
LENGUAJES, TECNOLOGÍAS Y PARADIGMAS DE PROGRAMACIÓN

TEMA 4:

PROGRAMACIÓN LÓGICA

(EJERCICIOS DE AULA)



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Informática  **etsinf**

1. Indica cuál de las siguientes sustituciones es el mgu de las expresiones $\text{append}([b], [c, d], L)$ y $\text{append}([X|X_s], Y_s, [X|Z_s])$:

- ☐ A $\{X/[b], Y_s/[c, d], X_s/[], Z_s/[], L/[b]\}$.
- ☐ B $\{X/b, Y_s/[c, d], X_s/[], L/[b|Z_s]\}$.
- ☐ C $\{X/b, Y_s/[c, d], X_s/[], Z_s/[], L/[[b]|Z_s]\}$.
- ☐ D no existe ningún unificador porque $\{[b]\}$ y $\{[X|X_s]\}$ no unifican.

2. Indica la respuesta correcta en relación al objetivo $\text{sublista}(X, [1, 2, 3])$ y al siguiente código Prolog:

```
sublista([], _).  
sublista([X|L1], [X|L2]) :- sublista(L1, L2).  
sublista([X|L1], [Y|L2]) :- sublista([X|L1], L2).
```

- ☐ A La primera solución que encuentra Prolog es $X=[]$.
- ☐ B La primera solución que encuentra Prolog es $X=[1]$
- ☐ C La primera solución que encuentra Prolog es $X=[1, 2, 3]$
- ☐ D La primera solución que encuentra Prolog es $X=[3]$

3. Indica por qué es incorrecto el siguiente programa en Prolog:

```
suma_elementos([], 0).  
suma_elementos([H|T], H+Y1) :- suma_elementos(T, Y1).
```

- ☐ A Los predicados no pueden tener el símbolo “_” en su nombre.
- ☐ B La expresión $H+Y1$ no devuelve el resultado de efectuar la suma.
- ☐ C Las variables deben contener solo letras y la primera en mayúsculas, por lo que $Y1$ es incorrecto.
- ☐ D La primera letra de nombre de los predicados debe escribirse siempre con mayúsculas.

4. Indica cuál de las siguientes afirmaciones es **FALSA** respecto al siguiente programa lógico P:

```
impar(s(X)) :- par(X).
```

- ☐ A toda derivación usando P computa la respuesta $\{X/s(X)\}$
- ☐ B toda derivación usando P es finita.
- ☐ C en este programa no tiene éxito ningún objetivo.
- ☐ D cualquier derivación usando P a partir de un objetivo inicial falla.

5. Dado el siguiente programa lógico:

```
p(X):- s(X),r(X).  
s(a).  
s(b).  
r(a).  
r(b).  
r(c).
```

¿Cuáles son **TODAS** las respuestas computadas para el objetivo ?- p(X) .
usando la estrategia de búsqueda predefinida de Prolog?

- ☐ A No hay respuestas.
- ☐ B X = a, X = b y X = c
- ☐ C X = a
- ☐ D X = a y X = b

6. Indica cuál de los siguientes predicados podría utilizarse para calcular la media aritmética de dos números:

- ☐ A media(X,Y):- (X + Y) / 2.
- ☐ B media(X,Y,Z):- Z is (X + Y) / 2.
- ☐ C media(X,Y,Z):- Z = (X + Y) / 2.
- ☐ D media(X,Y) = Y is (X + Y) / 2.

7. Dado el siguiente programa Prolog:

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

la respuesta computada para el objetivo ?- p(X,[1,2,3,1,5]) . es:

- ☐ A X = 3
- ☐ B X = 5
- ☐ C X = 2
- ☐ D X = 1

8. Completa el programa siguiente para contar los elementos de una lista:

```
contar([],0).  
contar([X],1).  
contar([X,Y|T],Z) :- .
```

- ☐ A contar(T,Z-2).
☐ B Z is 2+ contar(T).
☐ C contar(T,W), Z is 2 + W.
☐ D Z is contar(contar(T,[])).

9. Asumiendo definido un predicado de inversión de listas `invertir(L,M)`, indica cuál de las siguientes cláusulas Prolog define una relación *capicúa* sobre secuencias de elementos representados en forma de listas:

- ☐ A `capicua(A) :- invertir([X|A],[A|X]).`
☐ B `capicua(A) :- invertir(A,A).`
☐ C `capicua(A) :- invertir([X,Y|A],[A|Y,X]).`
☐ D `capicua(A) :- A=[X|L],invertir(L,[X|M]).`

10. Dado el siguiente programa Prolog:

```
p(Y) :- q(Y), p(Y).  
q(0).  
q(1).  
p(1).
```

y el objetivo `?- p(X).`, indica cuál de las siguientes es la primera respuesta computada que mostrará el intérprete por pantalla:

- ☐ A `X = 0`
☐ B `X = 1`
☐ C `X = Y`
☐ D ninguna de las anteriores

11. Completa correctamente el siguiente programa Prolog:

```
minimos([X],[X]).
```

```
minimos([X,Y|L1],[U,U|L2]):- , minimos([Z|L1],[U|L2]).
```

para que el objetivo `?- minimos(L1,L2)` verifique que L2 es la lista obtenida sustituyendo cada elemento de la lista numérica no vacía L1 por el menor elemento de L1. Es decir

```
?-minimos([4,2,5,1,3,6],L).
```

```
L=[1,1,1,1,1,1]
```

donde `min(X,Y)` es un operador aritmético predefinido que devuelve el menor de sus dos argumentos.

