

Analyse et Conception Formelle

Lesson 1

— Propositional logic First order logic

Bibliography

- *Cours de logique pour l'informatique*, J-F. Raskin, <http://www.ulb.ac.be/di/ssd/jfr/info-148.html>
- *Logique et fondements de l'informatique* de Richard Lassaigne et Michel de Rougemont. Hermes 1993.

A selected bibliography on the Isabelle/HOL prover

- <http://people.irisa.fr/Thomas.Genet/ACF/BiblioIsabelle>

The web page of the course

- <http://people.irisa.fr/Thomas.Genet/ACF>

Solutions of Isabelle/HOL exercises (uploaded after each lecture)

- <http://people.irisa.fr/Thomas.Genet/ACFSol>

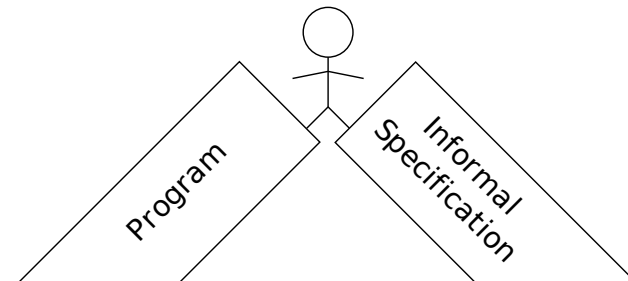
_____Acknowledgements_____

- Many thanks to T. Nipkow, J. Blanchette, L. Bulwahn and G. Riou for providing material, answering questions and for fruitful discussions.

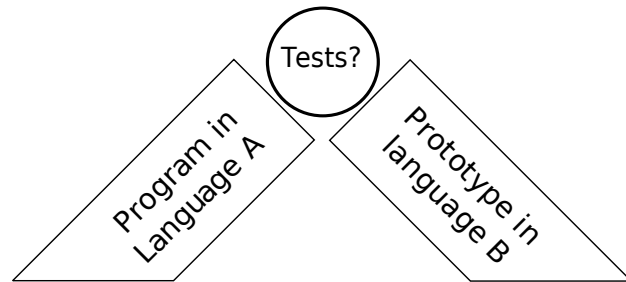
Outline

- Why using logic for specifying/verifying programs?
- Propositional logic
 - ▶ Formula syntax
 - ▶ Interpretations and models
 - ▶ Isabelle/HOL commands
- First-order logic
 - ▶ Formula syntax
 - ▶ Interpretations and models
 - ▶ Isabelle/HOL commands

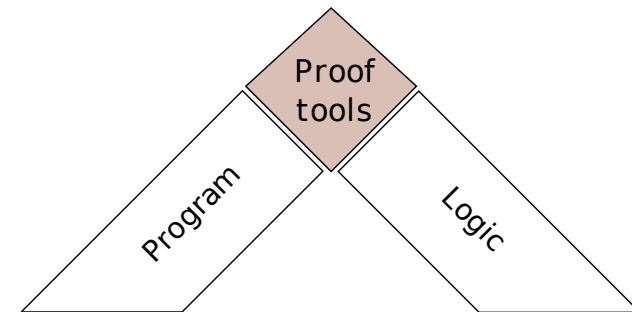
Why using logic for specifying/verifying programs?



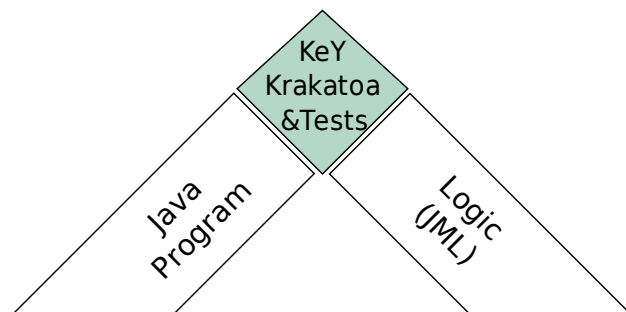
Why using logic for specifying/verifying programs?



