

---

# ***Sets***

# Overview

---

- Set notation
- Inductively defined sets

---

# ***Set notation***

# Sets

---

Sets over type 'a:

*'a set*

# Sets

---

Sets over type 'a:

$$'a \text{ set} = 'a \Rightarrow \text{bool}$$

# Sets

---

Sets over type 'a:

*'a set* = *'a ⇒ bool*

- $\{\}$ ,  $\{e_1, \dots, e_n\}$ ,  $\{x. P\ x\}$

# Sets

---

Sets over type 'a:

*'a set* = *'a ⇒ bool*

- $\{\}$ ,  $\{e_1, \dots, e_n\}$ ,  $\{x. P\ x\}$
- $e \in A$ ,  $A \subseteq B$

# Sets

---

Sets over type 'a:

'a set = 'a  $\Rightarrow$  bool

- $\{\}$ ,  $\{e_1, \dots, e_n\}$ ,  $\{x. P\ x\}$
- $e \in A$ ,  $A \subseteq B$
- $A \cup B$ ,  $A \cap B$ ,  $A - B$ ,  $\neg A$

# Sets

---

Sets over type 'a:

'a set = 'a  $\Rightarrow$  bool

- $\{\}$ ,  $\{e_1, \dots, e_n\}$ ,  $\{x. P\ x\}$
- $e \in A$ ,  $A \subseteq B$
- $A \cup B$ ,  $A \cap B$ ,  $A - B$ ,  $- A$
- $\bigcup_{x \in A} B\ x$ ,  $\bigcap_{x \in A} B\ x$

# Sets

---

Sets over type 'a:

'a set = 'a  $\Rightarrow$  bool

- $\{\}$ ,  $\{e_1, \dots, e_n\}$ ,  $\{x. P\ x\}$
- $e \in A$ ,  $A \subseteq B$
- $A \cup B$ ,  $A \cap B$ ,  $A - B$ ,  $- A$
- $\bigcup_{x \in A} B\ x$ ,  $\bigcap_{x \in A} B\ x$
- $\{i..j\}$

# Sets

---

Sets over type 'a:

*'a set* = *'a*  $\Rightarrow$  *bool*

- $\{\}$ ,  $\{e_1, \dots, e_n\}$ ,  $\{x. P\ x\}$
- $e \in A$ ,  $A \subseteq B$
- $A \cup B$ ,  $A \cap B$ ,  $A - B$ ,  $- A$
- $\bigcup_{x \in A} B\ x$ ,  $\bigcap_{x \in A} B\ x$
- $\{i..j\}$
- *insert* :: *'a*  $\Rightarrow$  *'a set*  $\Rightarrow$  *'a set*

# Sets

Sets over type 'a:

*'a set* = *'a*  $\Rightarrow$  *bool*

- $\{\}$ ,  $\{e_1, \dots, e_n\}$ ,  $\{x. P\ x\}$
- $e \in A$ ,  $A \subseteq B$
- $A \cup B$ ,  $A \cap B$ ,  $A - B$ ,  $- A$
- $\bigcup_{x \in A} B\ x$ ,  $\bigcap_{x \in A} B\ x$
- $\{i..j\}$
- *insert* :: *'a*  $\Rightarrow$  *'a set*  $\Rightarrow$  *'a set*
- $f\ 'A = \{y. \exists x \in A. y = f\ x\}$

# Sets

Sets over type 'a:

*'a set* = *'a*  $\Rightarrow$  *bool*

- $\{\}$ ,  $\{e_1, \dots, e_n\}$ ,  $\{x. P\ x\}$
- $e \in A$ ,  $A \subseteq B$
- $A \cup B$ ,  $A \cap B$ ,  $A - B$ ,  $- A$
- $\bigcup_{x \in A} B\ x$ ,  $\bigcap_{x \in A} B\ x$
- $\{i..j\}$
- *insert* :: *'a*  $\Rightarrow$  *'a set*  $\Rightarrow$  *'a set*
- $f\ 'A = \{y. \exists x \in A. y = f\ x\}$
- ...

