COM S 342

Recitation 12/2/2019 – 12/4/2019

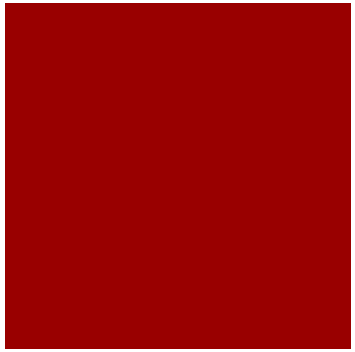
Topic

- Logic programming examples and exercises

- Q&A

Example 1

```
%% food.pl
indian(curry).
indian(dahl).
indian(tandoori).
indian(kurma).
mild(dahl).
mild(tandoori).
mild(kurma).
chinese(chow_mein).
chinese(chop_suey).
chinese(sweet_and_sour).
italian(pizza).
italian(spaghetti).
likes(sam,Food) :-
        indian(Food),
        mild(Food).
likes(sam,Food) :-
         chinese(Food).
likes(sam,Food) :-
         italian(Food).
likes(sam,chips).
```



Prolog

- prolog food.pl
  - If you install SWI-Prolog you can use swipl
- Find out what food sam likes

  ?- likes(sam, X).
- Type semi-colon (;) to see more values for X or type point (.) to stop the query
- ?- likes(dan, X). %% should return false.

Prolog

- Prolog supports integer variables and integer arithmetic
- Use the *is* operator
  - Takes an arithmetic expression as right operand
  - A variable as left operand
  - C is 17 + 10.

Arithmetic Expressions

```
?- X is 10 + 5.
- X = 15.
- true.
?- X is 10 * 5.
- X = 50.
- true.
```

Arithmetic Expressions

- The semantics are not the same of assignment. Example:
  - Sum is Sum + 5. %% error in prolog
- Sum is not instantiated, reference in right side is undefined
- If Sum is instantiated, the clause fails because the left operand cannot have the current instantiation

Arithmetic Expressions

```
%% distance.pl
speed(ford, 100).
speed(chevy, 105).
speed(dodge, 95).
speed(volvo, 80).
time(ford, 20).
time(chevy, 21).
time(dodge, 24).
time(volvo, 24).
distance(X, Y) :- speed(X, Speed),
        time(X, Time),
        Y is Speed * Time.
```

Arithmetic Expressions

?- distance(chevy, Chevy_Distance).
– Chevy_Distance = 2205.
?- distance(volvo, Volvo_Distance).
– Volvo_Distance = 1920.

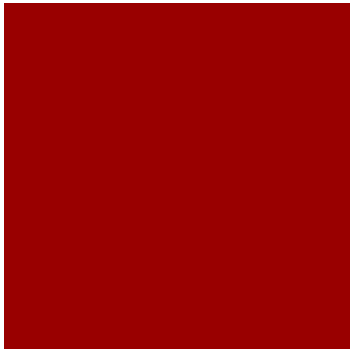
Debugging

- *Use the built-in structure trace to display instantiations of values at each step:*
  - *prolog distance.pl*
  - *?- trace.*
  - *?- distance(volvo, Volvo_Distance).*
- *Tracing model describe prolog programs in four events:*
  - *1) Call, attemps to satisfy a goal,*
  - *2) Exit, when a goal has been satisfied*
  - *3) Redo, when backtrack causes an attempt to resatisfy a goal*
  - *4) Fail, when a goal fails*

Debugging

```
trace, (distance(ford, Ford_Distance)).
        Call:distance(ford, _3964)
        Call:speed(ford, _4276)
        Exit:speed(ford, 100)
        Call:time(ford, _4276)
        Exit:time(ford, 20)
        Call: _3964 is 100*20
        Exit:2000 is 100*20
        Exit:distance(ford, 2000)
Ford_Distance = 2000
```

List Structures



- *Lists are sequences of any number of elements*
- *Lists can be composed by:*
  - *Atoms*
  - *Atomic prepositions*
  - *Any other terms, including lists*
- *Syntax:*
  - [apple, prune, grape, kumquat]
  - [] %% empty list
  - [X | Y] %% denotes head X and tail Y (car and cdr in LISP)

List Structures

- *Check if a term is a member of a list:*
  ?- member(a, [b, c, d]).
  – false.
  ?- member(b, [a, b, c]).
  – True.
- *member(X, L) :- ??*

List Structures

- *member(X, [X | _]).*
- *member(X, [_ | L]) :-*
  member(X, L).

Q&A