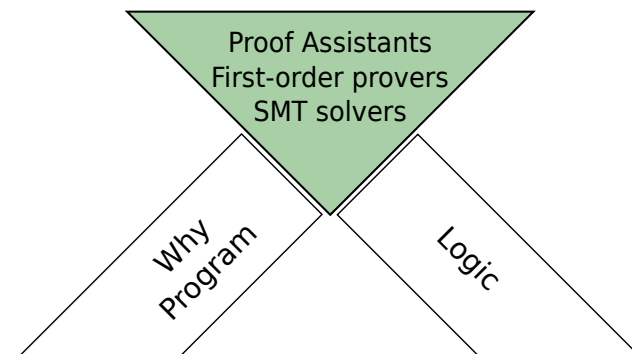
Why using logic for specifying/verifying programs?



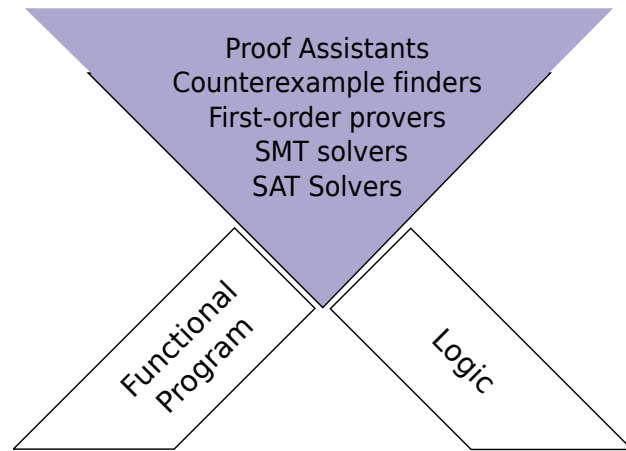
Why using functional paradigm to program?



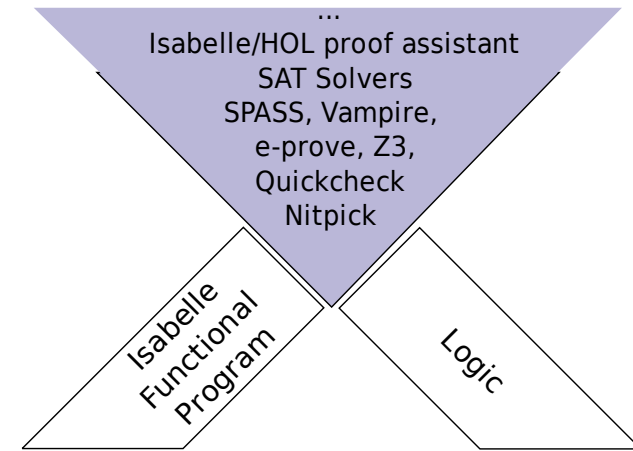
Why using functional paradigm to program?



Why using functional paradigm to program?



Why using functional paradigm to program?



Propositional logic: syntax and interpretations

Definition 1 (Propositional formula)

Let P be a set of propositional variables. The set of propositional formula is defined by

$$\phi ::= p \mid \neg\phi \mid \phi_1 \vee \phi_2 \mid \phi_1 \wedge \phi_2 \mid \phi_1 \longrightarrow \phi_2 \quad \text{where } p \in P$$

Definition 2 (Propositional interpretation)

An *interpretation* I associates to variables of P a value in $\{\text{True}, \text{False}\}$.

Example 3

Let $\phi = (p_1 \wedge p_2) \longrightarrow p_3$. Let I be the interpretation such that $I[p_1] = \text{True}$, $I[p_2] = \text{True}$ and $I[p_3] = \text{False}$.

