Application of Mathematical Logic in Functional Program Verification

Nikolaj Popov and Tudor Jebelean

Research Institute for Symbolic Computation, Linz

{popov,jebelean}@risc.uni-linz.ac.at

Outline

Preconditions and Postconditions. Total Correctness

Given the triple

$\{I\}F\{O\}$ (Input condition, Function definition, Output condition)

Total Correctness Formula

$(\forall n : I[n])\ (F[n] \downarrow \wedge O[n, F[n]])$

Example

$\{x \in \mathbb{R} \wedge n \in \mathbb{N}\}$

$pow[x, n] =$ **If** $n = 0$ **then** 1 **else** $x * pow[x, n-1]$

$\{x^n = pow[x, n]\}$

$(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (pow[x, n] \downarrow \wedge\ x^n = pow[x, n])$

Preconditions and Postconditions. Total Correctness

Given the triple

$\{I\}F\{O\}$ (Input condition, Function definition, Output condition)

Total Correctness Formula

$(\forall n : I[n])\ (F[n]\downarrow \wedge O[n, F[n]])$

Example

$\{x \in \mathbb{R} \wedge n \in \mathbb{N}\}$

$pow[x, n] = $ **If** $n = 0$ **then** 1 **else** $x * pow[x, n-1]$

$\{x^n = pow[x, n]\}$

$(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (pow[x, n]\downarrow \wedge x^n = pow[x, n])$

Preconditions and Postconditions. Total Correctness

Given the triple

$\{I\}F\{O\}$ (Input condition, Function definition, Output condition)

Total Correctness Formula

$(\forall n : I[n])\ (F[n]\downarrow \wedge O[n, F[n]])$

Example

$\{x \in \mathbb{R} \wedge n \in \mathbb{N}\}$

$pow[x, n] =$ **If** $n = 0$ **then** 1 **else** $x * pow[x, n-1]$

$\{x^n = pow[x, n]\}$

$(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (pow[x, n]\downarrow \wedge\ x^n = pow[x, n])$

Building up Correct Programs

Basic Functions e.g. +, -, *, etc.

New Functions in Terms of Already Known Functions

- Input and output predicates;
- Prove total correctness.

Modularity. After proving correctness, use only the specification.

$\{x \in \mathbb{R} \wedge n \in \mathbb{N}\}$ *Input condition*

$pow[x, n] = \ldots$

$\{x^n = pow[x, n]\}$ *Output condition*

Building up Correct Programs

Basic Functions e.g. +, -, *, etc.

New Functions in Terms of Already Known Functions

- Input and output predicates;
- Prove total correctness;

Modularity. After proving correctness, use only the specification.

$\{x \in \mathbb{R} \wedge n \in \mathbb{N}\}$ *Input condition*

$pow[x, n] = \ldots$

$\{x^n = pow[x, n]\}$ *Output condition*

Building up Correct Programs

Basic Functions e.g. +, -, *, etc.

New Functions in Terms of Already Known Functions

- Input and output predicates;
- Prove total correctness;

Modularity. After proving correctness, use only the specification.

$\{x \in \mathbb{R} \wedge n \in \mathbb{N}\}$ *Input condition*

$pow[x, n] = \ldots$

$\{x^n = pow[x, n]\}$ *Output condition*

Building up Correct Programs

Basic Functions e.g. +, -, *, etc.

New Functions in Terms of Already Known Functions

- Input and output predicates;
- Prove total correctness;

Modularity. After proving correctness, use only the specification.

$\{x \in \mathbb{R} \wedge n \in \mathbb{N}\}$ *Input condition*

$pow[x, n] = \ldots$

$\{x^n = pow[x, n]\}$ *Output condition*

Building up Correct Programs

Basic Functions e.g. +, -, *, etc.

New Functions in Terms of Already Known Functions

- Input and output predicates;
- Prove total correctness;

Modularity. After proving correctness, use only the specification.

