

Logic Programming

Section #1

Eman Safwat

10/10/2009

Prolog

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

There are two types of languages:

-**Procedural**: like C++, C#.

-**Declarative**: like Prolog.

- Prolog is **declarative**, which means the program logic is expressed in terms of **relations**, and execution is triggered by running queries over these relations.

البرولوج ليها Inference Engine وهو اى ال Compiler لى لغة تانية.

- Prolog rules are used for the knowledge representation, and the Prolog inference engine is used to derive conclusions.

ال Inference Engine ده بيعمل حاجتين Matching و Backtracking.

البرنامج بيتكون من حاجتين هما ال Facts وال Rules.

Facts (Relations) are like:

parent (ahmed, ali). \implies Ahmed is the parent of Ali.

age (ali, 4). \implies Ali's age is 4.

male (ali). \implies Ali is male.

team (ali, hossam, zaki). \implies Ali, Hossam, and Zaki are a team.

And so on...

لما بنيجى نكتب البرنامج بيكون عندنا كذا Region اول واحدة هيا ال Predicates بنكتب فيها ال Relations اللى بنعرفها فى البرنامج .

Predicates

non determ likes (symbol, symbol).

male (symbol).

sport (symbol).

Symbol ده object من نوع Symbol وممكن يبقى عندى object واحد أو اتنين أو أكثر فى ال Relation.
 ال Full Stop بتنهى كل جملة فى البرنامج زى ال Semi Colon (;) فى ال C++ ومن غيرها هيديك .error
 بعد كده Region ال Clauses بنكتب فيها ال Facts وال Rules بتاعت البرنامج.

- There are two types of relations, **determine** and **non determine** so write before **likes** "non determ", unless you'll get an error .. I don't remember why **it's** non besra7a :D.

Clauses

هنبدأ بال Facts:

sport (football).

sport (tennis).

sport (swimming).

male (ali).

male (hossam).

male (ahmed).

likes (ahmed, football).

likes (ahmed, icecream).

likes (hossam, swimming).

likes (ali, icecream).

likes (ahmed, tennis).

ودى ال Rule:

likes (ali, X) :-

likes (ahmed, X).

معناها:

-Ali likes X if Ahmed likes X.

X دى variable ممكن تبقى football أو أى حاجة، وزى ما ملاحظين X ده Upper-Case Letter

مع إن أحمد وعلى Lower- Case عادى!!

—أى حاجة Capital بيعتبرها ال Engine.. variable.. عشان كده بكتب أى اسم Small Letters.

- ويمكن ال Variable يبدأ ب (_) Underscore أو Capital Letter بس مينفعش بيتدى برقم
زى فى ال C++, C#.

كل حاجة فى ال Clauses زى بعد لازم تتكتب ورا بعد وإلا هيدى Error، لو لاحظنا هنلاقى likes ورا بعد.
ال :- دى معناها if، و

likes (ali, X).

ببسموها ال Head أو ال Conclusion اللي عاوزه أوصلها من ال Rule.

آخر Region فى البرنامج اللي هيا ال Goal وبكتب فيها ال Question أو الحاجة اللي عاوزه أشوف نتيجتها من
البرنامج.

Goal

- likes (ahmed, football).

Yes

ده Yes or No Question هيرجعلى Yes or No عن طريق انو يروح يعمل Checking لل Facts ازاي؟
هيعمل Matching بين كل كلمة والتانية الأول likes لو لاقاها بيخش ي match بين أحمد وال object اللي لاقاه
فى ال Facts لو صح يكمل وي match ال Object اللي بعده Football لو صح يرجع Yes غير كده No.
وهنا طبعا Yes.

- likes (ahmed, swimming).

No

- likes (ahmed, What).

هنا What Question عشان What هنا variable فهيرجع كل الحجات اللي أحمد likes سواء كانت

What أو X أو أى حاجة مادام بتبدأ ب Upper-Case letter.