☐ A `Z = min(X,Y).`

☐ B `Z == min(X,Y).`

☐ C `Z is min(X,Y).`

☐ D `Z <min(X,Y).`

12. Indica cuál de las siguientes afirmaciones es cierta:

☐ A en una lista `[H|T]`, H es el primer elemento de la lista y T es la lista que contiene el resto de los elementos.

☐ B el predicado `append` de concatenación de listas es de orden superior porque sus tres argumentos son listas de longitud variable.

☐ C en los programas lógicos, todos los argumentos de un predicado son de entrada, excepto el último que siempre es de salida.

☐ D en Prolog no se pueden definir predicados recursivos si no se usa el `is`.

13. Completa el programa siguiente para sumar los elementos de una lista:

```
sumaLista([],0).
```

```
sumaLista([X],X).
```

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

☐ A `Z is X + W, sumaLista(Y,W).`

☐ B `X+sumaLista(Y,Z).`

☐ C `sumaLista(Y,X+Z).`

☐ D `sumaLista(Y,W), Z is X + W.`

14. Indica cuál de las siguientes afirmaciones es **CIERTA** en relación con este programa:

```
mascota(X) :- animal(X), amo(X,Y), persona(Y).  
animal(X) :- gato(X); perro(X).  
gato(miau).  
amo(miau,enrique).  
amo(guau,enrique).  
persona(juan).  
persona(enrique).
```

- ☐ A Si lanzamos el objetivo `?-mascota(miau).` el intérprete da como respuesta `X=miau`.
- ☐ B Si añadimos al programa el hecho `perro(guau).` y lanzamos el objetivo `?-mascota(guau).` el intérprete da como respuesta `yes`.
- ☐ C Si lanzamos el objetivo `?-mascota(X).` el intérprete nos da dos soluciones: la primera es `X=miau` y la segunda es `X=guau`.
- ☐ D Ninguna de las anteriores.
15. Completa el siguiente programa Prolog para que compute el factorial de un número:

```
factorial(0,1).  
factorial(N,F) :- N>0, N1 is N-1, factorial(N1,F1), 
```

- ☐ A `F = N * F1.`
- ☐ B `F1 is N * F.`
- ☐ C `F is N * F1.`
- ☐ D no es necesario añadir nada.
16. Dado el siguiente programa:

```
legusta(pepe,pesca).  
legusta(maria,bailar).  
legusta(ana,pesca).  
legusta(pepe,musica).  
legusta(maria,musica).  
legusta(ana,bailar).
```

¿Con cuál de los siguientes objetivos podemos saber qué le gusta a ana?

- ☐ A `?- legusta(X,Y).`
- ☐ B `?- legusta (X,ana).`
- ☐ C `?- legusta(ana,X).`
- ☐ D `?- X=legusta(ana).`

17. Indica cuál de los siguientes programas Prolog representa la siguiente información:

- a) Si X está arriba de Y y se tocan, entonces X está encima de Y .
- b) La taza está arriba del libro.
- c) La taza toca al libro.

- ☐ A `arriba(taza,libro).`
`toca(taza,libro).`
`arriba(X,Y):-encima(X,Y).`
`toca(X,Y):-encima(X,Y).`
- ☐ B `arriba(taza,libro).`
`toca(taza,libro).`
`encima(X,Y):-arriba(X,Y),toca(X,Y).`
- ☐ C `arriba(taza,libro).`
`toca(taza,libro).`
`arriba(X,Y),toca(X,Y):-encima(X,Y).`
- ☐ D `arriba(taza,libro).`
`toca(taza,libro).`
`encima(X,Y):-arriba(X,Y).`
`encima(X,Y):-toca(X,Y).`

CUESTIONES

1. Los invitados a una fiesta se representan con los predicados hombre/1 y mujer/1 que determinan su sexo.

```
hombre(alfredo).  
hombre(felipe).  
hombre(francisco).  
mujer(sonia).  
mujer(eva).  
mujer(carmen).
```

Además, conocemos las bebidas preferidas de cada invitado definidas por el predicado bebe/2:

```
bebe(alfredo, whisky).  
bebe(alfredo, ron_cola).  
bebe(felipe, cerveza).  
bebe(felipe, gin_tonic).  
bebe(felipe, ron_cola).  
bebe(francisco, vino).  
bebe(francisco, malibu).  
bebe(sonia, gin_tonic).  
bebe(sonia, malibu).  
bebe(eva, vino).  
bebe(eva, cerveza).  
bebe(carmen, whisky).  
bebe(carmen, ron_cola).
```

Define un predicado pareja(X,Y) que tenga éxito cuando X es un hombre e Y una mujer y tengan al menos una bebida favorita en común. Define también el predicado pareja2(X,Y) que se satisface cuando X e Y tienen al menos dos bebidas favoritas en común.