Propositional logic: syntax and interpretations (II)

We extend the domain of I to formulas as follows:

$$I[\neg\phi] = \begin{cases} \text{True} & \text{iff } I[\phi] = \text{False} \\ \text{False} & \text{iff } I[\phi] = \text{True} \end{cases}$$

$$I[\phi_1 \vee \phi_2] = \text{True} \text{ iff } I[\phi_1] = \text{True} \text{ or } I[\phi_2] = \text{True}$$

$$I[\phi_1 \wedge \phi_2] = \text{True} \text{ iff } I[\phi_1] = \text{True} \text{ and } I[\phi_2] = \text{True}$$

$$I[\phi_1 \longrightarrow \phi_2] = \text{True} \text{ iff } \begin{cases} I[\phi_1] = \text{False} \text{ or} \\ I[\phi_1] = \text{True} \text{ and } I[\phi_2] = \text{True} \end{cases}$$

Example 4

Let $\phi = (p_1 \wedge p_2) \longrightarrow p_3$ and I the interpretation such that $I[p_1] = \text{True}$, $I[p_2] = \text{True}$ and $I[p_3] = \text{False}$.

We have $I[p_1 \wedge p_2] = \text{True}$ and $I[(p_1 \wedge p_2) \longrightarrow p_3] = \text{False}$.

Propositional logic: syntax and interpretations (III)

The presentation using truth tables is generally preferred:

a	$\neg a$
False	True
True	False

a	b	$a \vee b$
False	False	False
True	False	True
False	True	True
True	True	True

a	b	$a \wedge b$
False	False	False
True	False	False
False	True	False
True	True	True

a	b	$a \rightarrow b$
False	False	True
True	False	False
False	True	True
True	True	True

Propositional logic: models

Definition 5 (Propositional model)

I is a *model* of ϕ , denoted by $I \models \phi$, if $I[\phi] = \text{True}$.

Definition 6 (Valid formula/Tautology)

A formula ϕ is *valid*, denoted by $\models \phi$, if for all I we have $I \models \phi$.

Example 7

Let $\phi = (p_1 \wedge p_2) \rightarrow p_3$ and $\phi' = (p_1 \wedge p_2) \rightarrow p_1$. Let I be the interpretation such that $I[p_1] = \text{True}$, $I[p_2] = \text{True}$ and $I[p_3] = \text{False}$. We have $I \not\models \phi$, $I \models \phi'$, and $\models \phi'$.

Propositional logic: decidability and tools in Isabelle/HOL

Property 1

In propositional logic, given ϕ , the following problems are decidable:

- Is $I \models \phi$?
 - Is there an interpretation I such that $I \models \phi$?
 - Is there an interpretation I such that $I \not\models \phi$?
- To automatically prove that $\models \phi$ **apply auto**
(if the formula is not valid, there remains some unsolved goals)
 - To build I such that $I \not\models \phi$ (or $I \models \neg\phi$) **nitpick**
(i.e. find a counterexample... may take some time on large formula)
- _____ Other useful commands _____
- To close the proof of a proven formula **done**
 - To abandon the proof of an unprovable formula **oops**
 - To abandon the proof of (potentially) provable formula **sorry**

Writing and proving propositional formulas in Isabelle/HOL

Example 8 (Valid formula)

```
lemma "(p1 /\ p2) --> p1"
  apply auto
  done
```

Example 9 (Unprovable formula)

```
lemma "(p1 /\ p2) --> p3"
  nitpick
  oops
```

Isabelle/HOL: ASCII notations

Symbol	ASCII notation
True	True
False	False
\wedge	\wedge
\vee	\vee
\neg	\sim
\neq	$\sim =$
\longrightarrow	$-->$
\longleftrightarrow	$=$
\forall	ALL
\exists	?
λ	%

Propositional logic: exercises in Isabelle/HOL

Exercise 1

Using Isabelle/HOL, for each formula, say if it is valid or give a counterexample interpretation, otherwise.