$\{x \in \mathbb{R} \wedge n \in \mathbb{N}\}$ *Input condition*

$pow[x, n] = \ldots$

$\{x^n = pow[x, n]\}$ *Output condition*

Building up Correct Programs

Appropriate values for the auxiliary functions

No input condition of an auxiliary function will be violated

Example

$F[x] =$ **If** $Q[x]$ **then** $H[x]$ **else** $G[x]$

- $(\forall x : I_F[x]) \; (Q[x] \Rightarrow I_H[x])$
- $(\forall x : I_F[x]) \; (\neg Q[x] \Rightarrow I_G[x])$

Building up Correct Programs

Appropriate values for the auxiliary functions

No input condition of an auxiliary function will be violated

Example

$F[x] =$ **If** $Q[x]$ **then** $H[x]$ **else** $G[x]$

- $(\forall x : I_F[x])\ (Q[x] \Rightarrow I_H[x])$
- $(\forall x : I_F[x])\ (\neg Q[x] \Rightarrow I_G[x])$

Building up Correct Programs

Appropriate values for the auxiliary functions

No input condition of an auxiliary function will be violated

Example

$F[x] =$ **If** $Q[x]$ **then** $H[x]$ **else** $G[x]$

- $(\forall x : \; I_F[x]) \;\; (Q[x] \Rightarrow I_H[x])$
- $(\forall x : \; I_F[x]) \;\; (\neg Q[x] \Rightarrow I_G[x])$

Building up Correct Programs

Appropriate values for the auxiliary functions

No input condition of an auxiliary function will be violated

Example

$F[x] =$ **If** $Q[x]$ **then** $H[x]$ **else** $G[x]$

- $(\forall x : \; I_F[x]) \; (Q[x] \Rightarrow I_H[x])$
- $(\forall x : \; I_F[x]) \; (\neg Q[x] \Rightarrow I_G[x])$

Building up Correct Programs

Appropriate values for the auxiliary functions

No input condition of an auxiliary function will be violated

Example

$F[x] = $ **If** $Q[x]$ **then** $H[x]$ **else** $G[x]$

- $(\forall x : I_F[x])\ (Q[x] \Rightarrow I_H[x])$
- $(\forall x : I_F[x])\ (\neg Q[x] \Rightarrow I_G[x])$

Coherent Programs

Simple Recursive Programs

$F[x] =$ **If** $Q[x]$ **then** $S[x]$ **else** $C[x, F[R[x]]]$

Conditions for coherency

- $(\forall x : I_F[x])\ (Q[x] \Rightarrow I_S[x])$
- $(\forall x : I_F[x])\ (\neg Q[x] \Rightarrow I_F[R[x]])$
- $(\forall x : I_F[x])\ (\neg Q[x] \Rightarrow I_R[x])$
- $(\forall x, y : I_F[x])\ (\neg Q[x] \wedge O_F[R[x], y] \Rightarrow I_C[x, y])$

Coherent Programs

Simple Recursive Programs

$F[x] = \textbf{ If } Q[x] \textbf{ then } S[x] \textbf{ else } C[x, F[R[x]]]$

Conditions for coherency

- $(\forall x : I_F[x])\ (Q[x] \Rightarrow I_S[x])$
- $(\forall x : I_F[x])\ (\neg Q[x] \Rightarrow I_F[R[x]])$
- $(\forall x : I_F[x])\ (\neg Q[x] \Rightarrow I_R[x])$
- $(\forall x, y : I_F[x])\ (\neg Q[x] \wedge O_F[R[x], y] \Rightarrow I_C[x, y])$

Coherent Programs

Simple Recursive Programs

$F[x] =$ **If** $Q[x]$ **then** $S[x]$ **else** $C[x, F[R[x]]]$

Conditions for coherency

- $(\forall x : \ I_F[x]) \ (Q[x] \Rightarrow I_S[x])$
- $(\forall x : \ I_F[x]) \ (\neg Q[x] \Rightarrow I_F[R[x]])$
- $(\forall x : \ I_F[x]) \ (\neg Q[x] \Rightarrow I_R[x])$
- $(\forall x, y : \ I_F[x]) \ (\neg Q[x] \wedge O_F[R[x], y] \Rightarrow I_C[x, y])$

Coherent Programs

Simple Recursive Programs

$F[x] = $ **If** $Q[x]$ **then** $S[x]$ **else** $C[x, F[R[x]]]$

Conditions for coherency

- $(\forall x : I_F[x])\ (Q[x] \Rightarrow I_S[x])$
- $(\forall x : I_F[x])\ (\neg Q[x] \Rightarrow I_F[R[x]])$
- $(\forall x : I_F[x])\ (\neg Q[x] \Rightarrow I_R[x])$
- $(\forall x, y : I_F[x])\ (\neg Q[x] \wedge O_F[R[x], y] \Rightarrow I_C[x, y])$

