

Corrigé de l'examen en Programmation logique

Questions de compréhension (5 points): Validez par Vrai ou Faux les assertions suivantes en corrigeant les réponses fausses (-0.5 pt pour une réponse fausse)

1. un fait Prolog est une clause de Horn positive. **Vraie**
2. Dans la formule : $\forall y p(y) \rightarrow (\exists y p(y) \vee \forall x p(x))$, la variable x nécessite un renommage. **Fausse, c'est la variable Y qui doit être renommée car elle est quantifiée deux fois (par \forall et \exists)**
3. l'UPG des $T1=f(x,g(h(z),x),y)$ et $T2= f(k(z),y,g(u, z))$ est $\{ x \backslash k(z), y \backslash g(h(z),k(z)), u \backslash h(z), z \backslash k(z) \}$. **Fausse, car en appliquant l'algorithme d'unification on obtient Z qui appartient à k(Z)à, ce qui déclenche un échec, de ce fait, T1 et T2 ne sont pas unifiables.**
4. La requête `display(2+3)` renvoie 5. **Fausse, la requête renvoie +(2,3)**
5. `Op(231 , yfx, toto)` signifie que l'opérateur toto est associatif à gauche. **Vraie**

Exercice 1 (5 points): 1- Les étudiants qui ont la grippe doivent prendre un congé de maladie. Les étudiants qui ont de la fièvre et qui toussent ont la grippe. Ceux qui ont une température supérieure à 38° ont de la fièvre. Mohammed tousse et a une température supérieure à 38°.

```

conge(X) :-grippe(X).
fièvre(X) :- temp(X,T), sup(T,38).
grippe(X) :-fièvre(X),tousse(X).
tousse(mohamed).
temp(mohamed, t).
sup(t,38).

```

2- La problématique à résoudre : Est-ce que Mohamed doit prendre un congé de maladie ? **Conge(mohamed).**

Exercice 2(5 points): Soit le programme Prolog suivant :

```

p(1).
p(2) :- !.
p(3).

```

R1 : `p(X).` **1 ;2**

R2 : `q(X).` **echec**

R3 : `p(X),p(Y).` **$X = 1 Y = 1 ? ; X = 1 Y = 2 ? ; X = 2 Y = 1 ? ; X = 2 Y = 2$**

R4 : `p(X,X).` **echec**

R5 : `p(X),!,p(Y).` **$X = 1 Y = 1 ? ; X = 1 Y = 2$**

Exercice 3 (5 points): Codez les clauses suivantes

```

dernier(X,[X]).
dernier(X,[_|Q]) :- dernier(X,Q).
absolue(X,X) :-X>= 0.
absolue(X,Y) :-X< 0, Y is -X.

```

```
occurrences(X,[], 0)
occurrences(X,[X|Q], N) :- occurrences(X,Q, M), N is M+1.
occurrences(X,[F|Q], N) :- X\==F, occurrences(X,Q, N) .
fact(0,1).
fact(X,F) :-X>0 , P is X-1, fact(P,V), F is X*V.
```