



Analyse d'un programme PROLOG

Un programme Prolog commence par les lignes suivantes:

```
element(X,[X|_]).
```

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

Question 1: De quel type sont les arguments du prédicat member?

Question 2: Que fait le prédicat member? Indiquez la propriété qui est toujours vérifiée entre ses arguments.

Le programme continue avec les lignes suivantes:

```
inverse(L,R):- inverse(L,[],R).
```

```
inverse([],R,R).
```

```
inverse([_|H|T],S,R):- inverse(T,[H|S],R).
```

Question 3: De quels types sont les arguments du prédicat reverse? Que remarquez-vous?

Question 4: Que fait le prédicat reverse? Indiquez la propriété qui est toujours vérifiée entre ses arguments.

Le programme continue avec les lignes suivantes:

```
route(Z,Z,L,R) :- inverse(L,R).
```

```
route(X,Y,L,R) :- pas(X,Z), not(element(Z,L)), route(Z,Y,[Z|L],R).
```

```
pas(1,8).
```

```
pas(3,4).
```

```
pas(8,3).
```

Question 5: De quels types sont les arguments du prédicat path? Tenez compte des réponses aux questions précédentes pour préciser.

Question 6: Quelle est la réponse à la requête route(1,2,[1],R) ?

On ajoute au programme les lignes suivantes:

```
pas(1,6).
```

```
pas(2,7). pas(2,9).
```

```
pas(3,8).
```

```
pas(4,3). pas(4,9).
```

```
pas(6,1).
```

```
pas(6,7).
```

```
pas(7,2).
```

```
pas(7,6).
```

```
pas(8,1).
```

```
pas(9,4). pas(9,2).
```

Question 7: Quelle est maintenant la réponse à la requête route(1,2,[1],R) ?

Question 8: Plus généralement, que provoque la requête route(P,Q,[P],R) ? Il n'est pas demandé de calculer la réponse, mais de décrire en quelques phrases ce qu'elle exprime.

Durée: 1 heure 30 minutes

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