9))), \sigma_{10}) \neq \texttt{aflat}(\sigma_7, \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$$

Hypothesis:

$$\texttt{app}(\texttt{flat}(\sigma_7), \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10}))) = \texttt{aflat}(\sigma_7, \texttt{cons}(\sigma_8, \texttt{aflat}(\sigma_9, \sigma_{10})))$$

Rewriting with induction hypotheses

Conclusion:

$$\mathtt{app}(\mathtt{app}(\mathtt{flat}(\sigma_7), \mathtt{cons}(\sigma_8, \mathtt{flat}(\sigma_9))), \sigma_{10}) \neq \mathtt{aflat}(\sigma_7, \mathtt{cons}(\sigma_8, \mathtt{aflat}(\sigma_9, \sigma_{10})))$$

Hypothesis:

$$\mathtt{app}(\mathtt{flat}(\sigma_7), \mathtt{cons}(\sigma_8, \mathtt{aflat}(\sigma_9, \sigma_{10}))) = \mathtt{aflat}(\sigma_7, \mathtt{cons}(\sigma_8, \mathtt{aflat}(\sigma_9, \sigma_{10})))$$

$$\frac{l = r \vee D \quad s[l] \neq t \vee C}{\mathtt{cnf}(F \rightarrow \forall \overline{y}.(s[r] = t)[\overline{y}])} \; (\mathtt{IndHRW})$$

- $s \neq t$ and $l = r$ are induction conclusion and hypothesis literals
- $l \not\succeq r$
- $F \rightarrow \forall \overline{y}.(s[r] = t)[\overline{y}]$ is a valid induction formula

Motivating example I. – conclusion

We proved $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$ with:

Motivating example I. – conclusion

We proved $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$ with:

- non-trivial induction formulas with strengthened hypotheses

Motivating example I. – conclusion

We proved $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$ with:

- non-trivial induction formulas with strengthened hypotheses
- rewriting by function definitions without ordering constraints

Motivating example I. – conclusion

We proved $\forall x, y \in \texttt{btree}.\ \texttt{app}(\texttt{flat}(x), y) = \texttt{aflat}(x, y)$ with:

- non-trivial induction formulas with strengthened hypotheses
- rewriting by function definitions without ordering constraints
- special treatment of equational induction hypotheses

Motivating example II.

Prove: $\forall x, y.\texttt{even}(y) \rightarrow \texttt{even}(\texttt{mult}(x, y))$

Motivating example II.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
2.	$\neg\mathtt{even}(\mathtt{mult}(\sigma_1, \sigma_2))$	[conj.]

Motivating example II.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1. $\mathtt{even}(\sigma_2)$ [conj.]
2. $\neg\mathtt{even}(\mathtt{mult}(\sigma_1, \sigma_2))$ [conj.]

$$\frac{\overline{L}[\overline{t}] \vee C}{\mathtt{cnf}(F \rightarrow \forall \overline{y}.L[\overline{y}])}\ (\mathtt{Ind})$$

- L is ground
- $F \rightarrow \forall \overline{y}.L[\overline{y}]$ is valid induction formula

Motivating example II.

Prove: $\forall x, y.\texttt{even}(y) \rightarrow \texttt{even}(\texttt{mult}(x, y))$

1. $\texttt{even}(\sigma_2)$ [conj.]
2. $\neg\texttt{even}(\texttt{mult}(\sigma_1, \sigma_2))$ [conj.]

$$\frac{\overline{L}[\overline{t}] \vee C}{\texttt{cnf}(F \rightarrow \forall \overline{y}.L[\overline{y}])}\ (\texttt{Ind})$$

- L is ground
- $F \rightarrow \forall \overline{y}.L[\overline{y}]$ is valid induction formula

$$\begin{pmatrix} L[0] \wedge \\ \forall x.\ (L[x] \rightarrow L[\mathtt{s}(x)]) \end{pmatrix} \rightarrow \forall z.L[z]$$

Motivating example II.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

---

1. $\mathtt{even}(\sigma_2)$ [conj.]
2. $\neg\mathtt{even}(\mathtt{mult}(\sigma_1, \sigma_2))$ [conj.]
3. $\neg\mathtt{even}(\mathtt{mult}(0, \sigma_2)) \lor \mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2)) \lor \mathtt{even}(\mathtt{mult}(x, \sigma_2))$ [Ind 2]
4. $\neg\mathtt{even}(\mathtt{mult}(0, \sigma_2)) \lor \neg\mathtt{even}(\mathtt{mult}(\mathtt{s}(\sigma_3), \sigma_2)) \lor \mathtt{even}(\mathtt{mult}(x, \sigma_2))$ [Ind 2]

$$\frac{\overline{L}[\overline{t}] \lor C}{\mathtt{cnf}(F \rightarrow \forall \overline{y}.L[\overline{y}])} \ (\mathtt{Ind})$$

- L is ground
- $F \rightarrow \forall \overline{y}.L[\overline{y}]$ is valid induction formula

$$\begin{pmatrix} L[0] \land \\ \forall x.\ (L[x] \rightarrow L[\mathtt{s}(x)]) \end{pmatrix} \rightarrow \forall z.L[z]$$

Motivating example II.