# *Proofs about sets*

---

Natural deduction proofs:

- equalityI:  $\llbracket A \subseteq B; B \subseteq A \rrbracket \implies A = B$

# *Proofs about sets*

---

Natural deduction proofs:

- equalityI:  $\llbracket A \subseteq B; B \subseteq A \rrbracket \Longrightarrow A = B$
- subsetI:  $(\bigwedge x. x \in A \Longrightarrow x \in B) \Longrightarrow A \subseteq B$

# *Proofs about sets*

---

Natural deduction proofs:

- equalityI:  $\llbracket A \subseteq B; B \subseteq A \rrbracket \Longrightarrow A = B$
- subsetI:  $(\bigwedge x. x \in A \Longrightarrow x \in B) \Longrightarrow A \subseteq B$
- ... (see Tutorial)

---

## ***Demo: proofs about sets***

# *Bounded quantifiers*

---

- $\forall x \in A. P x$

# *Bounded quantifiers*

---

- $\forall x \in A. P x \equiv \forall x. x \in A \longrightarrow P x$

# *Bounded quantifiers*

---

- $\forall x \in A. P x \equiv \forall x. x \in A \longrightarrow P x$
- $\exists x \in A. P x$

# *Bounded quantifiers*

---

- $\forall x \in A. P x \equiv \forall x. x \in A \longrightarrow P x$
- $\exists x \in A. P x \equiv \exists x. x \in A \wedge P x$

# Bounded quantifiers

---

- $\forall x \in A. P x \equiv \forall x. x \in A \longrightarrow P x$
- $\exists x \in A. P x \equiv \exists x. x \in A \wedge P x$
- ballI:  $(\bigwedge x. x \in A \implies P x) \implies \forall x \in A. P x$
- bspec:  $\llbracket \forall x \in A. P x; x \in A \rrbracket \implies P x$

# Bounded quantifiers

---

- $\forall x \in A. P x \equiv \forall x. x \in A \longrightarrow P x$
- $\exists x \in A. P x \equiv \exists x. x \in A \wedge P x$
- ballI:  $(\bigwedge x. x \in A \implies P x) \implies \forall x \in A. P x$
- bspec:  $\llbracket \forall x \in A. P x; x \in A \rrbracket \implies P x$
- bexI:  $\llbracket P x; x \in A \rrbracket \implies \exists x \in A. P x$
- bexE:  $\llbracket \exists x \in A. P x; \bigwedge x. \llbracket x \in A; P x \rrbracket \implies Q \rrbracket \implies Q$

---

# ***Inductively defined sets***

## *Example: even numbers*

---

Informally:

## *Example: even numbers*

---

Informally:

- 0 is even

## *Example: even numbers*

---

Informally:

- 0 is even
- If  $n$  is even, so is  $n + 2$

## *Example: even numbers*

---

Informally:

- 0 is even
- If  $n$  is even, so is  $n + 2$
- These are the only even numbers

## *Example: even numbers*

---

Informally:

- 0 is even
- If  $n$  is even, so is  $n + 2$
- These are the only even numbers

In Isabelle/HOL:

**inductive\_set** *Ev* :: *nat set*

— The set of all even numbers

# Example: even numbers

---

Informally:

- 0 is even
- If  $n$  is even, so is  $n + 2$
- These are the only even numbers

In Isabelle/HOL:

**inductive\_set**  $Ev :: nat\ set$

— The set of all even numbers

**where**

$0 \in Ev \quad |$

$n \in Ev \implies n + 2 \in Ev$

# *Format of inductive definitions*

---

**inductive\_set**  $S :: \tau$  *set*

# *Format of inductive definitions*

---

**inductive\_set**  $S :: \tau$  *set*

**where**

$\llbracket a_1 \in S; \dots ; a_n \in S; A_1; \dots ; A_k \rrbracket \implies a \in S \mid$

$\vdots$

# *Format of inductive definitions*

---

**inductive\_set**  $S :: \tau$  *set*

**where**

$\llbracket a_1 \in S; \dots ; a_n \in S; A_1; \dots ; A_k \rrbracket \implies a \in S \mid$

$\vdots$

where  $A_1; \dots ; A_k$  are side conditions not involving  $S$ .

