Basic Prolog

Comp3031 Lab 06
Fall 2014

SU Zhiyang, JIA Xiaoying

What’s Prolog?

- Prolog is short for PROgramming in LOGic
- Conceived in early 1970s, first developed in 1972
- Prolog is one of the first logic programming languages, and remains the most popular among such languages today [Wikipedia]

Prolog Usage

- Widely used by AI researchers
  - Theorem proving
  - Expert systems
  - Games
  - Automated answering systems
  - Ontology
  - Control systems

Rules and Facts

- Prolog programs describe relations by clauses
- Rules
  - Clauses with bodies
  - Head :- Body.
  - Read as: “Head is true if Body is true”.
  - Body consists of calls to predicates

Rules and Facts

- Prolog programs describe relations by clauses
- Facts
  - Clauses with empty bodies
  - Example
    - cat(tom).
    - Equivalent to the rule:
    - cat(tom) :- true.
    - The built-in predicate “true” is always true

Query

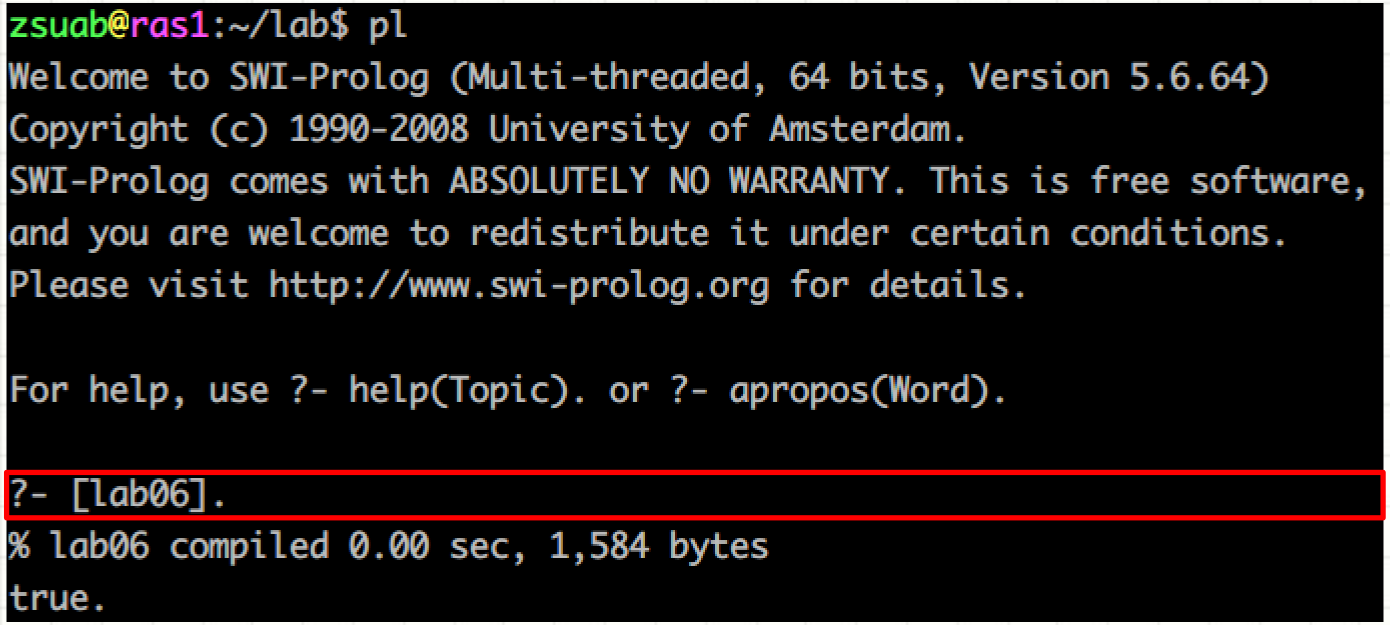
- *Is tom a cat?*
  - ?- cat(tom).
  - true.
- *What things are cats?*
  - ?- cat(X).
  - X = tom

Where to Store Rules, Facts and Queries?

- Rules and facts
  - In a file (*.pl)
  - Load into Prolog
- Queries
  - Enter at toplevel
- Stupid design?
  - No
  - cat(tom).
    - In a file: state a fact
    - In toplevel: ask Prolog to try proving whether tom is a cat

Where to Store Rules, Facts and Queries?

- Load file in Prolog
  - [filename] - load “filename.pl”
  - [‘filename.xx’] - load “filename.xx”



```
zsuab@ras1:~/lab$ pl
Welcome to SWI-Prolog (Multi-threaded, 64 bits, Version 5.6.64)
Copyright (c) 1990-2008 University of Amsterdam.
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to redistribute it under certain conditions.
Please visit http://www.swi-prolog.org for details.

For help, use ?- help(Topic). or ?- apropos(Word).

?- [lab06].
% lab06 compiled 0.00 sec, 1,584 bytes
true.
```

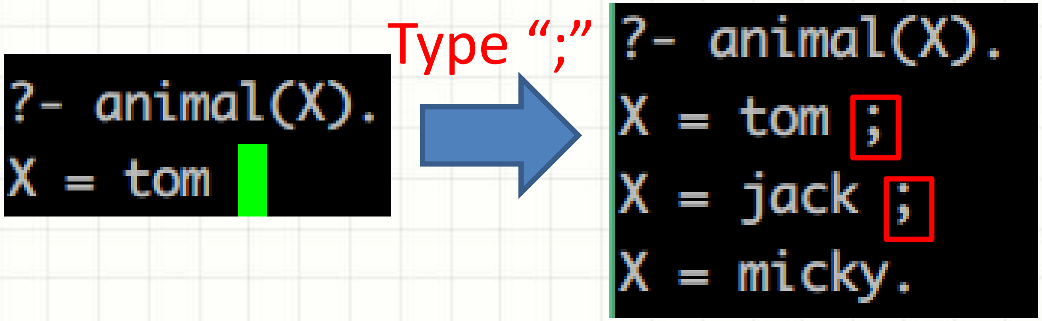
Download lab06.pl from the lab website

Rules, Facts and Queries Example

Add these rules

- cat(tom).
- dog(jack).
- mouse(micky).
- animal(X) :- cat(X).
- animal(X) :- dog(X).
- animal(X) :- mouse(X).

Ask *what things are animals?*



```
?- animal(X).
X = tom
```

Type “;”

```
?- animal(X).
X = tom ;
X = jack ;
X = micky.
```

- Prolog can produce all of the possible answers
  - If the user types a semicolon ';' , Prolog will look for a next answer
  - If the user just hits Enter, then Prolog stops looking for answers

Comparison Operators

- Two types of comparisons in Prolog
- Term comparisons to compare the terms literally
  - ==, \==
- Arithmetic comparisons to compare the arithmetic values of the terms
  - =:=, =\=, <, =<, >, >=

Comparison Operators Examples

- ?- 2+1 == 3.
- false
- ?- 2+1 =:= 3.
- true
- 2+1 \== 3.
- true
- 2+1 =\= 3.
- false

Exercise

- Download family.pl
- Modify the file in order to achieve the following goals
  - Modify the sibling rule so that 'James' will not be shown as an answer for **sibling(X, 'James')**
  - Write a query to list the people who are older than 30 in this family
  - Define a new relation **brother(X,Y)** where X and Y are siblings and X is male