

# TP DE PROGRAMMATION LOGIQUE EN PROLOG

## PARTIE 3 : EXERCICES PRATIQUES

### Exercice 1 - Ajout de nouveaux faits et requêtes

Questions :

1. Qui est le père de Marc ?

```
?- pere(X, marc).  
X = jacques.
```

2. Marc a-t-il des enfants ?

```
?- pere(marc, X)  
|  
.  
X = sophie.
```

### Exercice 2 - Règle pour les Grands-parents

Ajouter la règle :

```
grand_parent(X, Y) :- parent(X, Z), parent(Z, Y).
```

Questions :

1. Qui est le grand-parent de Paul ?

```
?- grand_parent(X, marc).  
false.
```

(car nous n'avons pas défini les parents de Pierre ou Marie)

2. Jacques est-il grand-parent de Sophie ?

```
?- grand_parent(X, sophie)  
|  
.  
X = jacques ■
```

### Exercice 3 - Règle pour les Frères et Sœurs

Ajouter la règle :

```
frere_soeur(X, Y) :- parent(Z, X), parent(Z, Y), X \= Y.
```

Question :

1. Paul a-t-il des frères ou des sœurs ?

```
?- frere_soeur(paul, X).  
false.
```

### Exercice 4 - Requêtes avec Variables

1. Trouvez tous les hommes dans la base de données :

```
?- homme(X).  
X = pierre ;  
X = marc ;  
X = paul ;  
X = jacques.
```

2. Recherchez tous les parents de Sophie :

```
?- parent(X,sophie).  
X = marc ;  
X = julie.
```

## PARTIE 4 : MANIPULATION DES LISTES

### Exercice 5 - Manipuler des Listes

1. Déclarez une liste :

```
?- L = [pierre, marie, paul, sophie].  
L = [pierre, marie, paul, sophie].
```

Décomposer une liste :

```
?- [Tete | Queue] = [pierre, marie, paul].  
Tete = pierre,  
Queue = [marie, paul].
```

### Exercice 6 - Longueur d'une Liste

Ajouter les règles :

```
longueur([], 0).  
longueur([_|Queue], N) :- longueur(Queue, M), N is M + 1.
```

Test :

```
?- longueur([pierre, marie, paul], N).  
N = 3.
```

### Exercice 7 - Trouver un Élément dans une Liste

Ajouter les règles :

```
membre(X, [X|_]).  
membre(X, [_|Queue]) :- membre(X, Queue).
```

Test :

```
?- membre(marie,[pierre,marie,paul]).  
true ■
```

## PARTIE 5 : DÉFIS SUPPLÉMENTAIRES

### Exercice 8 - Règle pour les Oncles et Tantes

Ajouter la règle :

```
oncle_tante(X, Y) :- parent(Z, Y), frere_soeur(X, Z).
```

Questions :

1. Marc est-il l'oncle de Paul ?

```
?- oncle_tante(marc,paul).  
false.
```

2. Quels sont les oncles de Sophie ?

```
?- oncle_tante(X,sophie).  
false.
```

## Exercice 9 - Arbre Généalogique Étendu

Ajouter la règle pour les cousins :

```
cousin(X, Y) :- parent(Z, X), parent(W, Y), frere_soeur(Z, W).
```

j'ai aussi rajouté la relation : `parent(jacques, pierre)`. Car Pierre

Test :

```
?- cousin(paul,X).  
X = sophie ;  
false.  
  
?- cousin(paul,sophie).  
true
```