

UE ELFE - Expression Logique et Fonctionnelle ... Évidemment

TD n°4

Exercice 1 :

Soit la base de faits suivante :

```
enfant_de(anne,bridget).  
enfant_de(bridget,caroline).  
enfant_de(caroline,donna).
```

pour chacune des définitions suivantes du prédicat `descendant_de`, donner l'arbre de résolution Prolog de la requête `?- descendant_de(anne,X).`

Q 1.

```
descendant_de(X,Y) :- enfant_de(X,Y).  
descendant_de(X,Y) :- enfant_de(X,Z), descendant_de(Z,Y).
```

Q 2.

```
descendant_de(X,Y) :- enfant_de(X,Z), descendant_de(Z,Y).  
descendant_de(X,Y) :- enfant_de(X,Y).
```

Q 3.

```
descendant_de(X,Y) :- descendant_de(Z,Y), enfant_de(X,Z).  
descendant_de(X,Y) :- enfant_de(X,Y).
```

Q 4.

```
descendant_de(X,Y) :- enfant_de(X,Y).  
descendant_de(X,Y) :- descendant_de(Z,Y), enfant_de(X,Z).
```

Exercice 2 :

Une liste ℓ est une structure définie récursivement par :

- soit ℓ est la liste `vide` ;
- soit ℓ est une paire formée d'un terme quelconque et d'une autre liste.

Dans cet exercice, les listes seront donc représentées par des termes définis inductivement par

- la constante `vide` pour la liste vide,
- et le foncteur `cons` d'arité 2 pour les listes non vides.

Q 1. En vous inspirant de la définition des entiers de Peano, définir le prédicat `liste/1`, satisfait pour tous les termes représentant des listes.

Q 2. Donner l'arbre de résolution de la requête `?- liste(X).`

Q 3. Définir les prédicats `head/2` et `tail/2` satisfaits lorsque leur premier argument est la tête ou respectivement le reste de la liste donnée en deuxième argument.

Q 4. Définir le prédicat `membre/2` satisfait lorsque le terme donné en premier argument est membre de la liste donnée en second argument.

Q 5. Donner l'arbre de résolution de la requête `?- membre(b, cons(a, cons(b, vide))).`

Q 6. Définir le prédicat `last/2` satisfait lorsque le terme donné en premier argument est le dernier élément de la liste donnée en second argument.

Q 7. Donner l'arbre de résolution de la requête `?- last(X, cons(a, cons(b, vide))).`

Exercice 3 :

Un arbre binaire a est une structure définie récursivement par :

- soit a est un arbre vide ;
- soit a est un triplet formé d'un terme quelconque, et de deux arbres.

Dans cet exercice, les arbres binaires seront représentés par des termes définis inductivement à l'aide

- de la constante `vide`
- et du foncteur d'arité 3 `noeud`.

Q 1 . Définir le prédicat `arbre/1`.

Q 2 . Définir un prédicat `reflet/2` satisfait lorsque les deux termes donnés en arguments sont des arbres binaires reflets l'un de l'autre. (Deux arbres binaires sont *reflets* l'un de l'autre si leurs représentations graphiques sont symétriques par rapport à un axe vertical).

Q 3 . Écrire un prédicat `membre/2` satisfait lorsque le terme donné en premier argument est dans l'arbre binaire donné en second argument. dans un arbre.