Coherent Programs

Simple Recursive Programs

$F[x] = $ **If** $Q[x]$ **then** $S[x]$ **else** $C[x, F[R[x]]]$

Conditions for coherency

- $(\forall x : \ I_F[x]) \ (Q[x] \Rightarrow I_S[x])$
- $(\forall x : \ I_F[x]) \ (\neg Q[x] \Rightarrow I_F[R[x]])$
- $(\forall x : \ I_F[x]) \ (\neg Q[x] \Rightarrow I_R[x])$
- $(\forall x, y : \ I_F[x]) \ (\neg Q[x] \wedge O_F[R[x], y] \Rightarrow I_C[x, y])$

Coherent Programs

Simple Recursive Programs

$F[x] = $ **If** $Q[x]$ **then** $S[x]$ **else** $C[x, F[R[x]]]$

Conditions for coherency

- $(\forall x : \ I_F[x]) \ (Q[x] \Rightarrow I_S[x])$
- $(\forall x : \ I_F[x]) \ (\neg Q[x] \Rightarrow I_F[R[x]])$
- $(\forall x : \ I_F[x]) \ (\neg Q[x] \Rightarrow I_R[x])$
- $(\forall x, y : \ I_F[x]) \ (\neg Q[x] \wedge O_F[R[x], y] \Rightarrow I_C[x, y])$

Verification Conditions Generation

Simple Recursive Program

$F[x] = $ **If** $Q[x]$ **then** $S[x]$ **else** $C[x, F[R[x]]]$

is correct if the verification conditions hold

- $(\forall x : I_F[x])\ (Q[x] \Rightarrow O_F[x, S[x]])$

Verification Conditions Generation

Simple Recursive Program

$F[x] = $ **If** $Q[x]$ **then** $S[x]$ **else** $C[x, F[R[x]]]$

is correct if the verification conditions hold

- $(\forall x : I_F[x])\ (Q[x] \Rightarrow O_F[x, S[x]])$
- $(\forall x, y : I_F[x])\ (\neg Q[x] \wedge O_F[R[x], y] \Rightarrow O_F[x, C[x, y]])$
- $(\forall x : I_F[x])\ (\neg Q[x] \Rightarrow F'[R[x]] = 0)$
- where:

$$F'[x] = \textbf{If } Q[x] \textbf{ then } 0 \textbf{ else } F'[R[x]]$$

Verification Conditions Generation

Simple Recursive Program

$F[x] = \textbf{ If } Q[x] \textbf{ then } S[x] \textbf{ else } C[x, F[R[x]]]$

is correct if the verification conditions hold

- $(\forall x : \; I_F[x]) \; (Q[x] \Rightarrow O_F[x, S[x]])$
- $(\forall x, y : \; I_F[x]) \; (\neg Q[x] \wedge O_F[R[x], y] \Rightarrow O_F[x, C[x, y]])$
- $(\forall x : \; I_F[x]) \; (\neg Q[x] \Rightarrow F'[R[x]] = 0)$
- where:

$$F'[x] = \textbf{ If } Q[x] \textbf{ then } 0 \textbf{ else } F'[R[x]]$$

Verification Conditions Generation

Simple Recursive Program

$F[x] = \textbf{ If } Q[x] \textbf{ then } S[x] \textbf{ else } C[x, F[R[x]]]$

is correct if the verification conditions hold

- $(\forall x : I_F[x]) \ (Q[x] \Rightarrow O_F[x, S[x]])$
- $(\forall x, y : I_F[x]) \ (\neg Q[x] \wedge O_F[R[x], y] \Rightarrow O_F[x, C[x, y]])$
- $(\forall x : I_F[x]) \ (\neg Q[x] \Rightarrow F'[R[x]] = 0)$
- where:

