CMPS 3500

Programming Languages

Dr. Chengwei Lei
CEECS
California State University, Bakersfield
Chapter 16

Logic Programming Languages
Facts with Arguments Examples 1

- eats(fred, oranges). /* "Fred eats oranges" */
- eats(fred, t_bone_steaks). /* "Fred eats T-bone steaks" */
- eats(tony, apples). /* "Tony eats apples" */
- eats(john, apples). /* "John eats apples" */
- eats(john, grapefruit). /* "John eats grapefruit" */

?- eats(fred, oranges). /* does this match anything in the database? */
  yes /* yes, matches the first clause in the database */
?- eats(john, apples). /* do we have a fact that says john eats apples? */
  yes /* yes we do, clause 4 of our eats database */
?- eats(mike, apples). /* how about this query, does mike eat apples */
  no /* not according to the above database. */
?- eats(fred, apples). /* does fred eat apples */
  no /* again no, we don't know whether fred eats apples */
- eats(fred,mangoes).

- ?- eats(fred,what).
- eats(fred,mangoes).

- 

- ?- eats(fred,What).
- eats(fred,mangoes).

- ?- eats(fred,what).

- ?- eats(fred,What).
  - As a result of this query, the variable What has matched (or unified) with mangoes. We say that the variable What now has the binding mangoes. When we pose a query, if the query is successful, Prolog prints both the variable and the variable name, as we see above.
loves(john,mary).
loves(fred,hobbies).

?- loves(john,Who). /* Who does john love? */
   Who=mary /* yes, Who gets bound to mary */
   yes /* and the query succeeds */
?- loves(arnold,Who) /* does arnold love anybody */
   no /* no, arnold doesn’t match john or fred */
?- loves(fred,Who). /* Who does fred love */
   Who = hobbies /* Note the to Prolog Who is just the name of a variable, it */
   yes /* semantic connotations are not picked up, hence Who unifies */
tape(1,van_morrison,astral_weeks,madam_george).
tape(2,beatles,sgt_pepper,a_day_in_the_life).
tape(3,beatles,abbey_road,something).
tape(4,rolling_stones,sticky_fingers,brown_sugar).
tape(5,eagles,hotel_california,new_kid_in_town).

?- tape(5,Artist,Album,Fave_Song). /* what are the contents of tape 5 */
  Artist=eagles
  Album=hotel_california
  Fave_Song=new_kid_in_town
  yes

?- tape(4,rolling_stones,sticky_fingers,Song). /* find just song */
  Song=brown_sugar /* which you like best from the album */
  yes
- mortal(X) :- human(X).
- human(socrates).

?- mortal(socrates).
  - Yes

?- mortal(P).
  - P = socrates
  - yes
For example, How can we represent the sentence 'Something is fun if its a red car or a blue bike or it is ice cream'?
For example, How can we represent the sentence '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 */
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 */

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 */
  yes /* to show the program succeeded */
?- fun(harley_davidson). /* to which Prolog will reply */
  yes /* to show the program succeeded */

?- fun(What).
  What=vw_beatle
  yes