

## TD5 – Prolog

### Suppléments

Blandine Ginon, Marie Lefevre, Lucie Martinet

#### PARTIE 1 – TRADUCTION D'ÉNONCÉ

Soit l'énoncé suivant :

*Adam aime les pommes. Clara aime les carottes. Olivier aime les oranges. Les pommes sont des fruits. Les oranges sont des fruits. Les carottes sont des légumes. Ceux qui aiment les fruits sont en bonne santé.*

**Question 8 :** Formalisez ces faits et règles en PROLOG.

**Question 9 :** Quelle est la requête pour savoir qui est en bonne santé ?

**Question 10 :** Quelle est la requête pour "Qui aime les pommes?" ?

**Question 11 :** Comment savoir les fruits que connaît le programme ?

**Question 12 :** Donnez des phrases en français pour les faits et règles suivants:

*homme(sorate)*

*mortel(X) :- homme(X).*

*animal(X) :- chien(X).*

*ornithorynque(X) :- mammifere(X),ovipare(X),amphibie(X).*

**PARTIE 2 – MANIPULATION DE LISTES**

**Question 13 :** Définir le prédicat `longueur(L,N)`, qui étant donnée la liste `L` calcule sa longueur `N`.

**Question 14 :** Définir le prédicat `concat(L1,L2,L3)` où `L3` est le résultat de la concaténation de `L1` et `L2` (sans utiliser `append`).

**Question 15 :** Définir le prédicat `palindrome(L)` qui retourne vrai si la liste `L` est sa propre image renversée.

**Question 16 :** Définir un prédicat `rang_pair(X,Y)` qui extrait les éléments de la liste `X` qui ont des indices de rang pair afin de construire la liste `Y`.

Ex. `rang_pair([a,b,c,d,e],L)` .  $\rightarrow$  `L=[b,d]`

**Question 17 :** Définir le prédicat `indice(X,L,N)`, qui étant donné un élément `X` et une liste `L`, `X` appartenant à `L`, calcule `N` l'indice de la première occurrence de `X` dans `L`. Peut-on utiliser ce prédicat pour formuler une requête permettant de calculer le *i*ème élément d'une liste ?

**Question 18 :** Écrire le prédicat `remplace(X1,X2,L1,L2)` qui construit la liste `L2` qui est la liste `L1` dans laquelle `X1` est remplacé par `X2`.

**Question 29 :** Définir le prédicat `partage(L,X,L1,L2)`, qui étant donné une liste de nombre `L` et un nombre `X`, calcule la liste `L1` des nombres de `L` inférieurs à `X`, et la liste `L2` des nombres de `L` supérieurs ou égaux à `X`.