$$F'[x] = \textbf{ If } Q[x] \textbf{ then } 0 \textbf{ else } F'[R[x]]$$

Verification Conditions Generation

Simple Recursive Program

$F[x] = $ **If** $Q[x]$ **then** $S[x]$ **else** $C[x, F[R[x]]]$

is correct if the verification conditions hold

- $(\forall x: \ I_F[x]) \ (Q[x] \Rightarrow O_F[x, S[x]])$
- $(\forall x, y: \ I_F[x]) \ (\neg Q[x] \wedge O_F[R[x], y] \Rightarrow O_F[x, C[x, y]])$
- $(\forall x: \ I_F[x]) \ (\neg Q[x] \Rightarrow F'[R[x]] = 0)$
- where:

$$F'[x] = \textbf{ If } Q[x] \textbf{ then } 0 \textbf{ else } F'[R[x]]$$

Verification Conditions Generation

Simple Recursive Program

$F[x] =$ **If** $Q[x]$ **then** $S[x]$ **else** $C[x, F[R[x]]]$

is correct if the verification conditions hold

- $(\forall x : \; I_F[x]) \; (Q[x] \Rightarrow O_F[x, S[x]])$
- $(\forall x, y : \; I_F[x]) \; (\neg Q[x] \wedge O_F[R[x], y] \Rightarrow O_F[x, C[x, y]])$
- $(\forall x : \; I_F[x]) \; (\neg Q[x] \Rightarrow F'[R[x]] = 0)$
- where:

$$F'[x] = \textbf{ If } Q[x] \textbf{ then } 0 \textbf{ else } F'[R[x]]$$

Soundness and Completeness

$\langle \mathit{Program}, \mathit{Specification} \rangle \overset{VCG}{\longrightarrow} \mathit{VerificationConditions}$

$\langle F[x], \langle I_F[x], O_F[x, F[x]] \rangle \rangle \overset{VCG}{\longrightarrow} \mathit{VerificationConditions}$

Soundness

if $\models \varphi_1[x] \wedge \cdots \wedge \varphi_n[x]$

then $\forall n\, (I[n] \Rightarrow F[n] \downarrow \wedge O[n, F[n]])$

Completeness

if $\forall n\, (I[n] \Rightarrow F[n] \downarrow \wedge O[n, F[n]])$

then $\models \varphi_1[x] \wedge \cdots \wedge \varphi_n[x]$

Soundness and Completeness

$\langle \mathit{Program}, \mathit{Specification} \rangle \xrightarrow{VCG} \mathit{VerificationConditions}$

$\langle F[x], \langle I_F[x], O_F[x, F[x]] \rangle \rangle \xrightarrow{VCG} \mathit{VerificationConditions}$

Soundness

if $\models \varphi_1[x] \wedge \cdots \wedge \varphi_n[x]$

then $\forall n\, (I[n] \Rightarrow F[n] \downarrow \wedge O[n, F[n]])$

Completeness

if $\forall n\, (I[n] \Rightarrow F[n] \downarrow \wedge O[n, F[n]])$

then $\models \varphi_1[x] \wedge \cdots \wedge \varphi_n[x]$

Soundness and Completeness

$$\langle \mathit{Program}, \mathit{Specification} \rangle \stackrel{VCG}{\longrightarrow} \mathit{VerificationConditions}$$

$$\langle F[x], \langle I_F[x], O_F[x, F[x]] \rangle \rangle \stackrel{VCG}{\longrightarrow} \mathit{VerificationConditions}$$

Soundness

if $\models \varphi_1[x] \wedge \cdots \wedge \varphi_n[x]$

then $\forall n \, (I[n] \Rightarrow F[n] \downarrow \wedge O[n, F[n]])$

Completeness

if $\forall n \, (I[n] \Rightarrow F[n] \downarrow \wedge O[n, F[n]])$

then $\models \varphi_1[x] \wedge \cdots \wedge \varphi_n[x]$

Soundness and Completeness

$$\langle Program, Specification \rangle \stackrel{VCG}{\longrightarrow} VerificationConditions$$

$$\langle F[x], \langle I_F[x], O_F[x, F[x]] \rangle \rangle \stackrel{VCG}{\longrightarrow} VerificationConditions$$

