



UNIVERSIDAD POLITÉCNICA

SALESIANA

ECUADOR

Nombre: David Leon
9no Ciclo

SISTEMAS EXPERTOS

QUE ES PROLOG?

- Prolog es un lenguaje de programación declarativo que se utiliza para resolver problemas en los que existen objetos y las relaciones entre ellos.
- Básicamente consiste en :
 - *Declarar hechos sobre los objetos y sus relaciones,*
 - *Definir reglas sobre dichos objetos y relaciones*
 - *Hacer consultas.*



Historia



- Prolog fue creado en los años 80 en la Universidad de Provenza por Alain Colmerauer y Philippe Roussel. Su nombre se deriva del francés programmation en logique.



Sintaxis

■ Constantes

- *Numeros puede representar por numeros enteros y reales*

number(286). number(2.48).

- *Tenemos los **Objetos** que representar las propiedades o relaciones, se debe empezar con minúscula*

string(david). string(nombre).

■ Variables

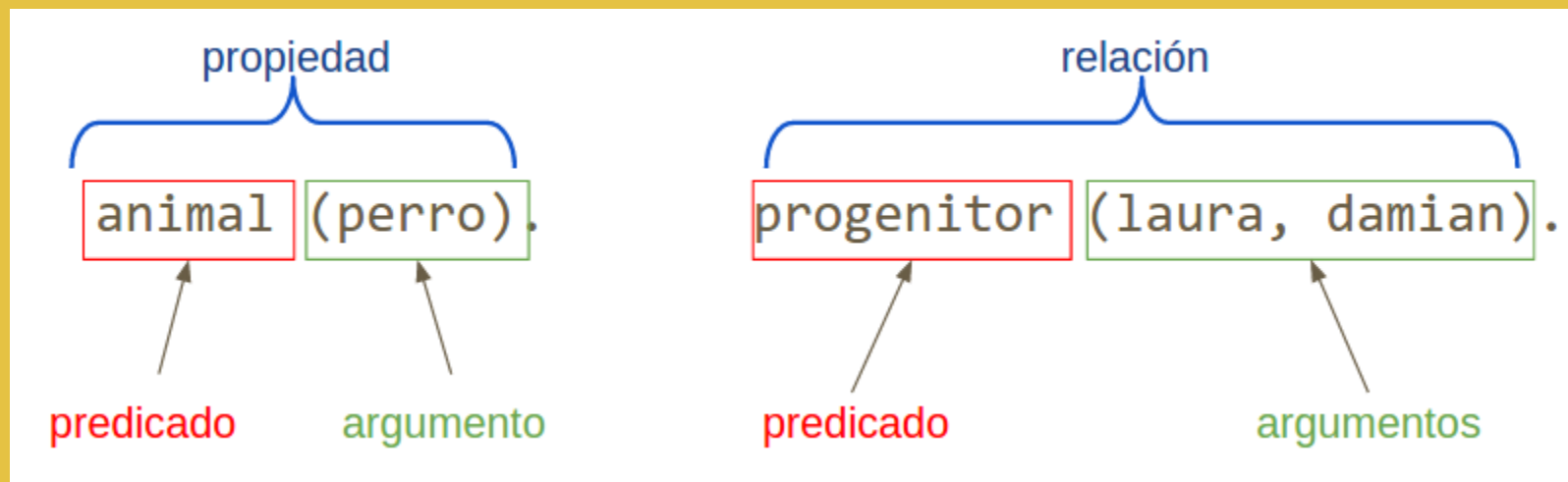
- *Se representa como cadenas o letras, siempre comienza con una mayuscula*

variabl(X). variabl(Y).

■ Estructuras

- *Son términos compuestos por otros términos. Se construyen mediante un símbolo*

punto(31,25)



realMadridBBC(bale,benzema,cristiano).



