Interrogation écrite – Prolog – 10 mars 2020 – durée : 1h – Sans documents. CORRIGE

Exercice 1.

Est-ce que chaque groupe d'expressions suivant est unifiable ?
Si oui, donnez la substitution la plus générale (mgu). Sinon donnez la raison pour laquelle les deux termes ne s'unifient pas.
Dans ces expressions, les lettres en majuscules représentent des variables, et celles en minuscule sont des constantes. On se place dans une unification avec test d'occurrence.

1. p(a) et p(X)

Les deux termes s'unifient. Mgu = X=a.

2. p(f(Z),Z) et p(X,X)

Les deux termes ne s'unifient pas à cause du test d'occurrence (occur-check). Il faudrait pouvoir unifier Z et f(Z).

3. p(Y,c) et p(d,X)

Les deux termes s'unifient. Mgu = Y=d, X=c.

4. p(f(X), g(Y)), p(a,b)

Les deux termes ne s'unifient pas.

5. p(X, f(X)), p(a, Z)

Les deux termes s'unifient. Mgu = X=a, Z = f(a).

6. p([X,Y],[X|Y]) et p(Z,Z).

Les deux termes ne s'unifient pas (occur-check). Il faudrait pouvoir unifier Y et [Y].
En Prolog :

```
?- p([X,Y],[X|Y])= p(Z,Z).
Y = [Y],
Z = [X, Y].

?- unify_with_occurs_check(p([X,Y],[X|Y]), p(Z,Z)).
false.
```

Exercice 2.

a. Ecrire en Prolog le prédicat inclus/2 défini par :
inclus(L1,L2) est vrai si tous les éléments de la liste L1 appartiennent à la liste L2.

```
mem(X, [X|_]).
mem(X, [_|L]) :- mem(X, L).

inclus([], _).
inclus([X| L], M):- mem(X, M), inclus(L, M).
```

b. Ecrire en Prolog le prédicat remplacer/4 défini par : remplacer(X,Y, L, T) vrai si T est la liste obtenue en remplaçant dans L toutes les occurrences de X par Y.

```
remplacer(X, Y, [], []).
remplacer(X, Y, [X|L], [Y|P]):- remplacer(X, Y, L, P).
remplacer(X, Y, [Z|L], [Z|P]):- X \= Z, remplacer(X, Y, L, P).

?- remplacer(a, b, [1,a, a, 3], L).
L = [1, b, b, 3] ;
false.

?- remplacer(a, b, L, [1,b,b,3]).
L = [1, a, a, 3] ;
L = [1, a, b, 3] ;
L = [1, b, a, 3] ;
L = [1, b, b, 3] ;
false.

?- remplacer(a, X, [1,a, a, 3],[1,b,b,3]).
X = b ;
false.
```

c. Ecrire en Prolog le prédicat somlist/3 défini par : somlist(L1, L2, S) vrai si L1 est une sous-liste d'éléments de L2 (dans le même ordre, mais non nécessairement consécutifs) dont la somme fait S. La somme des éléments de la liste vide sera égale à 0. Par exemple,

```
?- somlist(L, [1,2,3,4], 4).
L = [1, 3] ;
L = [1, 3] ;
L = [4] ;
false.
```

```
somlist([], _, 0).
somlist([X|L1], [X|L2], S):- S1 is S-X, somlist(L1, L2, S1).
somlist(L1, [_|L2], S):- somlist(L1, L2, S).
```

Exercice 4.

Donner une preuve par réfutation pour l'ensemble de clauses suivant (vous indiquerez comment chaque nouvelle clause est obtenue en utilisant res(i,j)).

1. $\neg e(X) \vee v(X) \vee s(X, f(X))$
2. $\neg e(X) \vee v(X) \vee c(f(X))$
3. $p(a)$
4. $e(a)$
5. $\neg s(a, Y) \vee p(Y)$
6. $\neg p(Z) \vee \neg v(Z)$
7. $\neg p(W) \vee \neg c(W)$

Preuve par résolution :

8. res(3,6) : not(v(a))
9. res(2,4) : v(a) or c(f(a))
10. res(8,9) : c(f(a))
11. res(1,4) : v(a) or s(a, f(a))
12. res(11, 8) : s(a, f(a))
13. res(12,5) : p(f(a))
14. res(13,7) : not(c(f(a))

15. res(10, 15) : clause vide.

Exercice 5 (emprunté à L. Gacogne)
L'inspecteur Maigret veut connaître les suspects qu'il doit interroger pour un certain nombre d'affaires : il tient un individu pour suspect dès qu'il était présent dans un lieu un jour où un vol a été commis et s'il a pu voler la victime.
Un individu a pu voler, s'il était sans argent ou par jalousie.

On dispose des faits suivants :
Marie a été volée lundi à l'hippodrome, Jean a été volé mardi au bar, Luc a été volé jeudi au stade,
Max est sans argent,
Eve est très jalouse de Marie,
Max était présent au bar mercredi, Eric au bar mardi et Eve à l'hippodrome lundi.

On ne prend pas en compte la présence des victimes comme possibilité qu'elles aient été aussi voleurs ce jour-là.

1. En utilisant les prédicats present/3, vol/3, sansargent/1 et jaloux/2, écrire les faits connus.
2. Définir, en Prolog, les prédicats apuvoler/2 et suspect/1.
3. Quelles sont toutes les réponses renvoyées par Prolog à la requête suspect(X) ?
4. Représentez l'arbre de résolution de Prolog.

```
vol(bar, mardi, jean).
vol(stade, jeudi,luc).
vol(hipp, lundi, marie).
sansargent(max).
jaloux(eve, marie).
present(max, bar, mercredi).
present(eric, bar, mardi).
present(eve, hipp, lundi).

suspect(X) :- present(X, L, J), vol(L, J, V), apuvoler(X, V).
apuvoler(X, _) :- sansargent(X).
apuvoler(X, Y) :- jaloux(X, Y).

?- suspect(X).
X = eve.
```