



Universidad Tecnológica Nacional
FACULTAD REGIONAL CORDOBA

PARADIGMAS DE PROGRAMACION

Unidad V
Paradigma Lógico
Parte II



CONTENIDOS ABORDADOS

- Prolog:
 - Sintaxis
 - Tipos de datos
 - Operadores
 - Caso de estudio

Prolog : Sintaxis

Comentarios

- En línea: %.
 % Hola, esto es un comentario.
- En múltiples líneas: /* ... */
 /* Esto es un comentario
 también. */

Finalización de expresiones

- Las expresiones o cláusulas terminan con punto (punto).
 docente(ana).

Prolog : Sintaxis

Variable lógica:

- Es una incógnita, algo que está sin resolver.

docente(X).

X comienza con mayúscula, lo que indica que no es un valor fijo.

- Las variables deben escribirse con Mayúsculas.
- Caso especial: Variables anónimas representadas por `_` (guión bajo), son variables sin nombre.

docente(_).

Prolog : Sintaxis

Constantes

- Representan valores concretos ligados a las expresiones.
- Las constantes se escriben en Minúsculas.
- No se pueden dejar espacios entre los nombres de las constantes.
- Son términos cero-arios, pueden ser constantes de carácter (átomos) o constantes numéricas.

• Ejemplo: `docente(ana).`

`?- docente(X).`

resultado X=ana

el valor ana se liga a X

Prolog : Tipos de datos

- No posee asignación de tipos fuerte.
- La lógica se preocupa de las relaciones entre objetos que del tipo de éstos, dando a todos ellos un tratamiento similar.
- Los datos que maneja son los **términos**.
- El valor que puede tomar una variable consiste en cualquier término, por ejemplo:

j(3), 23.2, 'hola que tal', etc.

- Posee definiciones de números, secuencias de caracteres, listas, tuplas y patrones.

Prolog : Tipos de datos

- Tipos simples:
 - Booleanos: true, false
 - Númericos: enteros, reales
 - Secuencias de caracteres: átomos, string o symbol.
- Tipos compuestos:
 - Objetos compuestos.
 - Listas.

Prolog : Operadores

- **Operadores lógicos**

Operador	Sintaxis en Prolog
AND (conjunción y)	, (coma)
OR (disyunción o)	; (punto y coma)
NOT (negación)	not ó \-
IF (implicación si)	:- (dos puntos y guión)

Prolog : Operadores

Unificación de términos

- No existe la asignación / no hay efecto colateral: una vez más una variable no es una posición de memoria que almacena estados intermedios.
- Una variable es una incógnita, no tiene sentido que yo le haga $X = X + 1$, porque esa condición nunca se puede cumplir.
- Existe la unificación que consiste en que las variables lógicas toman un valor o se ligan a ellos.

Prolog : Operadores

- **Operadores de igualdad**

- **=** (Unificación) Es verdadero si ambos términos unifican, en cuyo caso se realiza la unificación, Ej: $X = 2$. $Y = 1+1$.
- **\=** (No Unificación) Es verdadero si ambos términos NO unifican, en cuyo caso NO se realiza la unificación.
- **==** (Identidad) comprueba si las dos expresiones son exactamente la misma. Compara términos sin evaluar expresiones. Ambos términos tienen que estar instanciados. Ej:
 - ?- $2==2 \Rightarrow \text{true}$. ?- $2==1+1 \Rightarrow \text{false}$.
 - ?- $X=2$, $X==2 \Rightarrow X=2$.
 - ?- $X=2$, $Y=2$, $X==Y \Rightarrow X=Y$, $Y=2$
- **\==** (No identidad) comprueba si dos expresiones no son exactamente la misma. Es falso cuando dos términos son exactamente iguales.

Prolog : Operadores

- **Operadores aritméticos**

+	Suma
-	Resta binaria, o menos unario
*	Multiplicación
/	División
//	Cociente de la división entera
mod	Resto de la división entera (módulo)
^	Potencia (primer término elevado al segundo término)

- Ejemplo: operador suma ("+")

forma prefija: $+ (2, 5)$

forma infija: $2 + 5$.

