

Exercices sur les listes

Remarque: Parallèle avec d'autres langages de programmation.

Les tableaux en prolog sont des listes. 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 et où R est le reste de la liste L (R est une liste : c'est en fait la tranche de L qui démarre après E). Les éléments d'une liste peuvent être séparés par des virgule: [1,2,3]. La liste vide est [].

La liste [1,2,3] peut s'écrire également [1|[2,3]] ou [1,2,3|[]] ou encore [1,2|[3|[]]] .

1. affiche(L) est vrai si tous les éléments de la liste L sont écrits.
2. affiche2(L) est vrai si tous les éléments de la liste L sont écrits en ordre inverse.
3. Retrouver le premier élément d'une liste : premier1(L,X) est vrai si X est le premier élément de L.
4. Afficher le premier élément d'une liste (bis) : premier2(L) est vrai si le premier élément de la liste L est affiché (et aucun autre).
5. Retrouver le dernier élément d'une liste : dernier1(L,X) est vrai si X est le dernier élément de L.
6. Afficher le dernier élément d'une liste (bis) : dernier2(L) est vrai si le dernier élément de la liste L est affiché (et aucun autre).
7. element(X,L) est vrai si X est élément de la liste L.
Remarque: On peut l'utiliser sous la forme element(2,[1,3,5,8]) mais aussi element(X,[1,3,5,8]).
Remarque: En fait il existe un prédicat prédéfini en Prolog qui fait exactement la même chose : member(X,L), vrai si X est dans la liste L.
8. compte(L,N) est vrai si N est le nombre d'éléments dans la liste L.
9. somme(L,N) est vrai si N est la somme des éléments de la liste d'entiers L.
Remarque: Attention, un prédicat somme(...) a déjà été défini dans le TD n°2. Si vous rangez toutes vos définitions dans le même fichier cela va entraîner des conflits. Dans ce cas il faut en renommer un des deux.
10. nieme(N,L,X) est vrai si X est le N-ème élément de la liste L.
11. occurrence(L,X,N) est vrai si N est le nombre de fois où X est présent dans la liste L.
12. sous-ensemble(L1,L2) est vrai si tous les éléments de la liste L1 font partie de la liste L2.
13. substitue(X,Y,L1,L2) est vrai si L2 est le résultat du remplacement de X par Y dans L1.
14. retourne(L,L1) est vrai si la liste L1 est la liste L dans l'ordre inverse.

15. aplatir(LL, L) est vrai si L est la liste LL aplatie.

16. Arithmétique des grands nombre :

On implémente des nombres entiers qui contiennent des dizaines voire des centaines de chiffres avec des listes de chiffres. Ecrire les prédicats de somme, soustraction et multiplication de tels "nombres" ainsi que la relation d'ordre.