KRE/t5 Alan Smaill
13 Feb 2006

# Tutorial Exercises for sixth week

1. Which words in the following are being *used* and which are *mentioned*?

   > The artist formerly known as the artist formerly known as prince
   > is now known as the artist.

2. (For those who know Prolog) Which terms in the following logic program are being used, and which are being mentioned?

```
flatten(X,[X]) :- var(X),!.
flatten([],[]) :- !.
flatten([H|T],L3) :-
        !,
        flatten(H,L1),
        flatten(T,L2),
        append(L1,L2,L3).
flatten(X,[X]).
```

3. Suppose a `clause/2` predicate is available such that a call of the form `clause( H, B)` (with `H` instantiated) unifies `B` with the literals that form the body of an appropriate clause (for a unit clause, `B` is just `true`). The *vanilla meta-interpreter* has the logical reading as follows.

```
solve( true ).
solve( (A,B) ) <-  solve(A) /\ solve(B).
solve( A )     <-  clause(A,B) /\ solve(B).
```

   Give a logical description that counts the depth in the search space at which a solution to a query is found, ie the length of the longest chain of uses of the third statement above needed to establish the solution.

   Prolog users may note note that Prolog search may investigate longer unsuccessful branches before finding the first solution.

4. (For Prolog users) Consider the following meta-interpreter, intended to search at random for Prolog derivations. What outputs would you expect for successive calls to `ran_solve(member(X,[1,2,3,4]))`, with the usual member program?

```
% non-deterministic interpreter
% does no back_tracking
ran_solve( true ) :- !.
ran_solve( (A,B) ) :- !, ran_solve(A), ran_solve(B).
ran_solve( HH ) :-  random_clause( HH, B ),
                    ran_solve(B).
random_clause( H, B ) :- setof( (H,B), clause(H,B), S),
                         random_pick( (H,B), S).
random_pick(X,List) :- length(List,N), rp(X,List,N).
rp(X,[X],_) :- !.
rp(X,List,N) :- random(N,Nth), nth_member(Nth,X,List).
% random(+,-) returns a random number in the range 0 to N-1
nth_member(1,H,[H|_]).
nth_member(N,X,[_|T]) :- M is N-1, nth_member(M,X,T).
```

Can you think why someone might want to use such an interpreter?