Soundness

if $\models \varphi_1[x] \wedge \cdots \wedge \varphi_n[x]$

then $\forall n \, (I[n] \Rightarrow F[n] \downarrow \wedge O[n, F[n]])$

Completeness

if $\forall n \, (I[n] \Rightarrow F[n] \downarrow \wedge O[n, F[n]])$

then $\models \varphi_1[x] \wedge \cdots \wedge \varphi_n[x]$

Example

Sum $(\forall n : \mathbb{N})\ (Sum[n] = \frac{n(n+1)}{2})$

$$Sum[n] = \quad \textbf{If } n = 0 \textbf{ then } 0$$
$$\textbf{else } n + Sum[n-1].$$

is coherent if

- $(\forall n : \mathbb{N})\ (n \neq 0 \Rightarrow n \in \mathbb{N})$
- $(\forall n : \mathbb{N})\ (n = 0 \Rightarrow T)$

Example

Sum $(\forall n : \mathbb{N})\ (Sum[n] = \frac{n(n+1)}{2})$

$$Sum[n] = \quad \textbf{If } n = 0 \textbf{ then } 0$$
$$\textbf{else } n + Sum[n-1].$$

is coherent if

- $(\forall n : \mathbb{N})\ (n \neq 0 \ \Rightarrow n \in \mathbb{N})$
- $(\forall n : \mathbb{N})\ (n = 0 \ \Rightarrow \mathbf{T})$

Example

Sum $(\forall n : \mathbb{N})\ (Sum[n] = \frac{n(n+1)}{2})$

$$Sum[n] = \quad \textbf{If } n = 0 \textbf{ then } 0$$
$$\textbf{else } n + Sum[n-1].$$

is coherent if

- $(\forall n : \mathbb{N})\ (n \neq 0 \ \Rightarrow n \in \mathbb{N})$
- $(\forall n : \mathbb{N})\ (n = 0 \ \Rightarrow \mathbf{T})$

Example

Sum $(\forall n : \mathbb{N})\ (Sum[n] = \frac{n(n+1)}{2})$

$$Sum[n] = \begin{array}{l} \textbf{If } n = 0 \textbf{ then } 0 \\ \textbf{else } n + Sum[n-1]. \end{array}$$

is coherent if

- $(\forall n : \mathbb{N})\ (n \neq 0 \ \Rightarrow n \in \mathbb{N})$
- $(\forall n : \mathbb{N})\ (n = 0 \ \Rightarrow \mathbf{T})$

Example

Sum $(\forall n : \mathbb{N})\ (Sum[n] = \frac{n(n+1)}{2})$

$$Sum[n] = \quad \textbf{If } n = 0 \textbf{ then } 0$$
$$\textbf{else } n + Sum[n-1].$$

is correct if and only if

- $(\forall n : \mathbb{N})\ (n = 0 \ \Rightarrow 0 = \frac{n(n+1)}{2})$
- $(\forall n, m : \mathbb{N})\ (n \neq 0 \land m = \frac{(n-1)((n-1)+1)}{2} \ \Rightarrow n + m = \frac{n(n+1)}{2})$
- $(\forall n : \mathbb{N})\ (Sum'[n] = 0)$
- where:

$$Sum'[n] = \ \textbf{If } n = 0 \textbf{ then } 0 \textbf{ else } Sum'[n-1]$$

Example

Sum $(\forall n : \mathbb{N})\ (Sum[n] = \frac{n(n+1)}{2})$

$$Sum[n] = \begin{array}{l} \textbf{If } n = 0 \textbf{ then } 0 \\ \textbf{else } n + Sum[n-1]. \end{array}$$

is correct if and only if

- $(\forall n : \mathbb{N})\ (n = 0 \ \Rightarrow 0 = \frac{n(n+1)}{2})$
- $(\forall n, m : \mathbb{N})\ (n \neq 0 \wedge m = \frac{(n-1)((n-1)+1)}{2} \ \Rightarrow n + m = \frac{n(n+1)}{2})$
- $(\forall n : \mathbb{N})\ (Sum'[n] = 0)$
- where:

$$Sum'[n] = \textbf{ If } n = 0 \textbf{ then } 0 \textbf{ else } Sum'[n-1]$$

Example

Sum $(\forall n : \mathbb{N})\ (Sum[n] = \frac{n(n+1)}{2})$

$$Sum[n] = \begin{array}{l} \textbf{If } n = 0 \textbf{ then } 0 \\ \textbf{else } n + Sum[n-1]. \end{array}$$