- 1 $A \vee B$
- 2 $((A \wedge B) \longrightarrow \neg C) \vee (A \longrightarrow B) \longrightarrow A \longrightarrow C$
- 3 If it rains, Robert takes his umbrella. Robert does not have his umbrella hence it does not rain.
- 4 $(A \longrightarrow B) \longleftrightarrow (\neg A \vee B)$

First-order logic (FOL) / Predicate logic

- 1 Terms and Formulas
- 2 Interpretations
- 3 Models
- 4 Logic consequence and verification

First-order logic: terms

Definition 10 (Terms)

Let \mathcal{F} be a set of symbols and ar a function such that $ar : \mathcal{F} \Rightarrow \mathbb{N}$ associating each symbol of \mathcal{F} to its arity (the number of parameter). Let \mathcal{X} be a variable set.

The set $\mathcal{T}(\mathcal{F}, \mathcal{X})$, the set of *terms* built on \mathcal{F} and \mathcal{X} , is defined by:
 $\mathcal{T}(\mathcal{F}, \mathcal{X}) = \mathcal{X} \cup \{f(t_1, \dots, t_n) \mid ar(f) = n \text{ and } t_1, \dots, t_n \in \mathcal{T}(\mathcal{F}, \mathcal{X})\}$

Example 11

Let $\mathcal{F} = \{f : 1, g : 2, a : 0, b : 0\}$ and $\mathcal{X} = \{x, y, z\}$.

$f(x)$, a , z , $g(g(a, x), f(a))$, $g(x, x)$ are terms and belong to $\mathcal{T}(\mathcal{F}, \mathcal{X})$.

f , $a(b)$, $f(a, b)$, $x(a)$, $f(a, f(b))$ do not belong to $\mathcal{T}(\mathcal{F}, \mathcal{X})$.

First-order logic: formula syntax

Definition 12 (Formulas)

Let P be a set of predicate symbols all having an arity, i.e. $ar : P \Rightarrow \mathbb{N}$.
The set of formulas defined on \mathcal{F}, \mathcal{X} and P is:

$\phi ::= \neg\phi \mid \phi_1 \vee \phi_2 \mid \phi_1 \wedge \phi_2 \mid \phi_1 \longrightarrow \phi_2 \mid \forall x.\phi \mid \exists x.\phi \mid p(t_1, \dots, t_n)$

where $t_1, \dots, t_n \in \mathcal{T}(\mathcal{F}, \mathcal{X})$, $x \in \mathcal{X}$, $p \in P$ and $ar(p) = n$.

Example 13

Let $P = \{p : 1, q : 2, \leq : 2\}$, $\mathcal{F} = \{f : 1, g : 2, a : 0\}$ and $\mathcal{X} = \{x, y, z\}$.
The following expressions are all formulas:

- $p(f(a))$
- $q(g(f(a), x), y)$
- $\forall x.\exists y.y \leq x$
- $\forall x.\forall y.\forall z.x \leq y \wedge y \leq z \longrightarrow x \leq z$

Interlude: a touch of lambda-calculus

We need to define *anonymous* functions

- Classical notation for functions

$$f : \mathbb{N} \times \mathbb{N} \Rightarrow \mathbb{N}$$

$$f(x, y) = x + y$$

or, for short, $f : \mathbb{N}^2 \Rightarrow \mathbb{N}$

$$f(x, y) = x + y$$

- Lambda-notation of functions

$$f : \mathbb{N}^2 \Rightarrow \mathbb{N}$$

$$f = \lambda x y. x + y$$

$\lambda x y. x + y$ is an anonymous function adding two naturals

This corresponds to OCaml notations

- `fun x y -> x+y`, and
- `function x -> function y -> x + y`

Interlude: a touch of lambda-calculus (in Isabelle HOL)

Isabelle/HOL also use function update using $(:=)$ as in:

- $(\lambda x.x)(0 := 1, 1 := 2)$ the identity function except for 0 that is mapped to 1 and 1 that is mapped to 2
- $(\lambda x..)(a := b)$ a function taking one parameter and whose result is unspecified except for value a that is mapped to b