# *Proving properties of even numbers*

---

Easy:  $4 \in Ev$

$$0 \in Ev \implies 2 \in Ev \implies 4 \in Ev$$

# *Proving properties of even numbers*

---

Easy:  $4 \in Ev$

$$0 \in Ev \implies 2 \in Ev \implies 4 \in Ev$$

Trickier:  $m \in Ev \implies m+m \in Ev$

# Proving properties of even numbers

---

Easy:  $4 \in Ev$

$$0 \in Ev \implies 2 \in Ev \implies 4 \in Ev$$

Trickier:  $m \in Ev \implies m+m \in Ev$

Idea: induction on the length of the derivation of  $m \in Ev$

# Proving properties of even numbers

---

Easy:  $4 \in Ev$

$$0 \in Ev \implies 2 \in Ev \implies 4 \in Ev$$

Trickier:  $m \in Ev \implies m+m \in Ev$

Idea: induction on the length of the derivation of  $m \in Ev$

Better: induction on the *structure* of the derivation

# Proving properties of even numbers

---

Easy:  $4 \in Ev$

$$0 \in Ev \implies 2 \in Ev \implies 4 \in Ev$$

Trickier:  $m \in Ev \implies m+m \in Ev$

Idea: induction on the length of the derivation of  $m \in Ev$

Better: induction on the *structure* of the derivation

Two cases:  $m \in Ev$  is proved by

- rule  $0 \in Ev$

# Proving properties of even numbers

---

Easy:  $4 \in Ev$

$$0 \in Ev \implies 2 \in Ev \implies 4 \in Ev$$

Trickier:  $m \in Ev \implies m+m \in Ev$

Idea: induction on the length of the derivation of  $m \in Ev$

Better: induction on the *structure* of the derivation

Two cases:  $m \in Ev$  is proved by

- rule  $0 \in Ev$   
 $\implies m = 0 \implies 0+0 \in Ev$

# Proving properties of even numbers

---

Easy:  $4 \in Ev$

$$0 \in Ev \implies 2 \in Ev \implies 4 \in Ev$$

Trickier:  $m \in Ev \implies m+m \in Ev$

Idea: induction on the length of the derivation of  $m \in Ev$

Better: induction on the *structure* of the derivation

Two cases:  $m \in Ev$  is proved by

- rule  $0 \in Ev$   
 $\implies m = 0 \implies 0+0 \in Ev$
- rule  $n \in Ev \implies n+2 \in Ev$

# Proving properties of even numbers

---

Easy:  $4 \in Ev$

$$0 \in Ev \implies 2 \in Ev \implies 4 \in Ev$$

Trickier:  $m \in Ev \implies m+m \in Ev$

Idea: induction on the length of the derivation of  $m \in Ev$

Better: induction on the *structure* of the derivation

Two cases:  $m \in Ev$  is proved by

- rule  $0 \in Ev$   
 $\implies m = 0 \implies 0+0 \in Ev$
- rule  $n \in Ev \implies n+2 \in Ev$   
 $\implies m = n+2$  and  $n+n \in Ev$  (ind. hyp.!).

# Proving properties of even numbers

---

Easy:  $4 \in Ev$

$$0 \in Ev \implies 2 \in Ev \implies 4 \in Ev$$

Trickier:  $m \in Ev \implies m+m \in Ev$

Idea: induction on the length of the derivation of  $m \in Ev$

Better: induction on the *structure* of the derivation

Two cases:  $m \in Ev$  is proved by

- rule  $0 \in Ev$   
 $\implies m = 0 \implies 0+0 \in Ev$
- rule  $n \in Ev \implies n+2 \in Ev$   
 $\implies m = n+2$  and  $n+n \in Ev$  (ind. hyp.!)  
 $\implies m+m = (n+2)+(n+2) = ((n+n)+2)+2 \in Ev$