Prove: $\forall x, y.\texttt{even}(y) \rightarrow \texttt{even}(\texttt{mult}(x, y))$

---

1. $\texttt{even}(\sigma_2)$ [conj.]
2. $\neg\texttt{even}(\texttt{mult}(\sigma_1, \sigma_2))$ [conj.]
3. $\neg\texttt{even}(\texttt{mult}(0, \sigma_2))$ $\vee\ \texttt{even}(\texttt{mult}(\sigma_3, \sigma_2))$ $\vee\ \texttt{even}(\texttt{mult}(x, \sigma_2))$ [Ind 2]
4. $\neg\texttt{even}(\texttt{mult}(0, \sigma_2))$ $\vee\ \neg\texttt{even}(\texttt{mult}(\texttt{s}(\sigma_3), \sigma_2))$ $\vee\ \texttt{even}(\texttt{mult}(x, \sigma_2))$ [Ind 2]

$$\frac{\overline{L}[\overline{t}] \vee C}{\texttt{cnf}(F \rightarrow \forall \overline{y}.L[\overline{y}])}\ (\texttt{Ind})$$

- L is ground
- $F \rightarrow \forall \overline{y}.L[\overline{y}]$ is valid induction formula

$$\begin{pmatrix} L[0]\wedge \\ \forall x.\ (L[x] \rightarrow L[\texttt{s}(x)]) \end{pmatrix} \rightarrow \forall z.L[z]$$

Motivating example II.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1. $\mathtt{even}(\sigma_2)$ [conj.]
2. $\neg\mathtt{even}(\mathtt{mult}(\sigma_1, \sigma_2))$ [conj.]
5. $\neg\mathtt{even}(\mathtt{mult}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]
6. $\neg\mathtt{even}(\mathtt{mult}(0, \sigma_2)) \vee \neg\mathtt{even}(\mathtt{mult}(\mathtt{s}(\sigma_3), \sigma_2))$ [BR]

$$\frac{\overline{L}[\overline{t}] \vee C}{\mathtt{cnf}(F \rightarrow \forall \overline{y}.L[\overline{y}])}\ (\mathtt{Ind})$$

- L is ground
- $F \rightarrow \forall \overline{y}.L[\overline{y}]$ is valid induction formula

$$\left(\begin{array}{c} L[0]\wedge \\ \forall x.\ (L[x] \rightarrow L[\mathtt{s}(x)]) \end{array}\right) \rightarrow \forall z.L[z]$$

Motivating example II.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
2.	$\neg\mathtt{even}(\mathtt{mult}(\sigma_1, \sigma_2))$	[conj.]
5.	$\neg\mathtt{even}(\mathtt{mult}(0, \sigma_2))$ $\vee\ \mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{mult}(0, \sigma_2))$ $\vee\ \neg\mathtt{even}(\mathtt{mult}(\mathtt{s}(\sigma_3), \sigma_2))$	[BR]

Function definition axioms:

$\mathtt{add}(0, y) = y$

$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$

$\mathtt{mult}(0, y) = 0$

$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$

$\mathtt{even}(0)$

$\neg\mathtt{even}(\mathtt{s}(0))$

$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$

$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$

Motivating example II.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1. $\mathtt{even}(\sigma_2)$ [conj.]
2. $\neg\mathtt{even}(\mathtt{mult}(\sigma_1, \sigma_2))$ [conj.]
5. $\neg\mathtt{even}(0) \vee \mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]
6. $\neg\mathtt{even}(0) \vee \neg\mathtt{even}(\mathtt{mult}(\mathtt{s}(\sigma_3), \sigma_2))$ [BR]

Function definition axioms:

$\mathtt{add}(0, y) = y$

$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$

$\mathtt{mult}(0, y) = 0$

$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$

$\mathtt{even}(0)$

$\neg\mathtt{even}(\mathtt{s}(0))$

$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$

$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$

Motivating example II.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
2.	$\neg\mathtt{even}(\mathtt{mult}(\sigma_1, \sigma_2))$	[conj.]
5.	$\neg\mathtt{even}(0) \lor \mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(0) \lor \neg\mathtt{even}(\mathtt{mult}(\mathtt{s}(\sigma_3), \sigma_2))$	[BR]

Function definition axioms:

$\mathtt{add}(0, y) = y$

$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$

$\mathtt{mult}(0, y) = 0$

$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$

$\mathtt{even}(0)$

$\neg\mathtt{even}(\mathtt{s}(0))$

$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \lor \neg\mathtt{even}(x)$

$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \lor \mathtt{even}(x)$

Motivating example II.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
2.	$\neg\mathtt{even}(\mathtt{mult}(\sigma_1, \sigma_2))$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{mult}(\mathtt{s}(\sigma_3), \sigma_2))$	[BR]

Function definition axioms:

$\mathtt{add}(0, y) = y$

$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$

$\mathtt{mult}(0, y) = 0$

$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$

$\mathtt{even}(0)$

$\neg\mathtt{even}(\mathtt{s}(0))$

$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$

$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$

Motivating example II.

Prove: $\forall x, y. \texttt{even}(y) \rightarrow \texttt{even}(\texttt{mult}(x, y))$

1.	$\texttt{even}(\sigma_2)$	[conj.]
2.	$\neg\texttt{even}(\texttt{mult}(\sigma_1, \sigma_2))$	[conj.]
5.	$\texttt{even}(\texttt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\texttt{even}(\texttt{mult}(\texttt{s}(\sigma_3), \sigma_2))$	[BR]

Function definition axioms:

$\texttt{add}(0, y) = y$

$\texttt{add}(\texttt{s}(x), y) = \texttt{s}(\texttt{add}(x, y))$

$\texttt{mult}(0, y) = 0$

$\texttt{mult}(\texttt{s}(x), y) = \texttt{add}(\texttt{mult}(x, y), y)$

$\texttt{even}(0)$

$\neg\texttt{even}(\texttt{s}(0))$

$\neg\texttt{even}(\texttt{s}(\texttt{s}(x))) \vee \neg\texttt{even}(x)$

$\texttt{even}(\texttt{s}(\texttt{s}(x))) \vee \texttt{even}(x)$

Motivating example II.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
2.	$\neg\mathtt{even}(\mathtt{mult}(\sigma_1, \sigma_2))$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]

Function definition axioms:

$\mathtt{add}(0, y) = y$

$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$

$\mathtt{mult}(0, y) = 0$

$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$

$\mathtt{even}(0)$

$\neg\mathtt{even}(\mathtt{s}(0))$

$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$

$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$

Motivating example II.