is correct if and only if

- $(\forall n : \mathbb{N})\ (n = 0 \ \Rightarrow 0 = \frac{n(n+1)}{2})$
- $(\forall n, m : \mathbb{N})\ (n \neq 0 \wedge m = \frac{(n-1)((n-1)+1)}{2} \ \Rightarrow n + m = \frac{n(n+1)}{2})$
- $(\forall n : \mathbb{N})\ (Sum'[n] = 0)$
- where:

$$Sum'[n] = \textbf{ If } n = 0 \textbf{ then } 0 \textbf{ else } Sum'[n-1]$$

Example

Sum $(\forall n : \mathbb{N})\ (Sum[n] = \frac{n(n+1)}{2})$

$$Sum[n] = \begin{array}{l} \textbf{If } n = 0 \textbf{ then } 0 \\ \textbf{else } n + Sum[n-1]. \end{array}$$

is correct if and only if

- $(\forall n : \mathbb{N})\ (n = 0 \ \Rightarrow 0 = \frac{n(n+1)}{2})$
- $(\forall n, m : \mathbb{N})\ (n \neq 0 \wedge m = \frac{(n-1)((n-1)+1)}{2} \ \Rightarrow n + m = \frac{n(n+1)}{2})$
- $(\forall n : \mathbb{N})\ (Sum'[n] = 0)$
- where:

$$Sum'[n] = \textbf{If } n = 0 \textbf{ then } 0 \textbf{ else } Sum'[n-1]$$

Example

Sum $(\forall n : \mathbb{N})\ (Sum[n] = \frac{n(n+1)}{2})$

$$Sum[n] = \quad \textbf{If } n = 0 \textbf{ then } 0$$
$$\textbf{else } n + Sum[n-1].$$

is correct if and only if

- $(\forall n : \mathbb{N})\ (n = 0 \ \Rightarrow 0 = \frac{n(n+1)}{2})$
- $(\forall n, m : \mathbb{N})\ (n \neq 0 \wedge m = \frac{(n-1)((n-1)+1)}{2} \ \Rightarrow n + m = \frac{n(n+1)}{2})$
- $(\forall n : \mathbb{N})\ (Sum'[n] = 0)$
- where:

$$Sum'[n] = \ \textbf{If } n = 0 \textbf{ then } 0 \textbf{ else } Sum'[n-1]$$

Example

Binary powering $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ P[x, n] = x^n$

$$P[x, n] = \quad \textbf{If } n = 0 \textbf{ then } 1$$
$$\textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x * x, n/2]$$
$$\textbf{else } x * P[x * x, (n-1)/2].$$

is coherent if

- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \Rightarrow \mathrm{Even}[n])$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \Rightarrow \mathrm{Even}[n-1])$

Example

Binary powering $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ P[x, n] = x^n$

$$P[x,n] = \begin{array}{l} \textbf{If } n = 0 \textbf{ then } 1 \\ \textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x * x, n/2] \\ \textbf{else } x * P[x * x, (n-1)/2]. \end{array}$$

is coherent if

- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \ \Rightarrow \mathrm{Even}[n])$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \ \Rightarrow \mathrm{Even}[n-1])$

Example

Binary powering $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ P[x,n] = x^n$

$$P[x,n] = \begin{array}{l} \textbf{If } n = 0 \textbf{ then } 1 \\ \textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x * x, n/2] \\ \textbf{else } x * P[x * x, (n-1)/2]. \end{array}$$

is coherent if

- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \ \Rightarrow \mathrm{Even}[n])$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \ \Rightarrow \mathrm{Even}[n-1])$

Example

Binary powering $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ P[x, n] = x^n$

$$P[x,n] = \begin{array}{l} \textbf{If } n = 0 \textbf{ then } 1 \\ \textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x*x, n/2] \\ \textbf{else } x * P[x*x, (n-1)/2]. \end{array}$$

is coherent if

- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \ \Rightarrow \mathrm{Even}[n])$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \ \Rightarrow \mathrm{Even}[n-1])$