## ***Rule induction for Ev***

---

To prove

$$n \in Ev \implies P n$$

by *rule induction* on  $n \in Ev$  we must prove

## ***Rule induction for Ev***

---

To prove

$$n \in Ev \implies P n$$

by *rule induction* on  $n \in Ev$  we must prove

- $P 0$

## Rule induction for $Ev$

---

To prove

$$n \in Ev \implies P n$$

by *rule induction* on  $n \in Ev$  we must prove

- $P 0$
- $P n \implies P(n+2)$

## Rule induction for $Ev$

---

To prove

$$n \in Ev \implies P n$$

by *rule induction* on  $n \in Ev$  we must prove

- $P 0$
- $P n \implies P(n+2)$

Rule  $Ev.induct$ :

$$\llbracket n \in Ev; P 0; \bigwedge n. P n \implies P(n+2) \rrbracket \implies P n$$

## Rule induction for $Ev$

---

To prove

$$n \in Ev \implies P n$$

by *rule induction* on  $n \in Ev$  we must prove

- $P 0$
- $P n \implies P(n+2)$

Rule  $Ev.induct$ :

$$\llbracket n \in Ev; P 0; \bigwedge n. P n \implies P(n+2) \rrbracket \implies P n$$

An elimination rule

# *Rule induction in general*

---

Set  $S$  is defined inductively.

# *Rule induction in general*

---

Set  $S$  is defined inductively.

To prove

$$x \in S \implies P x$$

by *rule induction* on  $x \in S$

# *Rule induction in general*

---

Set  $S$  is defined inductively.

To prove

$$x \in S \implies P x$$

by *rule induction* on  $x \in S$

we must prove for every rule

$$\llbracket a_1 \in S; \dots ; a_n \in S \rrbracket \implies a \in S$$

that  $P$  is preserved:

# *Rule induction in general*

---

Set  $S$  is defined inductively.

To prove

$$x \in S \implies P x$$

by *rule induction* on  $x \in S$

we must prove for every rule

$$\llbracket a_1 \in S; \dots ; a_n \in S \rrbracket \implies a \in S$$

that  $P$  is preserved:

$$\llbracket P a_1; \dots ; P a_n \rrbracket \implies P a$$

# *Rule induction in general*

---

Set  $S$  is defined inductively.

To prove

$$x \in S \implies P x$$

by *rule induction* on  $x \in S$

we must prove for every rule

$$\llbracket a_1 \in S; \dots ; a_n \in S \rrbracket \implies a \in S$$

that  $P$  is preserved:

$$\llbracket P a_1; \dots ; P a_n \rrbracket \implies P a$$

In Isabelle/HOL:

`apply(erule S.induct)`

---

## ***Demo: inductively defined sets***

# *Inductive predicates*

---

$$x \in S \rightsquigarrow S x$$

# *Inductive predicates*

---

$$x \in S \rightsquigarrow S x$$

Example:

**inductive** *Ev* :: *nat*  $\Rightarrow$  *bool*

**where**

*Ev* 0 |

*Ev* n  $\Longrightarrow$  *Ev* (n + 2)

# Inductive predicates

---

$$x \in S \rightsquigarrow S x$$

Example:

**inductive** *Ev* :: *nat*  $\Rightarrow$  *bool*

**where**

*Ev* 0 |

*Ev* n  $\Longrightarrow$  *Ev* (n + 2)

Comparison:

**predicate:** simpler syntax

**set:** direct usage of  $\cup$  etc

# Inductive predicates

---

$$x \in S \rightsquigarrow S x$$

Example:

**inductive** *Ev* :: *nat*  $\Rightarrow$  *bool*

**where**

*Ev* 0 |

*Ev* n  $\Longrightarrow$  *Ev* (n + 2)

Comparison:

**predicate:** simpler syntax

**set:** direct usage of  $\cup$  etc

Inductive predicates can be of type  $\tau_1 \Rightarrow \dots \Rightarrow \tau_n \Rightarrow \mathit{bool}$