Prove: $\forall x, y.\texttt{even}(y) \rightarrow \texttt{even}(\texttt{mult}(x, y))$

1. $\texttt{even}(\sigma_2)$ [conj.]
2. $\neg\texttt{even}(\texttt{mult}(\sigma_1, \sigma_2))$ [conj.]
5. $\texttt{even}(\texttt{mult}(\sigma_3, \sigma_2))$ [BR]
6. $\neg\texttt{even}(\texttt{add}(\texttt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

Function definition axioms:

$\texttt{add}(0, y) = y$

$\texttt{add}(\texttt{s}(x), y) = \texttt{s}(\texttt{add}(x, y))$

$\texttt{mult}(0, y) = 0$

$\texttt{mult}(\texttt{s}(x), y) = \texttt{add}(\texttt{mult}(x, y), y)$

$\texttt{even}(0)$

$\neg\texttt{even}(\texttt{s}(0))$

$\neg\texttt{even}(\texttt{s}(\texttt{s}(x))) \vee \neg\texttt{even}(x)$

$\texttt{even}(\texttt{s}(\texttt{s}(x))) \vee \texttt{even}(x)$

Challenge: use induction hypothesis get rid of `mult` in 6.

Multi-clause induction

$$\frac{\overline{L}[\overline{t}] \vee C}{\mathtt{cnf}(F \rightarrow \forall \overline{y}.L[\overline{y}])}\ (\mathtt{Ind})$$

- L is ground
- $F \rightarrow \forall \overline{y}.L[\overline{y}]$ is valid induction formula

Multi-clause induction

$$\frac{\overline{L}[\overline{t}] \vee C}{\texttt{cnf}(F \rightarrow \forall \overline{y}.L[\overline{y}])}\ (\texttt{Ind})$$

- L is ground
- $F \rightarrow \forall \overline{y}.L[\overline{y}]$ is valid induction formula

---

$$\frac{L_1[\overline{t}] \vee C_1 \quad \dots \quad L_n[\overline{t}] \vee C_n \quad \overline{L}[\overline{t}] \vee C}{\texttt{cnf}(F \rightarrow \forall \overline{y}.(\bigwedge_{1 \leq i \leq n} L_i[\overline{y}] \rightarrow L[\overline{y}]))}\ (\texttt{IndMC})$$

- L and L_i for all $1 \leq i \leq n$ are ground
- $F \rightarrow \forall \overline{y}.(\bigwedge_{1 \leq i \leq n} L_i[\overline{y}] \rightarrow L[\overline{y}])$ is valid induction formula

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]

$$\frac{L_1[\overline{t}] \vee C_1 \quad \dots \quad L_n[\overline{t}] \vee C_n \quad \overline{L}[\overline{t}] \vee C}{\mathtt{cnf}(F \rightarrow \forall \overline{y}.(\bigwedge_{1 \leq i \leq n} L_i[\overline{y}] \rightarrow L[\overline{y}]))} \; (\mathtt{IndMC})$$

- L and L_i for all $1 \leq i \leq n$ are ground
- $F \rightarrow \forall \overline{y}.(\bigwedge_{1 \leq i \leq n} L_i[\overline{y}] \rightarrow L[\overline{y}])$ is valid induction formula

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]

$$\frac{L_1[\overline{t}] \vee C_1 \quad \ldots \quad L_n[\overline{t}] \vee C_n \quad \overline{L}[\overline{t}] \vee C}{\mathtt{cnf}(F \rightarrow \forall \overline{y}.(\bigwedge_{1 \leq i \leq n} L_i[\overline{y}] \rightarrow L[\overline{y}]))} \ (\mathtt{IndMC})$$

- L and L_i for all $1 \leq i \leq n$ are ground
- $F \rightarrow \forall \overline{y}.(\bigwedge_{1 \leq i \leq n} L_i[\overline{y}] \rightarrow L[\overline{y}])$ is valid induction formula

$$\left(\begin{array}{c} (\mathtt{even}(0) \rightarrow \mathtt{even}(\mathtt{add}(0, \sigma_2))) \wedge \\ (\mathtt{even}(\mathtt{s}(0)) \rightarrow \mathtt{even}(\mathtt{add}(\mathtt{s}(0), \sigma_2))) \wedge \\ \forall z. \left(\begin{array}{c} (\mathtt{even}(z) \rightarrow \mathtt{even}(\mathtt{add}(z, \sigma_2))) \rightarrow \\ (\mathtt{even}(\mathtt{s}(\mathtt{s}(z))) \rightarrow \mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(z)), \sigma_2))) \end{array} \right) \end{array} \right)$$

$$\rightarrow \forall x.(\mathtt{even}(x) \rightarrow \mathtt{even}(\mathtt{add}(x, \sigma_2)))$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\texttt{even}(y) \rightarrow \texttt{even}(\texttt{mult}(x, y))$

