



Programación Lógica

Clase 9 – Continuación - Introducción al paradigma lógico

PARADIGMAS DE PROGRAMACION

UTN - La Plata



Temario

1. Listas en Prolog
2. Ejemplos
3. Actividades

1. Listas

- Son conjuntos de elementos del mismo tipo.
- Las listas se denotan con corchetes, separando los elementos por comas.
Por ej. [1,2,3,4]
- Están divididas en cabeza (X) y cuerpo (Xs) $\rightarrow [X \mid Xs]$
- La lista vacía se denota [].
- Se recorren recursivamente.

1. Listas

- Contienen términos (variables, constantes o estructuras) en general. Es posible anidar listas.

Ejemplos:

- [1,2,3,4]
- [a,b,c,d]
- [1,'a', X,[1,2,3]]
- [[1,2],[3,4],[5,6]]

2. Listas: ejemplos

Longitud de una lista

`long([],C):-C is 0.`

`long([X|Xs], C):- long(Xs,C1), C is C1 + 1.`

Suma de los elementos de una lista

`sum([],0).`

`sum([X|Xs], S):-sum(Xs,S1), S is S1 + X.`

Cantidad de elementos pares de una lista

`par(X):- integer(X), 0 is X mod 2.`

`impar(X):-integer(X), 1 is X mod 2.`

`cantP([],0).`

`cantP([X|Xs], S):- par(X), cantP(Xs,S1), S is S1 + 1.`

`cantP([X|Xs], S):- impar(X), cantP(Xs,S1), S is S1 .`

2. Listas: ejemplos

Verifica si todos los elementos de una lista son hombres

hombre(juan).

hombre(pedro).

hombres([]).

hombres([X|Xs]):-hombre(X),hombres(Xs).

Verifica si todos los elementos de una lista son iguales

todoslg([]).

todoslg([X]).

todoslg([X,Y|Xs]):- X==Y, todoslg(Xs).

todoslg([X,Y|Xs]):- X \==Y, false.

2. Listas: ejemplos

Cuenta los elementos de una lista que son hombres

hombre(juan).

hombre(pedro).

hombres([],0).

hombres([X|Xs], Cont):-hombre(X),hombres(Xs, C), Cont is C +1.

hombres([X|Xs], Cont):- not (hombre(X)) ,hombres(Xs, C), Cont is C.

Cantidad de elementos mayores a 5 de una lista

cantMay5([],0).

cantMay5([X|Xs], S):- X > 5, cantMay5(Xs,S1), S is S1 + 1.

cantMay5([X|Xs], S):- X < 5, cantMay5(Xs,S1), S is S1 .

cantMay5([X|Xs], S):- X == 5, cantMay5(Xs,S1), S is S1 .

2. Listas: ejemplos

Imprimir los elementos de una lista

`mostrar([]).`

`mostrar([X|Xs]):- write(X), nl, mostrar(Xs).`

No existe un elemento en una lista

`noesta(X, []).`

`noesta(X, [Y|Ys]):- X \==Y, noesta(X,Ys).`

`noesta(X, [Y|Ys]):- X ==Y, false.`

`noesta(X, []).`

`noesta(X, [Y|Ys]):- X \==Y, noesta(X,Ys).`

Acá uso la hipótesis de mundo cerrado: lo que no está en la base de conocimiento, es falso.

2. Listas:ejemplos.

Existe un elemento en una lista

```
member(X,[ ]):-fail.
```

```
member(X,[ Y|Ys]):- X==Y, true.
```

```
member(X,[Y|Ys]) :- X\==Y, member(X,Ys).
```

Otra forma:

```
member(X,[X|Xs]).
```

```
member(X,[Y|Ys]) :- member(X,Ys).
```

Añadir un elemento a una lista (al pcpio)

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

```
add(X,L,[X|L]).
```

Otra forma

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

```
add(X,[Y|Ys],[X,Y|Ys]).
```

2. Listas: ejemplos

Borrar **un solo** elemento de una lista

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

`del(X,[Y|Ys],Ys):- X==Y.`

`del(X,[Y|Ys],[Z|Zs]):- X\==Y, del(X,Ys,Zs), Z is Y.`

Otra forma:

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

`del(X,[X|Xs],Xs).`

`del(X,[Y|Ys],[Y|Zs]) :- X\==Y, del(X,Ys,Zs).`

Verificar si hay algún elemento positivo en la lista

`unPosit([]):- false.`

`unPosit([X|Xs]):- X>0.`

`unPosit([X|Xs]):- (X < 0; X=0), unPosit(Xs).`

2. Listas: ejemplos

Dada una lista de apellidos de alumnos, y conociendo sus promedios, imprimir los nombres de aquellos alumnos cuyo promedio es mayor a 7.

```
prom(mia,5).  
prom(juan,8).  
prom(ana,6).  
prom(lucas,9).
```

```
impriProm([]).  
impriProm([X|Xs]):-prom(X,Val),Val>7, write(X),nl, impriProm(Xs).  
impriProm([X|Xs]):-prom(X,Val),Val<7,impriProm(Xs).  
impriProm([X|Xs]):-prom(X,Val),Val=7,impriProm(Xs).
```

2. Listas: ejemplos

Definir un programa lógico que dada una lista de alumnos:

- retorne la lista de aquellos que promocionan. Se sabe que un alumno promociona si cumplió con la asistencia solicitada y su nota es mayor o igual a 7.
- Imprima la lista de aprobados. Para aprobar la nota obtenida debe ser mayor o igual que 6.

La base de conocimiento contiene hechos sobre las notas de cada alumno y su asistencia.

nota(juan,6).

nota(pedro,8).

nota(luis,9).

nota(ana,7).

nota(andrea,6).

nota(tomas,4).

nota(belen,2).

asistencia(juan).

asistencia(pedro).

asistencia(ana).

asistencia(andrea).

asistencia(belen).

2. Listas: ejemplos

promociona(X):-nota(X,Z),(Z>7;Z=7),asistencia(X). **%uso el ; como OR %**

% promociona(X):-nota(X,Z),Z<7,fail. Opcional, falla por la nota

% promociona(X):-fail. Opcional, falla por asistencia

listaPromo([],[]).

listaPromo([X|Xs],[X|Ys]):-promociona(X), listaPromo(Xs,Ys).

listaPromo([X|Xs],L):-not(promociona(X)),listaPromo(Xs,L).

listaAprob([]).

listaAprob([X|Xs]):-nota(X,Z), (Z>6;Z=6), write(X),nl, listaAprob(Xs).

listaAprob([X|Xs]):-nota(X,Z), Z<6, listaAprob(Xs).

3. Actividades

- 1) Dada una lista de números definir el programa lógico que retorne la cantidad de elementos mayores a 4 que contiene la misma.

- 1) Dada una lista de números definir el programa lógico que retorne la suma de los elementos positivos que contiene la misma.

- 2) Dada una lista de apellidos y un apellido P, definir un programa lógico que retorne verdadero si P pertenece a la lista, o falso en caso contrario.