What = football

What = icecream

What = tennis

- likes (X, Y).

هيطلعلى كل ال likes فى البرنامج عشان X, Y are two variables.

Ex:

-Ali likes X if Ahmed likes X and X is a sport.

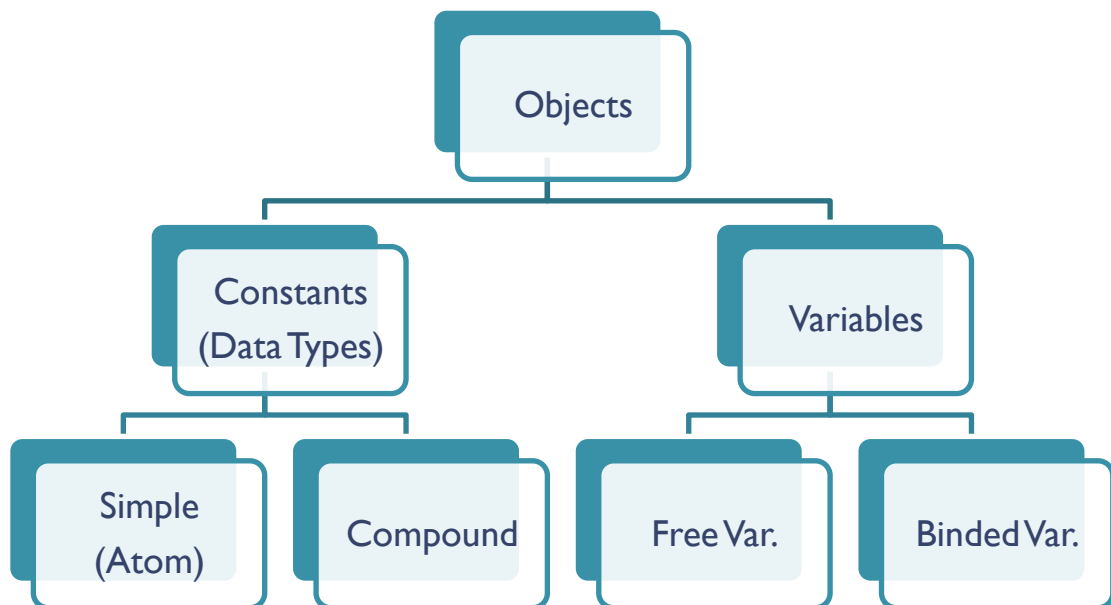
عاوزينها Rule ... ازاي؟

likes (ali, X) :-

likes (ahmed, X),

sport (X).

البرولوج بتتكون من Objects وال Objects نوعين:



Simple Data Types Like:

-integer.

-real (float, double).

-char.

-string (between quotes " ").

-symbol (starts with lowercase letter (a : z)).

Compound Data Types like:

- lists (arrays)
- compound objects.

Variables:

- Anything starts with an uppercase letter.

Free Variables:

- Empty variables (didn't have a value yet).

Binded Variables:

- Have a value.

Ex:

لو جيت كتبت في ال Predicates:

age (symbol, integer).

وفي ال Goal كتبت:

age(ahmed, X).

لازم هيرجع ال X ب integer.

ولو جيت كتبت مثلاً:

X = eman.

في أي حنة في البرنامج هيدى error، عشان X متعرف على انه integer.

Ex:

likes (ali, What).

هيروح ي Match في ال Clauses مش هيلاقى likes مع على في حاجة في ال Facts بس هيلاقى ال Rule

likes (ali, X) :-

likes (ahmed, X), sport (X).

هيشخ يحقق ال sub goal اللى هوا:

likes (ahmed, X), sport (X).

1. What = X

هيدور على likes (ahmed, X) فى ال Clauses:

Now el X kant free (makansh feha 7aga).

$X = \text{football} \implies (X \text{ ba2et binded})$

sport (X).