1.	$\texttt{even}(\sigma_2)$	[conj.]
5.	$\texttt{even}(\texttt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\texttt{even}(\texttt{add}(\texttt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]
	$\ldots$	
13.	$\neg\texttt{even}(\texttt{add}(0, \sigma_2)) \vee \texttt{even}(\texttt{s}(0))$ $\vee\ \texttt{even}(\texttt{s}(\texttt{s}(\sigma_4)))$ $\vee\ \neg\texttt{even}(x) \vee \texttt{even}(\texttt{add}(x, \sigma_2))$	[IndMC 5,6]
14.	$\neg\texttt{even}(\texttt{add}(0, \sigma_2)) \vee \texttt{even}(\texttt{s}(0))$ $\vee\ \neg\texttt{even}(\texttt{add}(\texttt{s}(\texttt{s}(\sigma_4)), \sigma_2))$ $\vee\ \neg\texttt{even}(x) \vee \texttt{even}(\texttt{add}(x, \sigma_2))$	[IndMC 5,6]
15.	$\neg\texttt{even}(\texttt{add}(0, \sigma_2)) \vee \texttt{even}(\texttt{s}(0))$ $\vee\ \neg\texttt{even}(\sigma_4) \vee \texttt{even}(\texttt{add}(\sigma_4, \sigma_2))$ $\vee\ \neg\texttt{even}(x) \vee \texttt{even}(\texttt{add}(x, \sigma_2))$	[IndMC 5,6]
	$\ldots$	

$$\frac{L_1[\overline{t}] \vee C_1 \quad \ldots \quad L_n[\overline{t}] \vee C_n \quad \overline{L}[\overline{t}] \vee C}{\texttt{cnf}(F \rightarrow \forall \overline{y}.(\bigwedge_{1 \le i \le n} L_i[\overline{y}] \rightarrow L[\overline{y}]))} \ (\texttt{IndMC})$$

- L and L_i for all $1 \le i \le n$ are ground
- $F \rightarrow \forall \overline{y}.(\bigwedge_{1 \le i \le n} L_i[\overline{y}] \rightarrow L[\overline{y}])$ is valid induction formula

$$\left(\begin{array}{c} (\texttt{even}(0) \rightarrow \texttt{even}(\texttt{add}(0, \sigma_2)))\wedge \\ (\texttt{even}(\texttt{s}(0)) \rightarrow \texttt{even}(\texttt{add}(\texttt{s}(0), \sigma_2)))\wedge \\ \forall z.\left(\begin{array}{c} (\texttt{even}(z) \rightarrow \texttt{even}(\texttt{add}(z, \sigma_2))) \rightarrow \\ (\texttt{even}(\texttt{s}(\texttt{s}(z))) \rightarrow \texttt{even}(\texttt{add}(\texttt{s}(\texttt{s}(z)), \sigma_2))) \end{array}\right) \end{array}\right)$$
$$\rightarrow \forall x.(\texttt{even}(x) \rightarrow \texttt{even}(\texttt{add}(x, \sigma_2)))$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

---

1. $\mathtt{even}(\sigma_2)$ [conj.]

5. $\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]

6. $\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

...

13. $\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0))$
$\vee\ \mathtt{even}(\mathtt{s}(\mathtt{s}(\sigma_4)))$ [IndMC 5,6]
$\vee\ \neg\mathtt{even}(x) \vee \mathtt{even}(\mathtt{add}(x, \sigma_2))$

14. $\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0))$
$\vee\ \neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$ [IndMC 5,6]
$\vee\ \neg\mathtt{even}(x) \vee \mathtt{even}(\mathtt{add}(x, \sigma_2))$

15. $\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0))$
$\vee\ \neg\mathtt{even}(\sigma_4) \vee \mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$ [IndMC 5,6]
$\vee\ \neg\mathtt{even}(x) \vee \mathtt{even}(\mathtt{add}(x, \sigma_2))$

...

$$\frac{L_1[\overline{t}] \vee C_1 \quad \ldots \quad L_n[\overline{t}] \vee C_n \quad \overline{L}[\overline{t}] \vee C}{\mathtt{cnf}(F \rightarrow \forall \overline{y}.(\bigwedge_{1 \le i \le n} L_i[\overline{y}] \rightarrow L[\overline{y}]))} \ (\mathtt{IndMC})$$

- L and L_i for all $1 \le i \le n$ are ground
- $F \rightarrow \forall \overline{y}.(\bigwedge_{1 \le i \le n} L_i[\overline{y}] \rightarrow L[\overline{y}])$ is valid induction formula

$$\left(\begin{array}{c} (\mathtt{even}(0) \rightarrow \mathtt{even}(\mathtt{add}(0, \sigma_2)))\wedge \\ (\mathtt{even}(\mathtt{s}(0)) \rightarrow \mathtt{even}(\mathtt{add}(\mathtt{s}(0), \sigma_2)))\wedge \\ \forall z.\left(\begin{array}{c} (\mathtt{even}(z) \rightarrow \mathtt{even}(\mathtt{add}(z, \sigma_2))) \rightarrow \\ (\mathtt{even}(\mathtt{s}(\mathtt{s}(z))) \rightarrow \mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(z)), \sigma_2))) \end{array}\right) \end{array}\right)$$
$$\rightarrow \forall x.(\mathtt{even}(x) \rightarrow \mathtt{even}(\mathtt{add}(x, \sigma_2)))$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

---

1. $\mathtt{even}(\sigma_2)$ [conj.]

5. $\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]

6. $\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

...

