

Programación declarativa (2007–08)

Tema 7: Listas, operadores y aritmética

José A. Alonso Jiménez

Grupo de Lógica Computacional
Departamento de Ciencias de la Computación e I.A.
Universidad de Sevilla

Tema 7: Listas, operadores y aritmética

1. Listas

- Construcción de listas
- Definición de relaciones sobre listas
 - Concatenación de listas
 - Relación de pertenencia

2. Disyunciones

3. Operadores

- Operadores aritméticos
- Definición de operadores

4. Aritmética

- Evaluación de expresiones aritméticas
- Definición de relaciones aritméticas

Tema 7: Listas, operadores y aritmética

1. Listas

Construcción de listas

Definición de relaciones sobre listas

Concatenación de listas

Relación de pertenencia

2. Disyunciones

3. Operadores

4. Aritmética

Construcción de listas

- ▶ Definición de listas:
 - ▶ La lista vacía $[]$ es una lista.
 - ▶ Si L es una lista, entonces $.(a,L)$ es una lista.

- ▶ Ejemplos:

$$?- .(X,Y) = [a].$$

$$X = a$$

$$Y = []$$

$$?- .(X,Y) = [a,b].$$

$$X = a$$

$$Y = [b]$$

$$?- .(X,.(Y,Z)) = [a,b].$$

$$X = a$$

$$Y = b$$

$$Z = []$$

Escritura abreviada

- ▶ Escritura abreviada:

$$| [X|Y] = .(X,Y)$$

- ▶ Ejemplos con escritura abreviada:

$$|- [X|Y] = [a,b] .$$

$$X = a$$

$$Y = [b]$$

$$|- [X|Y] = [a,b,c,d] .$$

$$X = a$$

$$Y = [b, c, d]$$

$$|- [X,Y|Z] = [a,b,c,d] .$$

$$X = a$$

$$Y = b$$

$$Z = [c, d]$$

Tema 7: Listas, operadores y aritmética

1. Listas

Construcción de listas

Definición de relaciones sobre listas

Concatenación de listas

Relación de pertenencia

2. Disyunciones

3. Operadores

4. Aritmética

Definición de concatenación (append)

- *Especificación:* $\text{conc}(A, B, C)$ se verifica si C es la lista obtenida escribiendo los elementos de la lista B a continuación de los elementos de la lista A . Por ejemplo,

```
?- conc([a,b],[b,d],C).
C = [a,b,b,d]
```

- *Definición 1:*

```
conc(A,B,C) :- A=[], C=B.
```

```
conc(A,B,C) :- A=[X|D], conc(D,B,E), C=[X|E].
```

- *Definición 2:*

```
conc([],B,B).
```

```
conc([X|D],B,[X|E]) :- conc(D,B,E).
```

Consultas con la relación de concatenación

- Analogía entre la definición de conc y la de suma,
- ¿Cuál es el resultado de concatenar las listas $[a,b]$ y $[c,d,e]$?

```
?- conc([a,b],[c,d,e],L).
L = [a,b,c,d,e]
```

