南京大学
Nanjing University

Logic Programming

Facts

Example.

father, mother, male and female are predicates

father(terach, abraham).

father(terach, nachor).

father(terach, haran).

father(abraham, issac).

father(haran, lot).

father(haran, milcah).

father(haran, yiscah).

mother(sarah, issac).

male(terach).

male(abraham).

male(nachor).

male(haran).

male(issac).

male(lot).

female(sarah).

female(milcah).

female(yiscah).

Facts

Example.

$0 < 1$.

$0 < 2$.

$1 < 2$.

$0 < 3$.

$1 < 3$.

$2 < 3$.

Queries

father(terach, abraham).

father(terach, nachor).

father(terach, haran).

father(abraham, issac).

father(haran, lot).

father(haran, milcah).

father(haran, yiscah).

mother(sarah, issac).

male(terach).

male(abraham).

male(nachor).

male(haran).

male(issac).

male(lot).

female(sarah).

female(milcah).

female(yiscah).

Query: father(abraham, issac)? ✔

Queries

father(terach, abraham).

father(terach, nachor).

father(terach, haran).

father(abraham, issac).

father(haran, lot).

father(haran, milcah).

father(haran, yiscah).

mother(sarah, issac).

male(terach).

male(abraham).

male(nachor).

male(haran).

male(issac).

male(lot).

female(sarah).

female(milcah).

female(yiscah).

Query: female(abraham)? ✘

Queries

father(terach, abraham).

father(terach, nachor).

father(terach, haran).

father(abraham, issac).

father(haran, lot).

father(haran, milcah).

father(haran, yiscah).

mother(sarah, issac).

male(terach).

male(abraham).

male(nachor).

male(haran).

male(issac).

male(lot).

female(sarah).

female(milcah).

female(yiscah).

Query: father(abraham, X)? X=issac.

father(abraham, lot)?, father(abraham, milcah)?, ...,
father(abraham, issac)?, ...
stops until an answer yes is given.

Universal Facts

likes(abraham, pomegranates).

likes(sarah, pomegranates).

.....

likes(X, pomegranates).

Conjunctive Queries

father(terach, abraham).

father(terach, nachor).

father(terach, haran).

father(abraham, issac).

father(haran, lot).

father(haran, milcah).

father(haran, yiscah).

mother(sarah, issac).

male(terach).

male(abraham).

male(nachor).

male(haran).

male(issac).

male(lot).

female(sarah).

female(milcah).

female(yiscah).

Query: father(haran, X), male(X)? X=lot.

Is there an X such that both ... ?

Rules

Rules are statements of the form:

$$A \leftarrow B_1 \Lambda B_2 \Lambda \ldots \Lambda B_n.$$

In Prolog,

$$\mathrm{A} \text{ :- } \mathrm{B}_1, \mathrm{B}_2, \ldots, \mathrm{B}_\mathrm{n}.$$

Example.

son(X, Y) :- father(Y, X), male(X).

grandfather(X, Y) :- father(X, Z), father(Z, Y).

Logic Program

Facts $\mathcal{F}$, rules $\mathcal{R}$ and a query Q:

$$\mathcal{F}, \mathcal{R} \vDash Q \ ?$$

Is Q the logical consequence of $\mathcal{F}$ and $\mathcal{R}$?

Program:

father(abraham, issac).

father(haran, lot).

father(haran, milcah).

father(haran, yiscah).

male(issac).

male(lot).

female(milcah).

female(yiscah).

son(X, Y) :- father (Y, X), male(X).

daughter(X, Y) :- father(Y, X), female(X).

son(lot, haran)?

Recursive Programming

natural_number(X) : X is a natural number.

natural_number(0).

natural_number(s(X)) :- natural_number(X).

Example - Arithmetic

X ≤ Y : X and Y are natural numbers, s.t.,
X is less than or equal to Y.

0 ≤ X :- natural_number(X).

s(X) ≤ s(Y) :- X ≤ Y.

natural_number(0).

natural_number(s(X)) :- natural_number(X).