هيرجح على ال Clauses ي Match هيلاقى. sport (football) هيرجح True.

2. Backtrack

$\text{likes (ahmed, X)} \implies (X \text{ reg3et free})$

وهيعمل Matching تانى هيلاقى. likes (ahmed, icecream).

$X = \text{icecream} \implies (X \text{ ba2et binded})$

هيرجح False.

3. Backtrack

$\text{likes (ahmed, X)} \implies (X \text{ reg3et free})$

$X = \text{tennis} \implies (X \text{ ba2et binded})$

هيرجح True.

4. Backtrack

$\text{likes (ahmed, X)} \implies (X \text{ reg3et free})$

هيلاقى hossam مش هي Match هيرجح False.

5. Backtrack

هيرجح لل Rule .. likes (ali, X) مش هي Match هيرجح False.

Final Sol:

What = football

What = tennis.

Ex:

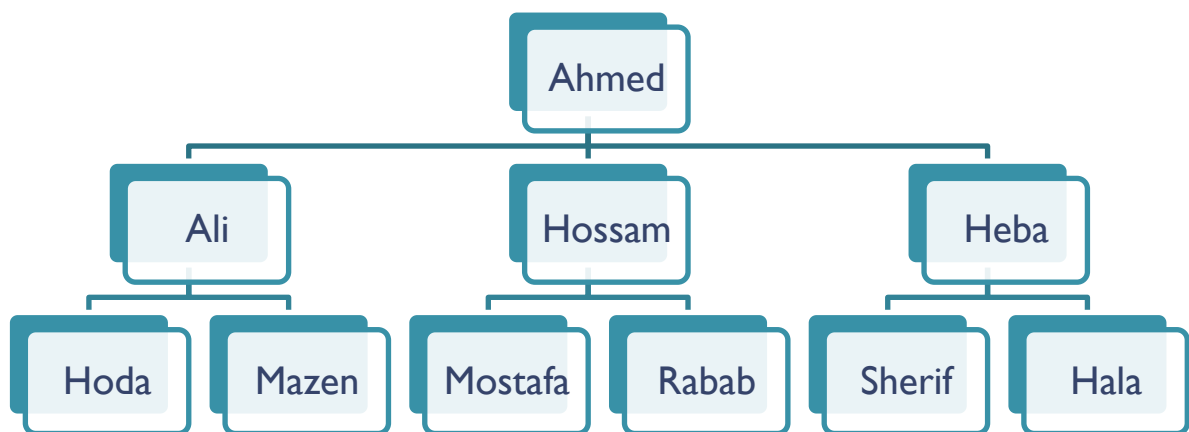
-Hatem likes any boy who likes football.

Rule??

likes (hatem, X) :-

likes (X, football), male (X).

آخر حاجة أخذنا ال Family Tree دى وفيها Assignment:



Predicates

parent (symbol, symbol).

male (symbol).

female (symbol).

father (symbol, symbol).

brother (symbol, symbol).

Clauses

male (ahmed).

male (ali).

male (hossam).

male (mazen).

male (mostafa).

male (sherif).

female (heba).

female (hoda).

female (rabab).

female (hala).

parent (ahmed, ali).

parent (ahmed, hossam).

parent (ahmed, heba).

parent (ali, hoda).

And so so... (el ba2y)

في السيكلشن جينا:

father (X, Y) :-

male (X), parent (X, Y).

-w mather nafs el fekra bas female.

brother (X, Y) :-

male (X), parent(Z, X), parent (Z, Y),

different (X, Y).

- أنا زودت هنا حتة ال Difference عشان ميرجعليش شخص أخو نفسه!!

Assignment:

grandfather??

aunt??

uncle??

cousin??

جربوا مع نفسكوا واتأكدوا على البرنامج.

See Also:

- [Prolog on Wiki](#)
- [Prolog Syntax](#)
- [Logic Programming](#)

Thanks 4 your patience, have fun .. cya next section.