

Introduction to Prolog

York University

Department of Computer Science and Engineering

Overview

- Introduction & Preliminaries
- Syntax
 - Characters
 - Constants
 - Variables
 - Operators
- Arithmetic

[ref.: Chapter 1 & 2- Clocksin]

Prolog

- From Wikipedia:

Prolog is a general purpose logic programming language associated with artificial intelligence and computational linguistics.

Prolog has its roots in first-order logic, and unlike many other programming languages, Prolog is declarative: the program logic is expressed in terms of relations, represented as facts and rules. A computation is initiated by running a query over these relations.

The language was first conceived by a group around Alain Colmerauer in Marseille, France, in the early 1970s and the first Prolog system was developed in 1972 by Colmerauer with Philippe Roussel.

- Note the goal is to use PROLOG to learn logic programming. The goal is not just to learn to program in Prolog.

Running Prolog

- We will use SWI-Prolog here.
- In the prism lab:
 - enter *swipl* or *pl*
- Download from www.swi-prolog.org

A sample program

student(john, 3401):-.
student(mary, 3401):-.
study_hard(john):-.



← Facts

← A rule

pass_3401(X):- student(X, 3401), study_hard(X).

:-pass_3401(john).

← A goal

In Prolog

```
consult(user).  
  student(john, 3401).  
  student(mary, 3401).  
  study_hard(john).  
  
  pass_3401(X):- student(X, 3401), study_hard(X).
```

Ctrl+D

```
:-pass_3401(john).  
true.  
:- pass_3401(mary).  
false.
```

Introductory notes

- Predicates shown by a name starting with lower case letters.
- Arguments are written in parentheses, separated by commas.
- A dot (period) comes at the end of each clause.
- The predicate name and the number of arguments is decided by the user.
- Queries are answered by matching with the knowledge given (facts and rules).
- A **false** means 'not enough knowledge to prove it' (does not mean it is actually false, maybe not enough knowledge given to Prolog).

In Prolog

```
consult(user).
```

```
student(john, 3401).
```

```
student(mary, 3401).
```

```
study_hard(john).
```

```
pass_3401(X):- student(X, 3401), study_hard(X).
```

Ctrl+D

A relationship defined by a predicate

An object constant

An object variable

Facts

A rule

```
:-pass_3401(john).
```

```
true.
```

```
:- pass_3401(mary).
```

```
false.
```

A satisfiable goal

An unsatisfiable goal or 'not able to prove satisfiable'

Goals

Prolog- Syntax

- Characters
 - Upper case A, B, ..., Z
 - Lower case a, b, ..., z
 - Digits 0, 1, ...,9
 - Sign characters +, -, ...\$, &
- Object Variables
 - Their names begin with a capital letter or ‘_’
 - The anonymous variable: _ (with a peculiar characteristic!)

Prolog- Syntax (cont.)

- Object Constants
 - Two categories: Atoms and Numbers
 - Numbers, such as 25, 9.99, -30, 7.08e-15
 - Symbols used for atoms are made up of letters and digits, must normally begin with a lower case, e.g. mary, john, ...
 - If enclosed in single quotes, may have any character, e.g. 'George-Smith', '23g', ...
 - Underline symbol '_' may be anywhere except the first letter
 - Some constants are made from signs only

Prolog- Syntax (cont.)

- Structures or compound terms
 - Defined as a function (called a functor in Prolog) and its arguments (called components)
 - e.g. `book(programming_in_prolog, clocksin, 2003)`

functor components
 - Can use anonymous variable when not enough info, e.g. `book(book123,_,_)`

Operators

- They are functions or predicates, implemented for easier use.
 - For example instead of $+(3,4)$ we can write $3+4$
 - $+$ is an **infix** operator
 - Instead of $+(*(2,3), *(8,9))$, we can write $2*3+8*9$
 - $*$ is another operator with a higher priority (**precedence**) over $+$
 - Arithmetic operators are **left associative**, e.g. $8/4/4$ means $(8/4)/4$
 - We will learn how to define new operators later in the course.

Arithmetic

- Some arithmetic operators
 - $X + Y$, $X - Y$, $X * Y$, X / Y
 - $X // Y$ (integer quotient)
 - $X \bmod Y$ (remainder)

 - $X ::= Y$ (same numbers ?)
 - $X \neq Y$ (different numbers?)

 - $X < Y$, $X > Y$, $X \leq Y$, $X \geq Y$ (comparisons)

 - is: an infix operator, evaluating the right-hand argument, then matching with left-hand argument

Examples

?- 3:=3.
true.

?- 3:=4.
false.

?- 3:=2+1.
true.

?- is(3,2+1).
true.

?- is(2+1,3).
false.

?- 2>3.
false.

?- 2+3.
ERROR: toplevel: Undefined procedure: (+)/2

?- is(X,2+3).
X = 5.

?- is(X,5/2).
X = 2.5.

?- is(X,5//2).
X = 2.

?- is(X,5 mod 2).
X = 1.

? X is 5//2.
X=2.

Example

pop(china, 800).

area(china, 4).

density(X,Y):-

pop(X,P),

area(X,A),

Y is P/A.

The population density of a country X is Y, if:

The population of X is P, and

The area of X is A, and

Y is calculated by dividing P by A.

:- density(china, X).

X=200.