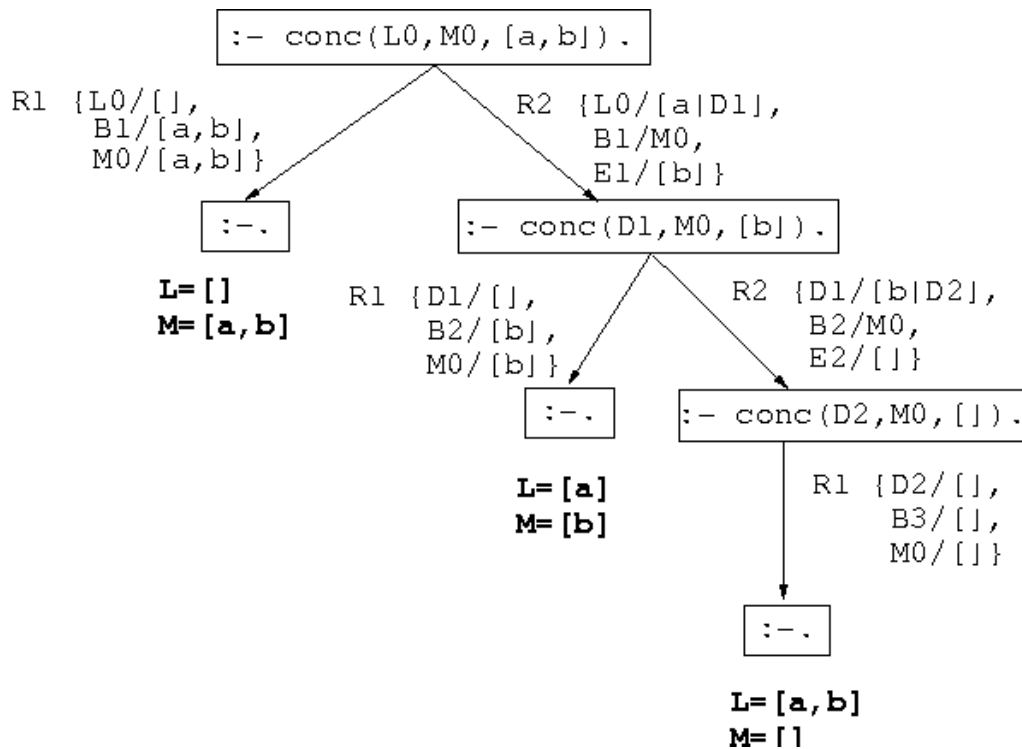
- ¿Qué lista hay que añadirle a la lista $[a,b]$ para obtener $[a,b,c,d]$?

```
?- conc([a,b],L,[a,b,c,d]).
L = [c,d]
```

- ¿Qué dos listas hay que concatenar para obtener $[a,b]$?

```
?- conc(L,M,[a,b]).
L = []           M = [a,b] ;
L = [a]         M = [b] ;
L = [a,b]       M = [] ;
No
```

Árbol de deducción de $?- \text{conc}(L, M, [a, b])$.



Definición de la relación de pertenencia (member)

- ▶ *Especificación:* pertenece(X,L) se verifica si X es un elemento de la lista L.
- ▶ *Definición 1:*

$\text{pertenece}(X, [X|L]).$
 $\text{pertenece}(X, [Y|L]) \text{ :- pertenece}(X, L).$

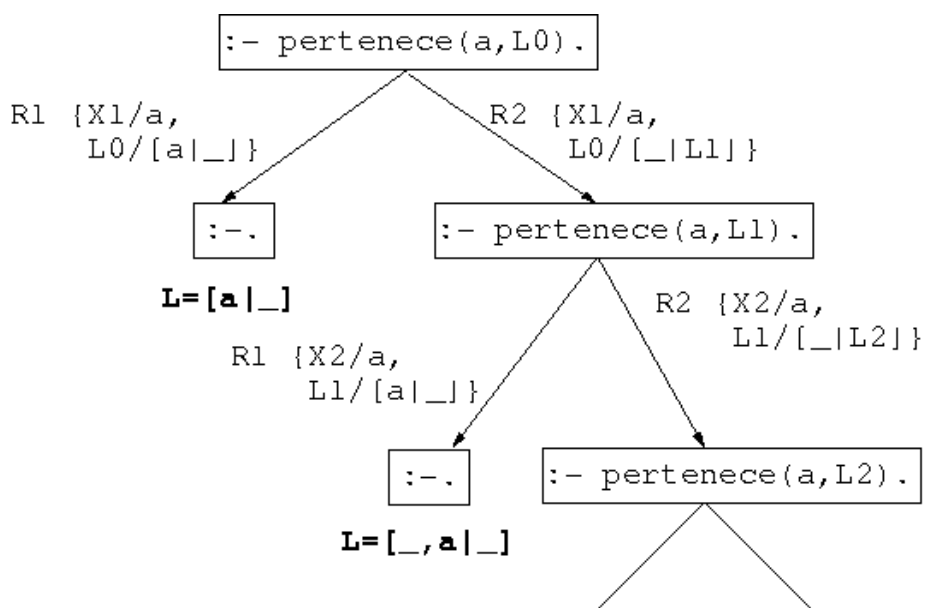
- ▶ *Definición 2:*

$\text{pertenece}(X, [X|_]).$
 $\text{pertenece}(X, [_|L]) \text{ :- pertenece}(X, L).$