13. $\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0))$
    $\vee\ \mathtt{even}(\mathtt{s}(\mathtt{s}(\sigma_4)))$ [BR]
    $\vee\ \mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$

14. $\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0))$
    $\vee\ \neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$ [BR]
    $\vee\ \mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$

15. $\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0))$
    $\vee\ \neg\mathtt{even}(\sigma_4) \vee \mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$ [BR]
    $\vee\ \mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$

...

$$\frac{L_1[\bar{t}] \vee C_1 \quad \ldots \quad L_n[\bar{t}] \vee C_n \quad \overline{L}[\bar{t}] \vee C}{\mathtt{cnf}(F \rightarrow \forall \bar{y}.(\bigwedge_{1 \leq i \leq n} L_i[\bar{y}] \rightarrow L[\bar{y}]))} \ (\mathtt{IndMC})$$

- L and L_i for all $1 \leq i \leq n$ are ground
- $F \rightarrow \forall \bar{y}.(\bigwedge_{1 \leq i \leq n} L_i[\bar{y}] \rightarrow L[\bar{y}])$ is valid induction formula

$$\left(\begin{array}{c} (\mathtt{even}(0) \rightarrow \mathtt{even}(\mathtt{add}(0, \sigma_2)))\wedge \\ (\mathtt{even}(\mathtt{s}(0)) \rightarrow \mathtt{even}(\mathtt{add}(\mathtt{s}(0), \sigma_2)))\wedge \\ \forall z.\left(\begin{array}{c} (\mathtt{even}(z) \rightarrow \mathtt{even}(\mathtt{add}(z, \sigma_2))) \rightarrow \\ (\mathtt{even}(\mathtt{s}(\mathtt{s}(z))) \rightarrow \mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(z)), \sigma_2))) \end{array}\right) \end{array}\right)$$
$$\rightarrow \forall x.(\mathtt{even}(x) \rightarrow \mathtt{even}(\mathtt{add}(x, \sigma_2)))$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

---

1. $\mathtt{even}(\sigma_2)$ [conj.]

5. $\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]

6. $\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

...

13. $\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0)) \vee \mathtt{even}(\mathtt{s}(\mathtt{s}(\sigma_4))) \vee \mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

14. $\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0)) \vee \neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2)) \vee \mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

15. $\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0)) \vee \neg\mathtt{even}(\sigma_4) \vee \mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2)) \vee \mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

...

$$\frac{L_1[\overline{t}] \vee C_1 \quad \dots \quad L_n[\overline{t}] \vee C_n \quad \overline{L}[\overline{t}] \vee C}{\mathtt{cnf}(F \rightarrow \forall \overline{y}.(\bigwedge_{1 \leq i \leq n} L_i[\overline{y}] \rightarrow L[\overline{y}]))} \; (\mathtt{IndMC})$$

- L and L_i for all $1 \leq i \leq n$ are ground
- $F \rightarrow \forall \overline{y}.(\bigwedge_{1 \leq i \leq n} L_i[\overline{y}] \rightarrow L[\overline{y}])$ is valid induction formula

$$\left(\begin{array}{c} (\mathtt{even}(0) \rightarrow \mathtt{even}(\mathtt{add}(0, \sigma_2)))\wedge \\ (\mathtt{even}(\mathtt{s}(0)) \rightarrow \mathtt{even}(\mathtt{add}(\mathtt{s}(0), \sigma_2)))\wedge \\ \forall z.\left(\begin{array}{c} (\mathtt{even}(z) \rightarrow \mathtt{even}(\mathtt{add}(z, \sigma_2))) \rightarrow \\ (\mathtt{even}(\mathtt{s}(\mathtt{s}(z))) \rightarrow \mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(z)), \sigma_2))) \end{array}\right) \end{array}\right) \rightarrow \forall x.(\mathtt{even}(x) \rightarrow \mathtt{even}(\mathtt{add}(x, \sigma_2)))$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1. $\mathtt{even}(\sigma_2)$ [conj.]

5. $\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]

6. $\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

...

13. $\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0)) \vee \mathtt{even}(\mathtt{s}(\mathtt{s}(\sigma_4)))$ [BR]

14. $\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0)) \vee \neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$ [BR]

15. $\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0)) \vee \neg\mathtt{even}(\sigma_4) \vee \mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$ [BR]

...

$$\frac{L_1[\overline{t}] \vee C_1 \quad \ldots \quad L_n[\overline{t}] \vee C_n \quad \overline{L}[\overline{t}] \vee C}{\mathtt{cnf}(F \rightarrow \forall \overline{y}.(\bigwedge_{1 \leq i \leq n} L_i[\overline{y}] \rightarrow L[\overline{y}]))} \;(\mathtt{IndMC})$$

- L and L_i for all $1 \leq i \leq n$ are ground
- $F \rightarrow \forall \overline{y}.(\bigwedge_{1 \leq i \leq n} L_i[\overline{y}] \rightarrow L[\overline{y}])$ is valid induction formula

$$\left(\begin{array}{c}(\mathtt{even}(0) \rightarrow \mathtt{even}(\mathtt{add}(0, \sigma_2)))\wedge \\ (\mathtt{even}(\mathtt{s}(0)) \rightarrow \mathtt{even}(\mathtt{add}(\mathtt{s}(0), \sigma_2)))\wedge \\ \forall z.\left(\begin{array}{c}(\mathtt{even}(z) \rightarrow \mathtt{even}(\mathtt{add}(z, \sigma_2))) \rightarrow \\ (\mathtt{even}(\mathtt{s}(\mathtt{s}(z))) \rightarrow \mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(z)), \sigma_2)))\end{array}\right)\end{array}\right) \rightarrow \forall x.(\mathtt{even}(x) \rightarrow \mathtt{even}(\mathtt{add}(x, \sigma_2)))$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]
	...	
13.	$\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0)) \vee \mathtt{even}(\mathtt{s}(\mathtt{s}(\sigma_4)))$	[BR]
14.	$\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0)) \vee \neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$	[BR]
15.	$\neg\mathtt{even}(\mathtt{add}(0, \sigma_2)) \vee \mathtt{even}(\mathtt{s}(0)) \vee \neg\mathtt{even}(\sigma_4) \vee \mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$	[BR]
	...	

