Examen Prolog — ESSI2

Michel Rueher

12 Janvier 1996

Documents autorisés: *supports (et notes) de cours ou TD*

1 Transformation d'une forme clausale en Prolog — 5 points

Soit les énoncés :
- Pour tout crime, il y a quelqu'un qui l'a commis;
- Seuls les gens malhonnêtes commettent des crimes;
- Ne sont arrêtés que les gens malhonnêtes;
- Les gens malhonnêtes arrêtés ne commettent pas de crimes;
- Il y a des crimes.

On cherche à étabir la proposition "*Il y a des gens malhonnêtes non arrêtés*".

(1) Mettre les énoncés et la proposition ci-dessus sous forme de clauses. On utilisera pour cela les prédicats suivants:
  - `ar(y)` : y est arrêté;
  - `ma(y)` : y est malhonnête;
  - `co(y,x)` : y commet x;
  - `cr(x)` : x est un crime.

(2) Peut on prouver ou refuter la proposition à l'aide d'un interprète Prolog en:
  (a) Transformant automatiquement les clauses obtenues en un programme Prolog;
  (b) Utilisant la clause correspondant à la proposition à établir comme résolvante.

2 Stratégie de recherche de Prolog — 5 points

(1) Est ce que la stratégie de recherche standard de Prolog garantit la correction et la complétude des résultats ?
Pourquoi ?

(2) Quelles autres stratégies de recherche pourraient être utilisées par un interprète Prolog?
Expliquer quels sont leurs avantages et leurs inconvénients.

3 Programmez .... — 5 points

Ecrire un prédicat Prolog `liaisons(A,B,C,T)` qui recherche l'ensemble des liaisons aériennes entre les villes A et B avec un coût inférieur à T. Le chemin utilisé est donné par C. L'existence d'un vol entre les villes A et B avec un coût de C est exprimé par un prédicat `vol(A,B,C)`; le coût associé à une escale de 1.

Exemple

```
vol(a,b,5).        vol(b,c,3).        vol(a,c,6)..
vol(c,a,9).        vol(b,a,1).        vol(c,d,5).
| ?- liaisons(a,d,C,19).              | ?- liaisons(a,d,C,15)..              | ?- liaisons(a,d,C,12).

C = [a,b,c,d] ? ;                     C = [a,c,d] ? ;                      no

C = [a,c,d] ? ;

no                                    no
```

4 Expliquez le programme mybagof/3 — 5 points

(1) Expliquer le rôle du fail dans la première clause du prédicat add_mylist/1 dans le programme bagof;

(2) Expliquer le fonctionnement de la deuxième clause du prédicat add_mylist/1 dans le programme bagof.

Rappel du programme bagof/3

```
mybagof(O,P,L):-
       P,
       add_mylist(O),
       fail.

mybagof(_,_,L) :- !,
       retract(mylist(L-[])).

add_mylist(O) :-
       \+(clause(mylist(_),_)), ! ,
       L = [O|S],
       assert(mylist(L-S)).
add_mylist(O) :-
       retract(mylist(L-S)),
       S=[O|S1],
       assert(mylist(L-S1)).


mysetof(O,P,L) :-
       mybagof(O,P,L1),
       sort(L1,L).
```