Consultas con la relación de pertenencia

```
?- pertenece(b, [a,b,c]).
Yes
?- pertenece(d, [a,b,c]).
No
?- pertenece(X, [a,b,a]).
X = a ;
X = b ;
X = a ;
No
?- pertenece(a,L).
L = [a|_G233] ;
L = [_G232, a|_G236] ;
L = [_G232, _G235, a|_G239]
Yes
```

Árbol de deducción de `?- pertenece(a,L)`.



Disyunciones

- ▶ Definición de pertenece con disyunción

$\text{pertenece}(X, [Y|L]) :- X=Y ; \text{pertenece}(X,L).$

- ▶ Definición equivalente sin disyunción

$\text{pertenece}(X, [Y|L]) :- X=Y.$

$\text{pertenece}(X, [Y|L]) :- \text{pertenece}(X,L).$

Tema 7: Listas, operadores y aritmética

1. Listas

2. Disyunciones

3. Operadores

Operadores aritméticos

Definición de operadores

4. Aritmética

Ejemplos de operadores aritméticos

- ▶ Ejemplos de notación infija y prefija en expresiones aritméticas:

?- +(X,Y) = a+b.

X = a

Y = b

?- +(X,Y) = a+b+c.

X = a+b

Y = c

?- +(X,Y) = a+(b+c).

X = a

Y = b+c

?- a+b+c = (a+b)+c.

Yes

?- a+b+c = a+(b+c).

No

Ejemplos de asociatividad y precedencia

- ▶ Ejemplos de asociatividad:

?- X^Y = a^b^c.

X = a Y = b^c

?- a^b^c = a^(b^c).

Yes

- ▶ Ejemplo de precedencia

?- X+Y = a+b*c.

X = a Y = b*c

?- X*Y = a+b*c.

No

?- X*Y = (a+b)*c.

X = a+b Y = c

?- a+b*c = (a+b)*c.

No

Operadores aritméticos predefinidos

Precedencia	Tipo	Operadores	
500	yfx	+, -	Infijo asocia por la izquierda
500	fx	-	Prefijo no asocia
400	yfx	*, /	Infijo asocia por la izquierda
200	xfy	^	Infijo asocia por la derecha

Tema 7: Listas, operadores y aritmética

1. Listas

2. Disyunciones

3. Operadores

Operadores aritméticos

Definición de operadores

4. Aritmética

Definición de operadores

- ▶ Definición (ejemplo_operadores.pl)

```
:-op(800,xfx,estudian).
```

```
:-op(400,xfx,y).
```

```
juan y ana estudian lógica.
```

- ▶ Consultas

```
?- [ejemplo_operadores].
```

```
?- Quienes estudian lógica.
```

```
Quienes = juan y ana
```

```
?- juan y Otro estudian Algo.
```

```
Otro = ana
```

```
Algo = lógica
```

Tema 7: Listas, operadores y aritmética

1. Listas

2. Disyunciones

3. Operadores

4. Aritmética

Evaluación de expresiones aritméticas

Definición de relaciones aritméticas

Evaluación de expresiones aritméticas

- ▶ Evaluación de expresiones aritmética con `is`.

```
?- X is 2+3^3.
```

```
X = 29
```

```
?- X is 2+3, Y is 2*X.
```

```
X = 5
```

```
Y = 10
```