Predicates in Isabelle/HOL

- A predicate is a function mapping values to $\{\text{True}, \text{False}\}$

For instance the predicate p on $\{a, b\}$

$$p = (\lambda x..)(a := \text{False}, b := 0)$$

First-order formulas: interpretations and valuations

Definition 14 (First-order interpretation)

Let ϕ be a formula and D a domain. An *interpretation* I of ϕ on the domain D associates:

- a function $f_I : D^n \Rightarrow D$ to each symbol $f \in \mathcal{F}$ such that $ar(f) = n$,
- a function $p_I : D^n \Rightarrow \{\text{True}, \text{False}\}$ to each predicate symbol $p \in P$ such that $ar(p) = n$.

Example 15 (Some interpretations of $\phi = \forall x.ev(x) \longrightarrow od(s(x))$)

- Let I be the interpretation such that domain $D = \mathbb{N}$ and $s_I \equiv \lambda x.x + 1$ $ev_I \equiv \lambda x.((x \bmod 2) = 0)$ $od_I \equiv \lambda x.((x \bmod 2) = 1)$
- Let I' be the interpretation such that domain $D = \{a, b\}$ and $s_{I'} \equiv \lambda x.if\ x = a\ then\ b\ else\ a$ $ev_{I'} \equiv \lambda x.(x = a)$ $od_{I'} \equiv \lambda x.False$

Definition 16 (Valuation)

Let D be a domain. A *valuation* V is a function $V : \mathcal{X} \Rightarrow D$.

First-order logic: interpretations and valuations (II)

Definition 17

The interpretation I of a formula ϕ for a valuation V is defined by:

- $(I, V)[[x]] = V(x)$ if $x \in \mathcal{X}$
- $(I, V)[[f(t_1, \dots, t_n)]] = f_I((I, V)[[t_1]], \dots, (I, V)[[t_n]])$ if $f \in \mathcal{F}$ and $ar(f) = n$
- $(I, V)[[p(t_1, \dots, t_n)]] = p_I((I, V)[[t_1]], \dots, (I, V)[[t_n]])$ if $p \in \mathcal{P}$ and $ar(p) = n$
- $(I, V)[[\phi_1 \vee \phi_2]] = \text{True}$ iff $(I, V)[[\phi_1]] = \text{True}$ or $(I, V)[[\phi_2]] = \text{True}$
- etc...
- $(I, V)[[\forall x. \phi]] = \bigwedge_{d \in D} (I, V + \{x \mapsto d\})[[\phi]]$
- $(I, V)[[\exists x. \phi]] = \bigvee_{d \in D} (I, V + \{x \mapsto d\})[[\phi]]$

where $(V + \{x \mapsto d\})(x) = d$ and $(V + \{x \mapsto d\})(y) = V(y)$ if $x \neq y$.

First-order logic: satisfiability, models, tautologies

Definition 18 (Satisfiability)

I and V satisfy ϕ (denoted by $(I, V) \models \phi$) if $(I, V)[[\phi]] = \text{True}$.

Definition 19 (First-order Model)

An interpretation I is a *model* of ϕ , denoted by $I \models \phi$, if for all valuation V we have $(I, V) \models \phi$.

Definition 20 (First-order Tautology)

A formula ϕ is a tautology if all its interpretations are models, i.e. $(I, V) \models \phi$ for all interpretations I and all valuations V .

Remark 1

Free variables are universally quantified (e.g. $P(x)$ equivalent to $\forall x. P(x)$)

First-order logic: decidability and tools in Isabelle/HOL

Property 2

In first-order logic, given ϕ , the following problems are *undecidable*:

- $Is \models \phi$?
 - Is there an interpretation I (and valuation V) such that $(I, V) \models \phi$?
 - Is there an interpretation I (and valuation V) such that $(I, V) \not\models \phi$?
- Try to automatically prove $\models \phi$ **apply auto**
Uses decision procedures (e.g. arithmetic) to try to prove the formula.
If it does not succeed, it does not mean that the formula is unprovable!
 - Try to build I and V such that $(I, V) \not\models \phi$ **nitpick**
If it does not succeed, it does not mean that there is no counterexample!