Function definition axioms:

$$\mathtt{add}(0, y) = y$$
$$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$$

$$\mathtt{mult}(0, y) = 0$$
$$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$$

$$\mathtt{even}(0)$$
$$\neg\mathtt{even}(\mathtt{s}(0))$$
$$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$$
$$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \to \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]
	...	
13.	$\neg\mathtt{even}(\sigma_2) \lor \mathtt{even}(\mathtt{s}(0))$ $\lor\ \mathtt{even}(\mathtt{s}(\mathtt{s}(\sigma_4)))$	[BR]
14.	$\neg\mathtt{even}(\sigma_2) \lor \mathtt{even}(\mathtt{s}(0))$ $\lor\ \neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$	[BR]
15.	$\neg\mathtt{even}(\sigma_2) \lor \mathtt{even}(\mathtt{s}(0))$ $\lor\ \neg\mathtt{even}(\sigma_4) \lor \mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$	[BR]
	...	

Function definition axioms:

$$\mathtt{add}(0, y) = y$$
$$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$$
$$\mathtt{mult}(0, y) = 0$$
$$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$$
$$\mathtt{even}(0)$$
$$\neg\mathtt{even}(\mathtt{s}(0))$$
$$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \lor \neg\mathtt{even}(x)$$
$$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \lor \mathtt{even}(x)$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

---

1. $\mathtt{even}(\sigma_2)$ [conj.]

5. $\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]

6. $\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

...

13. $\neg\mathtt{even}(\sigma_2) \vee \mathtt{even}(\mathtt{s}(0))$
$\vee\ \mathtt{even}(\mathtt{s}(\mathtt{s}(\sigma_4)))$ [BR]

14. $\neg\mathtt{even}(\sigma_2) \vee \mathtt{even}(\mathtt{s}(0))$
$\vee\ \neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$ [BR]

15. $\neg\mathtt{even}(\sigma_2) \vee \mathtt{even}(\mathtt{s}(0))$
$\vee\ \neg\mathtt{even}(\sigma_4) \vee \mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$ [BR]

...

Function definition axioms:

$\mathtt{add}(0, y) = y$

$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$

$\mathtt{mult}(0, y) = 0$

$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$

$\mathtt{even}(0)$

$\neg\mathtt{even}(\mathtt{s}(0))$

$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$

$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

---

1. $\mathtt{even}(\sigma_2)$ [conj.]

5. $\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]

6. $\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

...

13. $\mathtt{even}(\mathtt{s}(0)) \vee \mathtt{even}(\mathtt{s}(\mathtt{s}(\sigma_4)))$ [BR]

14. $\mathtt{even}(\mathtt{s}(0)) \vee \neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$ [BR]

15. $\mathtt{even}(\mathtt{s}(0)) \vee \neg\mathtt{even}(\sigma_4) \vee \mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$ [BR]

...

Function definition axioms:

$\mathtt{add}(0, y) = y$

$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$

$\mathtt{mult}(0, y) = 0$

$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$

$\mathtt{even}(0)$

$\neg\mathtt{even}(\mathtt{s}(0))$

$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$

$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]
	...	
13.	$\mathtt{even}(\mathtt{s}(0))$ $\vee\ \mathtt{even}(\mathtt{s}(\mathtt{s}(\sigma_4)))$	[BR]
14.	$\mathtt{even}(\mathtt{s}(0))$ $\vee\ \neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$	[BR]
15.	$\mathtt{even}(\mathtt{s}(0))$ $\vee\ \neg\mathtt{even}(\sigma_4) \vee \mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$	[BR]
	...	

Function definition axioms:

$\mathtt{add}(0, y) = y$

$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$

$\mathtt{mult}(0, y) = 0$

$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$

$\mathtt{even}(0)$

$\neg\mathtt{even}(\mathtt{s}(0))$

$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$

$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]
	...	
13.	$\mathtt{even}(\mathtt{s}(\mathtt{s}(\sigma_4)))$	[BR]
14.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$	[BR]
15.	$\neg\mathtt{even}(\sigma_4) \vee \mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$	[BR]
	...	

Function definition axioms:

$$\mathtt{add}(0, y) = y$$
$$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$$
$$\mathtt{mult}(0, y) = 0$$
$$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$$
$$\mathtt{even}(0)$$
$$\neg\mathtt{even}(\mathtt{s}(0))$$
$$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$$
$$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]
	...	
13.	$\mathtt{even}(\mathtt{s}(\mathtt{s}(\sigma_4)))$	[BR]
14.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$	[BR]
15.	$\neg\mathtt{even}(\sigma_4) \vee \mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$	[BR]
	...	

Function definition axioms:

$\mathtt{add}(0, y) = y$

$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$

$\mathtt{mult}(0, y) = 0$

$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$

$\mathtt{even}(0)$

$\neg\mathtt{even}(\mathtt{s}(0))$

$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$

$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

---

1. $\mathtt{even}(\sigma_2)$ [conj.]
5. $\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]
6. $\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

...