Prolog : Operadores

- **Operadores aritméticos de relación**

<code>==</code>	Igualdad aritmética
<code>==\=</code>	Distinto
<code>></code>	Mayor
<code><</code>	Menor
<code>>=</code>	Mayor o igual
<code><=</code>	Menor o igual

- **Ejemplo:**

`(2 + 1) == 5 --> False.`

`(2 + 1) == 3 --> True.`

Prolog : Operadores

- **Operadores is :evaluator.**
 - Se utiliza para evaluar las expresiones aritméticas y funciones.
 - Evalúa la parte de la derecha y unifica a la parte izquierda.
 - Si no se usa el *is*, las expresiones se mantienen en su forma original:
 $X = 3 + 3$ se unifica como $X = 3 + 3$
 $X \text{ is } 3 + 3$ se unifica como $X = 6$

Prolog : Operadores

- Ejemplos

?- X is 3+4.

X = 7

true

?- X+Y = 3+5.

X=3, Y=5

true

?- X = 3+5.

X=3+5

true

?- X = 8, X = 3+5.

false

?- X = 8, X is 3+5.

X = 8

true

?- 3 == 1+2. comparación de términos

false

?- 3 === 1+2. comparación numérica

True

Prolog : Operadores

Instanciación de variables con operadores

- Una variable está instanciada cuando *Prolog* le ha asignado un valor.
- Los operadores aritméticos y relacionales necesitan que *todas* las variables implicadas en la expresión correspondiente estén instanciadas en el momento en que se realiza la evaluación.

?- 5 is X + 4	¡Error! X no está instanciada y no se puede realizar la suma
?- Y is 1 + 4.	Correcto: Y = 5
?- X is 1, Y is X + 4.	Correcto: Y = 5
?- 5 =:= X + 4.	¡Error! X no está <i>aún</i> instanciada y no se puede realizar la suma
?- 5 =:= X + 4, X = 1	¡Error! X no está instanciada y no se puede realizar la suma
?- X = 1, 5 =:= X + 4	Correcto: X unifica con 1 y se puede realizar la suma para compararla con 5
?- X = 1, Y = 5, Y =:= X + 4 Correcto: X =1 y Y = 5	

Caso de estudio

- Con los datos siguientes:

Vendedores			
Código	Apellido	Nombre	Código sucursal
1111	Linares	Indiana	1
2222	Marchetti	Sofía	1
3333	Luchetti	María	1
4444	Bravi	Emilio	2
5555	Luchetti	Pedro	2

Sucursales	
Código de sucursal	Denominación
1	Buenos Aires
2	Córdoba

- Responder:Cuál es el nombre (denominación) de la sucursal de un vendedor determinado.

Caso de estudio

```
/*hechos*/
```

```
/*vendedor(codigo, apellido, nombre, codigo sucursal)*/
```

```
vendedor(1111,'Linares', 'Indiana',1).
```

```
vendedor(2222,'Marchetti', 'Sofía',1).
```

```
vendedor(3333,'Luchetti', 'María',1).
```

```
vendedor(4444,'Bravi', 'Emilio', 2).
```

```
vendedor(5555,'Luchetti', 'Vera',2).
```

```
/*sucursal(codigo, nombre)*/
```

```
sucursal(1,'Buenos Aires').
```

```
sucursal(2,'Córdoba').
```

Caso de estudio

/*reglas*/

/*Cuál es el nombre de la sucursal de un vendedor determinado? */

Variables de resultado

regla1(CV,NS) :- vendedor(CV,_,_,CS),
 sucursal(CS,NS).

Variable de enlace

Ejemplo de unificación:

vendedor(CV = 1111 y CS = 1) y sucursal(CS=1 y NS="Buenos Aires")

la variable CS se unifica con 1 en ambos casos

Caso de estudio

- **Evaluación:**
? regla1(1111,NS)
NS = "Buenos Aires"
? regla1(1,NS)
NS = libre
False
? regla1(1111,"Buenos Aires")
True
? regla1(1111,"Córdoba")
False
? regla1(CV,NS)
CV=1111 NS="Buenos Aires";
CV=2222 NS="Buenos Aires".