



Analyse d'un programme PROLOG

Soient les clauses suivantes:

```
append( [], X, X ).  
append( [X|Y], Z, [X|R] ) :- append(Y, Z, R).
```

Question 1: De quelle nature sont les arguments du prédicat append?

Question 2: Que fait cet ensemble de clauses? Expliquez la propriété qui est toujours vérifiée entre les arguments du prédicat append.

Questions 3, 4 et 5: Quelles sont les réponses aux requêtes suivantes:

```
append(X, "er", "penser").  
append(A, "er", "penser"), append(A, "ons", B).  
append(A, X, "penser"), append(A, Y, "pensons").
```

Rappel:

Le prédicat `name(X, Y)` est tel que la liste de caractères `Y` forme le nom `X`, comme nous l'avons vu en cours dans le cas des nombres. Ainsi la requête `name(X, "amener")` donne `X = amener`

On ajoute alors les clauses suivantes:

```
inconnu(X, Y, T) :- mot(X), fin(Y, Z),  
                    append(A, "er", X), append(A, Z, U), name(T, U).
```

```
mot("aimer").  
mot("lever").  
fin("nous", "ons").  
fin("vous", "ez").
```

Question 6: Quelle est la réponse à la requête: `inconnu(X, "nous", Y)`. Expliquez en vous servant des réponses aux questions 3 à 5.

Question 7: Quelle est la réponse à la requête: `inconnu(X, Y, Z)`.

On ajoute les clauses suivantes:

```
mot("lacher").  
fin("je", "e").  
fin("tu", "es").  
fin("il", "e").  
fin("elle", "e").  
fin("elles", "ent").  
fin("ils", "ent").
```

Question 8: Quelle est maintenant la réponse à la requête: `inconnu(X, Y, lachent)`.

Question 9: Quelle est la réponse à la requête: `inconnu(X, Y, aime)`.

Durée: 1 heure 30 minutes

Les Documents et calculettes sont autorisés, mais ne doivent en aucun cas circuler entre les étudiants