Example - Arithmetic

plus(X, Y, Z) : X, Y and Z are natural numbers, s.t.,
Z is the sum of X and Y.

```
plus(0, X, X) :- natural_number(X).

plus(s(X), Y, s(Z)) :- plus(X, Y, Z).

natural_number(0).

natural_number(s(X)) :- natural_number(X).
```

Example - List Concatenation

cat(X, Y, Z): X, Y and Z are lists, s.t.,
Z is the concatenation, i.e., Z = [X|Y].

```
cat([], Y, Y).

cat([V|X], Y, [V|Z]) :- cat(X, Y, Z).
```

Example - List Concatenation

cat(X, Y, Z): X, Y and Z are lists, s.t.,
Z is the concatenation, i.e., Z = [X|Y].

cat([], Y, Y).

cat([V|X], Y, [V|Z]) :- cat(X, Y, Z).

cat([1, 2], [3, 4], L)? L=[1, 2, 3, 4].

Example - List Concatenation

cat(X, Y, Z): X, Y and Z are lists, s.t.,
Z is the concatenation, i.e., Z = [X|Y].

$\mathcal{F}$: cat([], Y, Y).

$\mathcal{R}$: cat([V|X], Y, [V|Z]) :- cat(X, Y, Z).

Q: cat([1, 2], [3, 4], L)?

$\mathcal{F}, \mathcal{R} \vDash Q$?

cat([], [3,4], [3,4])

cat([2], [3, 4], [2, 3, 4])

cat([1, 2], [3, 4], [1, 2, 3, 4])

Search

- Idea: To solve a query Q,
  - If Q is a fact in the program, then done
  - Else, if Q :- Q_1,...,Q_n is a rule in the program, solve Q_1 ... Q_n.
  - Else, give up on this query.

"What" vs "How"

- Declarative vs Procedural Programming

- Procedural programming
  - The programmer has to specify how to get the output for the range of required inputs.
  - The programmer must know the appropriate algorithm.

- Declarative programming
  - Requires a more descriptive style.
  - The programmer must know what relationships hold between various entities.

Example - List Concatenation

In procedural style:

```
list procedure cat(list a,list b)
{
    list t = list u = copylist(a);
    while (t.tail != nil) t = t.tail;
    t.tail = b;
    return u;
}
```

In declarative style:

```
cat([], Y, Y).
cat([V|X], Y, [V|Z]) :- cat(X, Y, Z).
```

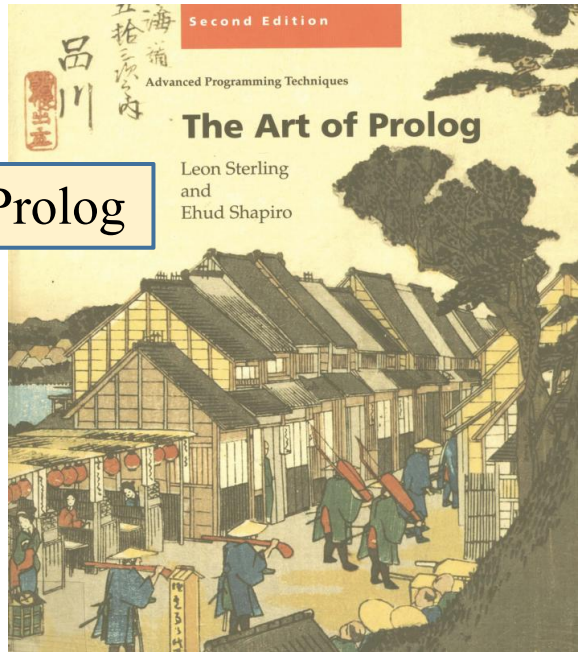
Logic Programming

- A declarative style programming paradigm.
- Computation through logical deduction.
- Uses the language of logic to express data and programs.
- Most of current LP languages use first-order logic (FOL).
- **Prolog** – the most popular logic programming language.

Logic Programming

- A declarative style programming paradigm.
- Computation through logical deduction.
- Uses the language of logic to express data and programs.
- Most of current LP languages use first order logic (FOL).
- **Prolog** – the most popular LP language.

The Art of Prolog



The Craft of Prolog