First-order logic: exercises in Isabelle/HOL

Exercise 2

Using Isabelle/HOL, for each formula, say if it is valid or give a counterexample interpretation and valuation otherwise.

- 1 $\forall x. p(x) \longrightarrow \exists x. p(x)$
- 2 $\exists x. p(x) \longrightarrow \forall x. p(x)$
- 3 $\forall x. ev(x) \longrightarrow od(s(x))$
- 4 $\forall x y. x > y \longrightarrow x + 1 > y + 1$
- 5 $x > y \longrightarrow x + 1 > y + 1$
- 6 $\forall m n. (\neg(m < n) \wedge m < n + 1) \longrightarrow m = n$
- 7 $\forall x. \exists y. x + y = 0$
- 8 $\forall y. (\neg p(f(y))) \longleftrightarrow p(f(y))$
- 9 $\forall y. (p(f(y)) \longrightarrow p(f(y + 1)))$

Isabelle/HOL notations: priority, associativity, shorthands

- Here are the logical operators in decreasing order of priority:
 - ▶ $=, \neg, \wedge, \vee, \longrightarrow, \forall, \exists$
 - ▶ «a priority operator first chooses its operands»
- For instance
 - ▶ $\neg\neg P = P$ means $\neg\neg(P = P)$!
 - ▶ $A \wedge B = B \wedge A$ means $A \wedge (B = B) \wedge A$!
 - ▶ $P \wedge \forall x. Q(x)$ will be parsed as $(P \wedge \forall)x. Q(x)$!
Hence, write $P \wedge (\forall x. Q(x))$ instead!
- All binary operators are associative to the right, for instance $A \longrightarrow B \longrightarrow C$ is equivalent to $A \longrightarrow (B \longrightarrow C)$
- Nested quantifications $\forall x. \forall y. \forall z. P$ can be abbreviated into $\forall x y z. P$
- Free variables are universally quantified, i.e. $P(x)$ is equiv. to $\forall x. P(x)$

All Isabelle/HOL tools will prefer $P(x)$ to $\forall x. P(x)$

First-order logic: satisfiability and models

Definition 21 (Satisfiable formula)

A formula ϕ is *satisfiable* if there exists an interpretation I and a valuation V such that $(I, V) \models \phi$.

Example 22

Let $\phi = p(f(y))$ with $\mathcal{F} = \{f\}$, $P = \{p\}$, $\mathcal{X} = \{y\}$.

The formula ϕ is satisfiable (there exists (I, V) such that $(I, V) \models \phi$)

- Let I be the interp. s.t. $D = \{0, 1\}$, $p_I \equiv \lambda x.(x = 0)$, $f_I = \lambda x.x$
- Let V be the valuation such that $V(y) = 0$

We have $(I, V) \models \phi$. With $V'(y) = 1$, $(I, V') \not\models \phi$. Hence, I is not a model of ϕ .

- Let I' be the interp. s.t. $D = \{0, 1\}$, $p_{I'} \equiv \lambda x.(x = 0)$, $f_{I'} = \lambda x.0$

We have $(I', V) \models \phi$ for all valuation V , hence I' is a model of ϕ .

First-order logic: contradictions

Definition 23 (Contradiction)

A formula is *contradictory* (or *unsatisfiable*) if it cannot be satisfied, i.e. $(I, V) \not\models \phi$ for all interpretation I and all valuation V .

Property 3

A formula ϕ is contradictory iff $\neg\phi$ is a tautology.

Example 24 (See in Isabelle cm1.thy file)

Let $\phi = (\neg p(f(y)) \longleftrightarrow p(f(y)))$. The formula ϕ is contradictory and $\neg\phi$ is a tautology.