13. $\mathtt{even}(\sigma_4)$ [BR]
14. $\neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$ [BR]
15. $\neg\mathtt{even}(\sigma_4) \vee \mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$ [BR]

...

Function definition axioms:

$$\mathtt{add}(0, y) = y$$
$$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$$

$$\mathtt{mult}(0, y) = 0$$
$$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$$

$$\mathtt{even}(0)$$
$$\neg\mathtt{even}(\mathtt{s}(0))$$
$$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$$
$$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\texttt{even}(y) \rightarrow \texttt{even}(\texttt{mult}(x, y))$

1.	$\texttt{even}(\sigma_2)$	[conj.]
5.	$\texttt{even}(\texttt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\texttt{even}(\texttt{add}(\texttt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]
	...	
13.	$\texttt{even}(\sigma_4)$	[BR]
14.	$\neg\texttt{even}(\texttt{add}(\texttt{s}(\texttt{s}(\sigma_4)), \sigma_2))$	[BR]
15.	$\neg\texttt{even}(\sigma_4) \vee \texttt{even}(\texttt{add}(\sigma_4, \sigma_2))$	[BR]
	...	

Function definition axioms:

$\texttt{add}(0, y) = y$

$\texttt{add}(\texttt{s}(x), y) = \texttt{s}(\texttt{add}(x, y))$

$\texttt{mult}(0, y) = 0$

$\texttt{mult}(\texttt{s}(x), y) = \texttt{add}(\texttt{mult}(x, y), y)$

$\texttt{even}(0)$

$\neg\texttt{even}(\texttt{s}(0))$

$\neg\texttt{even}(\texttt{s}(\texttt{s}(x))) \vee \neg\texttt{even}(x)$

$\texttt{even}(\texttt{s}(\texttt{s}(x))) \vee \texttt{even}(x)$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

---

1. $\mathtt{even}(\sigma_2)$ [conj.]

5. $\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]

6. $\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

$\ldots$

13. $\mathtt{even}(\sigma_4)$ [BR]

14. $\neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$ [BR]

15. $\mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$ [BR]

$\ldots$

Function definition axioms:

$$\mathtt{add}(0, y) = y$$
$$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$$
$$\mathtt{mult}(0, y) = 0$$
$$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$$
$$\mathtt{even}(0)$$
$$\neg\mathtt{even}(\mathtt{s}(0))$$
$$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$$
$$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

---

1. $\mathtt{even}(\sigma_2)$ [conj.]

5. $\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]

6. $\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

...

13. $\mathtt{even}(\sigma_4)$ [BR]

14. $\neg\mathtt{even}(\mathtt{add}(\mathtt{s}(\mathtt{s}(\sigma_4)), \sigma_2))$ [BR]

15. $\mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$ [BR]

...

Function definition axioms:

$\mathtt{add}(0, y) = y$

$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$

$\mathtt{mult}(0, y) = 0$

$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$

$\mathtt{even}(0)$

$\neg\mathtt{even}(\mathtt{s}(0))$

$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$

$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

---

1. $\mathtt{even}(\sigma_2)$ [conj.]

5. $\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]

6. $\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

...

13. $\mathtt{even}(\sigma_4)$ [BR]

14. $\neg\mathtt{even}(\mathtt{s}(\mathtt{add}(\mathtt{s}(\sigma_4), \sigma_2)))$ [BR]

15. $\mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$ [BR]

...

Function definition axioms:

$\mathtt{add}(0, y) = y$

$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$

$\mathtt{mult}(0, y) = 0$

$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$

$\mathtt{even}(0)$

$\neg\mathtt{even}(\mathtt{s}(0))$

$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$

$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]
	...	
13.	$\mathtt{even}(\sigma_4)$	[BR]
14.	$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(\mathtt{add}(\sigma_4, \sigma_2))))$	[BR]
15.	$\mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$	[BR]
	...	

Function definition axioms:

$$\mathtt{add}(0, y) = y$$
$$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$$
$$\mathtt{mult}(0, y) = 0$$
$$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$$
$$\mathtt{even}(0)$$
$$\neg\mathtt{even}(\mathtt{s}(0))$$
$$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$$
$$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]
	...	
13.	$\mathtt{even}(\sigma_4)$	[BR]
14.	$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(\mathtt{add}(\sigma_4, \sigma_2))))$	[BR]
15.	$\mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$	[BR]
	...	

Function definition axioms:

$$\mathtt{add}(0, y) = y$$
$$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$$
$$\mathtt{mult}(0, y) = 0$$
$$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$$
$$\mathtt{even}(0)$$
$$\neg\mathtt{even}(\mathtt{s}(0))$$
$$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$$
$$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\texttt{even}(y) \rightarrow \texttt{even}(\texttt{mult}(x, y))$

1.	$\texttt{even}(\sigma_2)$	[conj.]
5.	$\texttt{even}(\texttt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\texttt{even}(\texttt{add}(\texttt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]
	...	
13.	$\texttt{even}(\sigma_4)$	[BR]
14.	$\neg\texttt{even}(\texttt{add}(\sigma_4, \sigma_2))$	[BR]
15.	$\texttt{even}(\texttt{add}(\sigma_4, \sigma_2))$	[BR]
	...	