- ▶ Relaciones aritméticas: `<`, `=<`, `>`, `>=`, `==` y `=/=`

```
?- 3 =< 5.
```

```
Yes
```

```
?- 3 > X.
```

```
% [WARNING: Arguments are not sufficiently instantiated]
```

```
?- 2+5 = 10-3.
```

```
No
```

```
?- 2+5 == 10-3.
```

```
Yes
```

Tema 7: Listas, operadores y aritmética

1. Listas

2. Disyunciones

3. Operadores

4. Aritmética

Evaluación de expresiones aritméticas

Definición de relaciones aritméticas

Definición del factorial

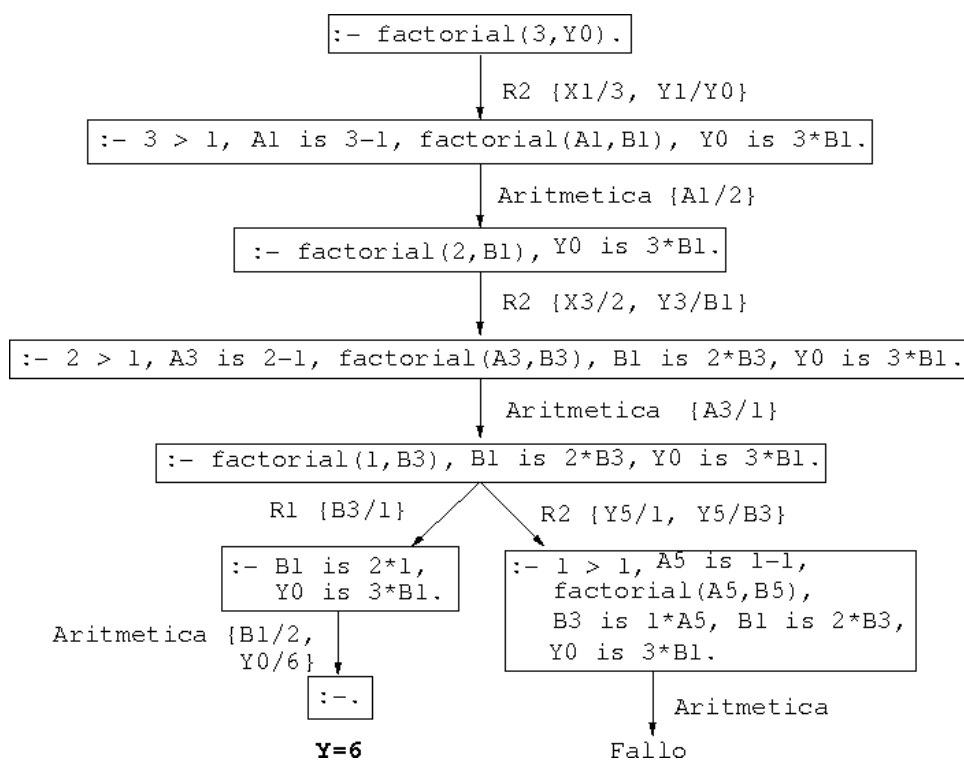
- ▶ `factorial(X,Y)` se verifica si Y es el factorial de X. Por ejemplo,

```
?- factorial(3,Y).
Y = 6 ;
No
```

- ▶ Definición:

```
factorial(1,1).
factorial(X,Y) :-
    X > 1,
    A is X - 1,
    factorial(A,B),
    Y is X * B.
```

Árbol de deducción de `?- factorial(3,Y)`.



Bibliografía

1. J.A. Alonso *Introducción a la programación lógica con Prolog*.
2. Bratko, I. *Prolog Programming for Artificial Intelligence (2nd ed.)* (Addison–Wesley, 1990)
3. Clocksin, W.F. y Mellish, C.S. *Programming in Prolog (Fourth Edition)* (Springer Verlag, 1994)
4. Covington, M.A.; Nute, D. y Vellino, A. *Prolog Programming in Depth* (Prentice Hall, 1997)
5. Sterling, L. y Shapiro, E. *L'art de Prolog* (Masson, 1990)
6. Van Le, T. *Techniques of Prolog Programming* (John Wiley, 1993)