Example

Binary powering $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ P[x,n] = x^n$

$$P[x,n] = \begin{array}{l}\textbf{If } n = 0 \textbf{ then } 1 \\ \textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x*x, n/2] \\ \textbf{else } x*P[x*x,(n-1)/2].\end{array}$$

is correct if and only if

- $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ (n = 0 \ \Rightarrow 1 = x^n)$
- $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \ \Rightarrow n/2 \in \mathbb{N})$
- $(\forall x,m:\mathbb{R})(\forall n:\mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \wedge m = (x*x)^{n/2} \ \Rightarrow m = x^n)$
- $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \ \Rightarrow (n-1)/2 \in \mathbb{N})$
- $(\forall x,m:\mathbb{R})(\forall n:\mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \wedge m = (x*x)^{(n-1)/2} \ \Rightarrow x*m = x^n)$
- $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ (P'[x,n] = 0)$

Example

Binary powering $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ P[x, n] = x^n$

$$P[x, n] = \quad \textbf{If } n = 0 \textbf{ then } 1$$
$$\textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x * x, n/2]$$
$$\textbf{else } x * P[x * x, (n - 1)/2].$$

is correct if and only if

- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n = 0 \ \Rightarrow 1 = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \ \Rightarrow n/2 \in \mathbb{N})$
- $(\forall x, m : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \wedge m = (x * x)^{n/2} \ \Rightarrow m = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \ \Rightarrow (n - 1)/2 \in \mathbb{N})$
- $(\forall x, m : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \wedge m = (x * x)^{(n-1)/2} \ \Rightarrow x * m = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (P'[x, n] = 0)$

Example

Binary powering $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ P[x, n] = x^n$

$$P[x, n] = \quad \textbf{If } n = 0 \textbf{ then } 1$$
$$\textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x * x, n/2]$$
$$\textbf{else } x * P[x * x, (n - 1)/2].$$

is correct if and only if

- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n = 0 \ \Rightarrow 1 = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \ \Rightarrow n/2 \in \mathbb{N})$
- $(\forall x, m : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \wedge m = (x * x)^{n/2} \ \Rightarrow m = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \ \Rightarrow (n - 1)/2 \in \mathbb{N})$
- $(\forall x, m : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \wedge m = (x * x)^{(n-1)/2} \ \Rightarrow x * m = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (P'[x, n] = 0)$

Example

Binary powering $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ P[x,n]=x^n$

$$P[x,n] = \begin{array}{l}\textbf{If } n=0 \textbf{ then } 1\\ \textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x*x,n/2]\\ \textbf{else } x*P[x*x,(n-1)/2].\end{array}$$

is correct if and only if

- $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ (n=0\ \Rightarrow 1=x^n)$
- $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ (n\neq 0\wedge \mathrm{Even}[n]\ \Rightarrow n/2\in\mathbb{N})$
- $(\forall x,m:\mathbb{R})(\forall n:\mathbb{N})\ (n\neq 0\wedge \mathrm{Even}[n]\wedge m=(x*x)^{n/2}\ \Rightarrow m=x^n)$
- $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ (n\neq 0\wedge \neg\mathrm{Even}[n]\ \Rightarrow (n-1)/2\in\mathbb{N})$
- $(\forall x,m:\mathbb{R})(\forall n:\mathbb{N})\ (n\neq 0\wedge \neg\mathrm{Even}[n]\wedge m=(x*x)^{(n-1)/2}\ \Rightarrow x*m=x^n)$
- $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ (P'[x,n]=0)$

Example

Binary powering $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ P[x,n] = x^n$

$$P[x,n] = \begin{array}{l} \textbf{If } n = 0 \textbf{ then } 1 \\ \textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x * x, n/2] \\ \textbf{else } x * P[x * x, (n-1)/2]. \end{array}$$

is correct if and only if

- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n = 0\ \Rightarrow 1 = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n]\ \Rightarrow n/2 \in \mathbb{N})$
- $(\forall x, m : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \wedge m = (x * x)^{n/2}\ \Rightarrow m = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n]\ \Rightarrow (n-1)/2 \in \mathbb{N})$
- $(\forall x, m : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \wedge m = (x * x)^{(n-1)/2}\ \Rightarrow x * m = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (P'[x,n] = 0)$

