Lecture 17

GProlog
Introduction to Prolog

3. Other resources - Cygwin for windows: https://cygwin.com/install.html

Support Linux, Windows & Mac

Materials adapted from Geraint Wiggins
Prolog is a logic language that is particularly suited to programs that involve symbolic or non-numeric computation. For this reason it is a frequently used language in Artificial Intelligence where manipulation of symbols and inference about them is a common task.

Prolog consists of a series of **FACTS** and **RULES**. A program is run by presenting some query and seeing if this can be proved against these known rules and facts.

- If the train arrives late and there are no taxis at the station then John is late for his meeting. John is not late for his meeting. The train did arrive late.
Fact represents a relation between items; facts are stored in a database

Two ways to load a database:

1) ?- [user].
   ...
   Ctrl+D

2) ?- [custom_file].
Introduction to Prolog

Fact represents a relation between items; facts are stored in a database

sunny.
raining(today).

Facts have some simple rules of syntax. Facts should always begin with a lowercase letter and end with a full stop. The facts themselves can consist of any letter or number combination, as well as the underscore _ character. However, names containing the characters -,+,*,/ or other mathematical operators should be avoided.

Query:

sunny.
raining(yesterday).
snowing(yesterday).

Hazlenuts.
tomsRedCar.
2Ideas.

Prolog only knows about relations that you tell it about.

Materials adapted from Geraint Wiggins
Variable

eats(fred, mangoes).

**Queries: What does fred eat?**

We use variables to achieve this -- variables are distinguished by starting with a capital letter

eats(fred, What). or eats(fred, X).

loves(john, mary).
loves(fred, hobbies).

?- loves(john, Who). /* Who does john love? */

?- loves(arnold, Who) /* who does arnold love */

*Materials adapted from Geraint Wiggins*
Unification

eats(fred,tomatoes)
eats(Whom,What)

eats(fred,Food)
eats(Person,jim)

cd(29,beatles,sgt_pepper).
cd(A,B,help).

f(X,a).
f(a,X).

likes(jane,X).
likes(X,jim).

f(X,Y).
f(P,P).

f(foo,L).
f(A1,A1).

Materials adapted from Geraint Wiggins
"All men are mortal"

\[ \text{mortal}(X) :\text{-} \text{human}(X). \]

1) Declarative interpretation is "For a given X, X is mortal if X is human.

2) Procedural interpretation is "To prove the main goal that X is mortal, prove the subgoal that X is human."

\[ \text{mortal}(X) :\text{-} \text{human}(X). \]
\[ \text{human(socrates).} \]
\[ \text{mortal(socrates).} \]

?- \text{mortal(P)}.

\textbf{Materials adapted from Geraint Wiggins}
Rule

“Something is fun if its a red car or a blue bike or it is ice cream”

fun(X) :- /* an item is fun if */
    red(X), /* the item is red */
    car(X). /* and it is a car */

fun(X) :- /* or an item is fun if */
    blue(X), /* the item is blue */
    bike(X). /* and it is a bike */

fun(ice_cream). /* and ice cream is also fun. */

Materials adapted from Geraint Wiggins
fun(X) :-
    red(X),
    car(X).

fun(X) :-
    blue(X),
    bike(X).

car(vw_beatle).

car(ford_escort).

bike(harley_davidson).

red(vw_beatle).

red(ford_escort).

blue(harley_davidson).

?- fun(harley_davidson). /* to which Prolog will reply */