Série de TP N°3 IA : Les listes en PROLOG

Avant-propos

- En prolog si nous voulons afficher la liste des réponses directement on utilise l'instruction : `?- findall(X, parent(X, paul), Liste).`
- L'instruction *write*() affiche sur l'écran ce qui est entre parenthèses, elle est applicable à une constante, une variable ou une chaîne de caractères entre guillemets, l'instruction *nl* permet le passage à la ligne.
- En prolog, on ne définit pas de fonction ou de procédures mais des prédicats (à valeur dans 0 , 1), éventuellement avec 0 argument.

```
affiche :- write('bonjour').
```

- La négation d'un prédicat peut être défini à l'aide du prédicat not.
- Les tableaux sont **des listes en PROLOG**. Contrairement à la plupart des langages de programmation, les indices des éléments ne sont pas disponibles. En revanche, une liste L peut toujours être décomposée en $L = [E|R]$ où E est le premier élément de la liste (E n'est pas une liste) et où R est le reste de la liste L (R est une liste : c'est en fait la tranche de L qui démarre apres E). La liste vide est [].
  Exemple : [0,1,2,3] = [0 | [1,2,3]] = [0 | [1 | [2,3]]] et ainsi de suite.

Exercice 1 Quelles sont les réponses de Prolog aux buts suivants (à exécuter sur machine).

```
?- [X,a|Y] = [c,a,a|K].

?- [X,a|Y] = [c,c,a|K].

?- [1|X] = [1,2,3,5].

?- [1,2|X] = [1,2].

?- [1,2|X] = [1,2,7,3,4].

?- [X] = [ ].

?- [X] = [1,2].

?- [X,Y/Z] = [1,2].
```

Exercice 2 Ecrire un programme PROLOG pour :

1. Calculer la longueur d'une liste
2. Afficher le dernier élément d'une liste
3. Inverser une liste d'entiers
4. Afficher le nième élément d'une liste
5. Supprimer le dernier élément d'une liste
6. Concaténer deux listes d'entiers

Exercice 3 Ecrire un programme Prolog permettant de trier une liste d'entiers de façon ascendante puis descendante.