Function definition axioms:

$\texttt{add}(0, y) = y$

$\texttt{add}(\texttt{s}(x), y) = \texttt{s}(\texttt{add}(x, y))$

$\texttt{mult}(0, y) = 0$

$\texttt{mult}(\texttt{s}(x), y) = \texttt{add}(\texttt{mult}(x, y), y)$

$\texttt{even}(0)$

$\neg\texttt{even}(\texttt{s}(0))$

$\neg\texttt{even}(\texttt{s}(\texttt{s}(x))) \vee \neg\texttt{even}(x)$

$\texttt{even}(\texttt{s}(\texttt{s}(x))) \vee \texttt{even}(x)$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

1.	$\mathtt{even}(\sigma_2)$	[conj.]
5.	$\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$	[BR]
6.	$\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$	[BR]
	...	
13.	$\mathtt{even}(\sigma_4)$	[BR]
14.	$\neg\mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$	[BR]
15.	$\mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$	[BR]
	...	

Function definition axioms:

$$\mathtt{add}(0, y) = y$$
$$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$$
$$\mathtt{mult}(0, y) = 0$$
$$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$$
$$\mathtt{even}(0)$$
$$\neg\mathtt{even}(\mathtt{s}(0))$$
$$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$$
$$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$$

Multi-clause induction - motivating example II. ctd.

Prove: $\forall x, y.\mathtt{even}(y) \rightarrow \mathtt{even}(\mathtt{mult}(x, y))$

---

1. $\mathtt{even}(\sigma_2)$ [conj.]

5. $\mathtt{even}(\mathtt{mult}(\sigma_3, \sigma_2))$ [BR]

6. $\neg\mathtt{even}(\mathtt{add}(\mathtt{mult}(\sigma_3, \sigma_2), \sigma_2))$ [BR]

...

13. $\mathtt{even}(\sigma_4)$ [BR]

14. $\square$ [BR]

15. $\mathtt{even}(\mathtt{add}(\sigma_4, \sigma_2))$ [BR]

...

Function definition axioms:

$$\mathtt{add}(0, y) = y$$
$$\mathtt{add}(\mathtt{s}(x), y) = \mathtt{s}(\mathtt{add}(x, y))$$
$$\mathtt{mult}(0, y) = 0$$
$$\mathtt{mult}(\mathtt{s}(x), y) = \mathtt{add}(\mathtt{mult}(x, y), y)$$
$$\mathtt{even}(0)$$
$$\neg\mathtt{even}(\mathtt{s}(0))$$
$$\neg\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \neg\mathtt{even}(x)$$
$$\mathtt{even}(\mathtt{s}(\mathtt{s}(x))) \vee \mathtt{even}(x)$$

Motivating example II. – conclusion

We proved $\forall x, y \in \texttt{nat}.\ \texttt{even}(y) \rightarrow \texttt{even}(\texttt{mult}(x, y))$ with:

Motivating example II. – conclusion

We proved $\forall x, y \in \texttt{nat}.\ \texttt{even}(y) \rightarrow \texttt{even}(\texttt{mult}(x, y))$ with:

- an induction formula allowing the use of assumptions (multi-clause induction)

Motivating example II. – conclusion

We proved $\forall x, y \in \texttt{nat}.\ \texttt{even}(y) \rightarrow \texttt{even}(\texttt{mult}(x, y))$ with:

- an induction formula allowing the use of assumptions (multi-clause induction)
- non-trivial induction formulas based on function definition `even`

Motivating example II. – conclusion

We proved $\forall x, y \in \texttt{nat}.\ \texttt{even}(y) \rightarrow \texttt{even}(\texttt{mult}(x, y))$ with:

- an induction formula allowing the use of assumptions (multi-clause induction)
- non-trivial induction formulas based on function definition even
- rewriting with function definitions

Experiments

Implementation: VAMPIRE

Experiments

Implementation: VAMPIRE
Benchmarks: UFDTLIA + own generated set (dty_RD)

Experiments

Implementation: VAMPIRE
Benchmarks: UFDTLIA + own generated set (dty_RD)
Configuration: 300s, 1 core, 16 GB / problem

Experiments

Implementation:	Vampire
Benchmarks:	UFDTLIA + own generated set (dty_RD)
Configuration:	300s, 1 core, 16 GB / problem
	default/portfolio mode

Experiments

Implementation: VAMPIRE
Benchmarks: UFDTLIA + own generated set (dty_RD)
Configuration: 300s, 1 core, 16 GB / problem
default/portfolio mode

	UFDTLIA 327 problems	dty_RD 3,397 problems
VAMPIRE	180 (0)	1,641 (0)
VAMPIRE*	**259 (30)**	**3,223 (497)**
ZIPPERPOSITION	174 (0)	2,534 (21)
CVC4	235 (12)	165 (0)

Experiments

VAMPIRE* forced option	UFDTLIA 327 problems	dty_RD 3,397 problems
default	**259 (1)**	3223 (3)
`-indmc off`	237 (0)	**3259 (33)**
`-indhrw off`	242 (0)	3192 (4)
`-fnrw off`	237 (3)	3001 (0)
`-sik one`	200 (1)	962 (0)

Summary

Improved inductive reasoning with term algebras and recursive function definitions:

Summary

Improved inductive reasoning with term algebras and recursive function definitions:

- new induction formula generation techniques

Summary

Improved inductive reasoning with term algebras and recursive function definitions:

- new induction formula generation techniques
- using function definitions as rewrite rules

Summary

Improved inductive reasoning with term algebras and recursive function definitions:

- new induction formula generation techniques
- using function definitions as rewrite rules
- special treatment of induction hypotheses during inductive steps

Summary

Improved inductive reasoning with term algebras and recursive function definitions:

- new induction formula generation techniques
- using function definitions as rewrite rules
- special treatment of induction hypotheses during inductive steps
- new inference rule for handling multiple clauses within induction

Questions?