Example

Binary powering $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ P[x,n]=x^n$

$$P[x,n] = \begin{array}{l}\textbf{If } n=0 \textbf{ then } 1\\ \textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x*x,n/2]\\ \textbf{else } x*P[x*x,(n-1)/2].\end{array}$$

is correct if and only if

- $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ (n=0\ \Rightarrow 1=x^n)$
- $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ (n\neq 0\wedge \mathrm{Even}[n]\ \Rightarrow n/2\in\mathbb{N})$
- $(\forall x,m:\mathbb{R})(\forall n:\mathbb{N})\ (n\neq 0\wedge \mathrm{Even}[n]\wedge m=(x*x)^{n/2}\ \Rightarrow m=x^n)$
- $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ (n\neq 0\wedge \neg\mathrm{Even}[n]\ \Rightarrow (n-1)/2\in\mathbb{N})$
- $(\forall x,m:\mathbb{R})(\forall n:\mathbb{N})\ (n\neq 0\wedge \neg\mathrm{Even}[n]\wedge m=(x*x)^{(n-1)/2}\ \Rightarrow x*m=x^n)$
- $(\forall x:\mathbb{R})(\forall n:\mathbb{N})\ (P'[x,n]=0)$

Example

Binary powering $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ P[x, n] = x^n$

$$P[x,n] = \quad \textbf{If } n = 0 \textbf{ then } 1$$
$$\textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x * x, n/2]$$
$$\textbf{else } x * P[x * x, (n-1)/2].$$

is correct if and only if

- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n = 0\ \Rightarrow 1 = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n]\ \Rightarrow n/2 \in \mathbb{N})$
- $(\forall x, m : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \wedge m = (x * x)^{n/2}\ \Rightarrow m = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n]\ \Rightarrow (n-1)/2 \in \mathbb{N})$
- $(\forall x, m : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \wedge m = (x * x)^{(n-1)/2}\ \Rightarrow x * m = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (P'[x, n] = 0)$

Counter-Example

Binary powering $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ P[x, n] = x^n$

$$P[x, n] = \begin{array}{l} \textbf{If } n = 0 \textbf{ then } 0 \\ \textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x * x, n/2] \\ \textbf{else } x * P[x * x, (n-1)/2]. \end{array}$$

is correct if and only if

- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n = 0 \ \Rightarrow 0 = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \ \Rightarrow n/2 \in \mathbb{N})$
- $(\forall x, m : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \wedge m = (x * x)^{n/2} \ \Rightarrow m = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \ \Rightarrow (n-1)/2 \in \mathbb{N})$
- $(\forall x, m : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \wedge m = (x * x)^{(n-1)/2} \ \Rightarrow x * m = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (P'[x, n] = 0)$

Counter-Example

Binary powering $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ P[x, n] = x^n$

$$P[x, n] = \begin{array}{l} \textbf{If } n = 0 \textbf{ then } \mathbf{0} \\ \textbf{elseif } \mathrm{Even}[n] \textbf{ then } P[x * x, n/2] \\ \textbf{else } x * P[x * x, (n-1)/2]. \end{array}$$

is correct if and only if

- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n = 0 \ \Rightarrow \mathbf{0} = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \ \Rightarrow n/2 \in \mathbb{N})$
- $(\forall x, m : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \mathrm{Even}[n] \wedge m = (x * x)^{n/2} \ \Rightarrow m = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \ \Rightarrow (n-1)/2 \in \mathbb{N})$
- $(\forall x, m : \mathbb{R})(\forall n : \mathbb{N})\ (n \neq 0 \wedge \neg\mathrm{Even}[n] \wedge m = (x * x)^{(n-1)/2} \ \Rightarrow x * m = x^n)$
- $(\forall x : \mathbb{R})(\forall n : \mathbb{N})\ (P'[x, n] = 0)$

Outline

Conclusions and Discussion

- The problem of proving program correctness is translated into a problem of proving first order formulae;
- Prove by hand;
- Prove by an automatic theorem prover.

Conclusions and Discussion

- The problem of proving program correctness is translated into a problem of proving first order formulae;
- Prove by hand;
- Prove by an automatic theorem prover.

Conclusions and Discussion

- The problem of proving program correctness is translated into a problem of proving first order formulae;
- Prove by hand;
- Prove by an automatic theorem prover.