

Ejercicios

1. ¿Qué ocurrirá si preguntamos

?- signo(7,X,Signo). ?

Justifique su respuesta.

Nos devolverá el primer hecho que es: horoscopo(aries,21,3,21,4). debido a que toma el 7 y lo analiza con el primer hecho, como 7 es menor que 21 se descarta que sea en el mes 3, porque a partir del 21 del mes 3 se puede decir que es aries, ahora checa que 7 es menor que 21 y eso entra dentro del mes 4. Por lo que suponiendo que X sea igual a 4 entonces el Signo es Aries.

?- signo(7,X,Signo).

X = 4,

Signo = aries .

2. ¿Qué ocurrirá si preguntamos

?- signo(X,7,Signo). ?

Justifique su respuesta.

Nos devuelve un error ya que hay más de un resultado posible, me refiero a que X puede ir del 1 al 21 del mes 7, no hay un valor exacto, como dice el error los argumentos no están lo suficientemente instanciados.

?- signo(X,7,Signo).

ERROR: Arguments are not sufficiently instantiated

ERROR: In:

ERROR: [9] _9822=<21

ERROR: [8] signo(_9848,7,cancer) at prolog.pl:14

ERROR: [7] <user>

3. ¿El ejemplo contesta afirmativamente a preguntas del tipo

?- signo(74,4,tauro). ?

Si la respuesta fue afirmativa, modifica el ejemplo para que trabaje con el número de días correcto para cada mes.

La respuesta es afirmativa, pero debe ser negativa debido a que la condición dice que si el día ingresado es mayor o igual al día de inicio o menor o igual al ultimo día. Pero realmente eso hace que el intervalo sean todos los números, por lo que para reducirlo a un intervalo que solo tenga los días del signo tenemos que cambiar el “y”

por una “o” , o como es en prolog tenemos que cambiar el “;” por “.”. Y así solo dirá verdadero a días dentro del intervalo del horóscopo. Y sería cambiar esta linea:

```
- signo(Dia,Mes,Signo) :- horoscopo(Signo,D1,M1,D2,M2), ((Mes=M1,Dia>=D1);  
(Mes=M2,Dia=<D2)).  
+ signo(Dia,Mes,Signo) :- horoscopo(Signo,D1,M1,D2,M2), ((Mes=M1,Dia>=D1),  
(Mes=M2,Dia=<D2)).
```

4. Definir un predicado estudiante_de(E,P) que se cumple si un estudiante E es en efecto el estudiante de un profesor P.

```
estudiante_de(E,P) :- curso(P1,X), estudiante(E1,X), (E=E1, P1=P, X=X).
```

Se deben de cumplir los 3, el punto aquí es si empata el curso.

5. En la mansión Dreadbury han asesinado a la tía Alicia, y el departamento de policía ha decidido usar el lenguaje de programación Prolog para resolver el caso. El ejercicio consiste en definir la base de conocimientos teniendo en cuenta que:
 - a) Un asesino siempre odia a su víctima y nunca es más rico que ella. El asesino de la tía Alicia, además vive en la mansión Dreadbury.
 - b) Tía Alicia, Bartolo y el Carnicero son las únicas personas que viven en la mansión.
 - c) Bartolo odia a todas las personas de la mansión que no son odiadas por la tía Alicia.
 - d) Alicia odia a todos los que viven en la mansión excepto al Carnicero.
 - e) Quien no es odiado por el Carnicero y vive en la mansión, es más rico que tía Alicia.
 - f) El Carnicero odia a las mismas personas que odia tía Alicia.

Una vez definida la base de conocimientos, se debe definir un predicado asesino(A,P) que se cumple si A es el asesino de una persona P.

```
vive_mansion(tiaAlicia).  
vive_mansion(bartolo).  
vive_mansion(carnicero).
```

```
odia(tiaAlicia,bartolo).  
odia(carnicero,bartolo).  
odia(bartolo,tiaAlicia).  
odia(bartolo,carnicero).
```

```
mas_rico(carnicero,tiaAlicia).  
mas_rico(carnicero,bartolo).  
mas_rico(tiaAlicia,bartolo).
```

```
asesino(A,P) :- odia(A,P), mas_rico(P,A), vive_mansion(A).
```

Bartolo asesinó a la tiaAlicia.

```
?- asesino(bartolo,tiaAlicia).  
true .
```

```
?- asesino(carnicero,tiaAlicia).  
false.
```

```
?- asesino(tiaAlicia,tiaAlicia).  
false.
```