

Apellidos:

Nombre:

Ejercicio 1 [2.5 puntos] Una *lista numérica izquierda* (lni) es un término definido de manera recursiva: (1) la constante `vacío` es una lni y representa la lista vacía (2) el término `lni(X,Y)` es una lni si X es una lni e Y es un número. El ejercicio tiene 2 apartados:

1. Definir el predicado `es_lni(+L)` que se verifique si L es una lni. Por ejemplo:

```
?- es_lni(vacio).
   Yes
?- es_lni(lni(lni(lni(vacio,3),2),1)).
   Yes
?- es_lni(lni(lni(lni(3,vacio),2),1)).
   No
```

2. Definir un predicado recursivo `elto(+L,?X)` que se verifique si X es un elemento de la lni L. Por ejemplo

```
?- elto(vacio,X).
   No
?- elto(lni(lni(lni(vacio,3),2),1),X).
   X = 1 ; X = 2 ; X = 3 ;
   No
```

SOLUCION:

(a) `es_lni(+L)`

```
es_lni(vacio).
es_lni(lni(X,Y)):-
    number(Y),
    es_lni(X).
```

(b) `elto(+L,?X)`

```
elto(lni(_,Y),Y).
elto(lni(X,_),Y):-
    elto(X,Y).
```

Ejercicio 2 [2.5 puntos] Consideremos el siguiente programa formado por hechos.

```
p1(a).    p2(a).    p3(a).    p4(a).
          p2(b).          p4(b).
                              p4(c).
```

Los símbolos de predicado que aparecen son `p1`, `p2`, `p3` y `p4`. Todos tienen aridad 1. `p1` y `p3` se verifican únicamente sobre la constante `a`, `p2` sobre `a` y `b`. Finalmente `p4` se verifica sobre `a`, `b` y `c`. Se pide definir un predicado `numero_sol(+Pred,?N)`, que tome como entrada un predicado `p1`, `p2`, `p3` o `p4` y devuelva el número N de constantes que verifican ese predicado. Por ejemplo

```
?- numero_sol(p1,N).          ?- numero_sol(p3,N).
   N = 1 ;
   No
?- numero_sol(p2,N).          ?- numero_sol(p4,N).
   N = 2 ;
   No                          N = 3 ;
                              No
```

SOLUCION:

```
numero_sol(Pred,N):-  
  G =.. [Pred,X],  
  findall(X,G,L),  
  length(L,N).
```

Ejercicio 3 [2.5 puntos] Definir el predicado `unificables(L1,X,L2)` que recibe como entrada una lista de términos L1 y un término T y devuelve la lista L2 de términos de L1 que son unificables con T. Por ejemplo:

```
?- unificables([X,f(Z,Z),f(a,Y),f(Z,b)],f(a,b),L).  
   L = [X, f(a,Y), f(Z, b)] ;  
   No
```

SOLUCION:

```
unificables([],_,[]).  
unificables([T|L1],X,L2):-  
  T \= X,  
  !,  
  unificables(L1,X,L2).  
unificables([T|L1],X,[T|L2]):-  
  unificables(L1,X,L2).
```

Ejercicio 4 [2.5 puntos] El cuadrado de 45 es 2025. Si partimos el número 2025 en dos trozos obtenemos 20 y 25. Por último $20 + 25 = 45$, que es el número del que partimos. Define un predicado `numprop(+N)` que se verifique si N es un número con esa propiedad, esto es, que su cuadrado tenga cuatro cifras y admita un tratamiento análogo al cuadrado de 45. (Nota: El operador para división entera es `//`).

SOLUCION:

```
numprop(N):-  
  between(32,99,N),  
  Z is N * N,  
  Z // 100 + Z mod 100 =:= N.
```