2. Modifica el ejercicio anterior para reflejar los siguientes hábitos de bebida:

- Pepe bebe cualquier cosa que beba Alfredo.
- Elena bebe cualquier cosa que beban Sonia o Felipe.

3. Queremos viajar por Andalucía usando un coche bastante antiguo e incómodo del que no nos fiamos. Por ello, decidimos que sólo podemos plantearnos un viaje de la provincia X a la provincia Y si éstas son fronterizas o si para llegar a ellas basta con atravesar una provincia.

Define un predicado viaje(X,Y) que se satisface cuando es posible viajar de la provincia X a la provincia Y.



Figura 1: Mapa de Andalucía

4. Sea el siguiente enunciado:

Pedro padece gripe
 Pedro padece hepatitis
 Juan padece hepatitis
 Maria padece gripe
 Carlos padece intoxicacion
 La fiebre es sintoma de gripe
 El cansancio es sintoma de hepatitis
 El vomito es sintoma de intoxicacion
 El cansancio es sintoma de gripe
 La aspirina suprime la fiebre
 El Motilium suprime el vomito

Un fármaco alivia una enfermedad si la enfermedad tiene un sintoma que sea suprimido por el fármaco. Una persona debería tomar un fármaco si padece una enfermedad que sea aliviada por el fármaco. Representa el enunciado como un programa Prolog y utiliza el programa para responder las siguientes preguntas:

- ¿Qué dolencia tiene Pedro?
- ¿Y María?
- ¿Quién padece gripe?
- ¿Qué síntomas presenta Pedro?
- ¿Quién padece vomito?
- ¿Y quién está cansado?
- ¿Hay algún fármaco que alivie a Pedro?
- ¿Hay algún síntoma que compartan Juan y María?

5. Sea el siguiente enunciado:

Bertoldo y Bartolo son rufianes.
Romeo y Bertoldo, como su nombre indica, son nobles.
Bartolo es un plebeyo.
Gertrudis y Julieta son damas.
Julieta es hermosa.
Los plebeyos desean a cualquier dama, mientras que los nobles solo a aquellas que son hermosas.
Los rufianes, para satisfacer sus instintos, raptan a las personas a las que desean.

Representa el enunciado como un programa Prolog y utiliza el programa para responder las siguientes preguntas:

- ¿Qué noble es un rufián?
- ¿A quién podría raptar Romeo?
- ¿Quién puede raptar a Julieta?
- ¿Quién rapta a quién?
- ¿A quién desea Bartolo?
- ¿Y Romeo?
- ¿Qué hermosa dama es deseada por Bartolo?

6. Definir la relación `palindromo(L)` que verifique si una lista `L` es un palíndromo. Por ejemplo,

```
?- palindromo([o,s,o]).  
yes
```

7. Supongamos que representamos los puntos del plano mediante términos de la forma `punto(X,Y)` donde `X` e `Y` son números, y los segmentos del plano mediante términos de la forma `segmento(P1,P2)` donde `P1` y `P2` son los puntos extremos del segmento. Define las relaciones `vertical(S)` y `horizontal(S)` que se verifiquen si `S` es un segmento vertical u horizontal, respectivamente. Por ejemplo,

```
?- vertical(segmento(punto(1,2),punto(1,3))).  
yes  
?- vertical(segmento(punto(1,2),punto(4,2))).  
no  
?- horizontal(segmento(punto(1,2),punto(5,2))).  
yes
```

8. Define una relación $f(X,Y)$ de forma que:

```
si  $X < 3$ , entonces  $Y = 0$ .  
si  $3 \leq X < 6$ , entonces  $Y = 2$   
si  $6 \leq X$ , entonces  $Y = 4$ 
```

y construye el árbol de derivaciones para los objetivos

```
?- f(1,Y), 2 < Y.  
?- f(7,Y).
```

9. ¿Qué hace el siguiente programa Prolog?

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

10. Escribe un programa Prolog que compruebe si la longitud de una lista es impar.
11. Escribe un programa Prolog que compruebe si la longitud de una lista es par.
12. Escribe una nueva versión de los dos problemas anteriores por recursión cruzada.
13. Escribe un programa en Prolog que elimine las repeticiones de elementos de una lista, guardando el resultado en una segunda lista. Por ejemplo, $[a,b,h,j]$ se obtendría como resultado de eliminar los elementos repetidos de la lista $[a,b,h,b,a,j]$.
14. Escribe un programa que cuente el número de veces que un elemento se encuentra repetido en una lista.
15. Escribe un programa en Prolog para calcular el mayor elemento de una lista de enteros.
16. Escribe un programa en Prolog para dividir una lista respecto un umbral, dejando los valores menores a la izquierda y los mayores a la derecha del umbral. Por ejemplo, el resultado de partir la lista $[2,7,4,8,9,1]$ respecto al umbral 6 serían las listas $[2,4,1]$ y $[7,8,9]$.
17. Escribe un programa en Prolog para sumar los elementos de una lista de enteros.