Es un hecho que es
Relación

Reglas

- Conjugaciones: El operador logico AND se utiliza como (,).

```
tia(X,Y):-hermana(X,Z),padre(Z,Y).
```

Regla

AND

- Disyunciones: El operador logico OR , se utiliza el punto y coma(;))

```
hijo(X,Y):-padre(Y,X);madre(Y,X).
```

Regla

OR



Expresiones aritmeticas



Ejemplos de expresiones aritmeticas

```
mayor(X,Y):- X>Y.  
  
diferencia(X,Y,Z):- Z is X-Y.  
  
producto(X,Y,Z):- Z is X*Y.  
  
senoCuadrado(X,Z):- producto(sin(X),sin(X),Z1), Z is Z1.  
  
comparacionIs(X,Y) :- X is Y. %Evalua antes de comparar  
  
comparacion(X,Y) :- X == Y. %Comparación sintáctica
```

Expresión	Operación
X+Y	suma de X e Y
X-Y	X menos Y
X*Y	producto de X por Y
X/Y	cociente real de la división de X por Y
X//Y	cociente entero de la división de X por Y
X^Y	potencia entera de X a la Y
X**Y	potencia real de X a la Y
X mod Y	resto de la división entera de X por Y
abs(X)	valor absoluto de X
acos(X)	arco coseno de X
asen(X)	arco seno de X
atan(X)	arco tangente de X
cos(X)	coseno de X
exp(X)	exponencial de X; [eX]
ln(X)	logaritmo neperiano de X
log(X)	logaritmo en base 2 de X
sin(X)	seno de X
sqrt(X)	raíz cuadrada de X
tan(X)	tangente de X
round(X,N)	redondeo del real X con N decimales

Predicados predefinidos

- **var** El objetivo `var(X)` se cumple si `X` es una variable no instanciada.
- **Nonvar** El objetivo `nonvar(X)` se cumple si `X` es una variable instanciada
- **atom** El objetivo `atom(X)` se cumple si `X` representa un átomo.
- **integer** El objetivo `integer(X)` se cumple si `X` representa un número entero.
- **atomic** El objetivo `atomic(X)` se cumple si `X` representa un entero o un átomo.
- **Predicado** Función
- **!(cut)** Fuerza al sistema a mantener ciertas elecciones que ha realizado.
- **true** Este objetivo siempre se cumple.
- **fail** Este objetivo siempre fracasa.
- **not** El objetivo `not(X)` se cumple si fracasa el intento de satisfacer `X`. El objetivo `not(X)` fracasa si el intento de satisfacer `X` tiene éxito. Es similar a la negación en la lógica de predicados.
- **repeat** forma auxiliar para generar soluciones múltiples mediante el mecanismo de reevaluación.
- **call** Se cumple si tiene éxito el intento de satisfacer `X`.
- **;** Especifica una disyunción de objetivos
- **,** Especifica una conjunción de objetivos

- Predicados de lectura y escritura

- Predicado Función

- **write** escribe el término X en la consola de salida.

- **nl** genera una nueva línea en la consola de salida.

- **read** lee el siguiente término en la consola de entrada.

- **display** funciona exactamente igual que write, excepto que

Listas

- En la siguiente imagen, se tiene que la cabeza es **a** y la Cola **[b,c,d]**, y en prolog se denota como **[A | B]**

- **Ejemplo**

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



Y consultamos: QUIEN ES EL ULTIMO DE LA LISTA?

`ultimo(X,[a,b,c,d]).`

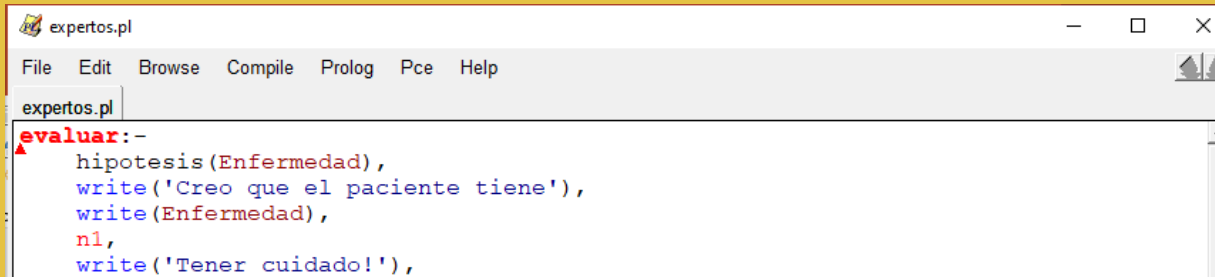
Lista	Cabeza	Cola
[a,b,c,d]	a	[b,c,d]
[a]	a	[]
[]	no tiene	no tiene
[[a,b],c]	[a,b]	[c]
[a,[b,c]]	a	[[b,c]]
[a,b,[c,d]]	a	[b,[c,d]]

SWINGS

- PROLOG TIENE SU PROPIA INTERFAZ , TANTO PARA LINUX Y PARA WINDOWS.
- Podemos descargarnos del siguiente link <https://www.swi-prolog.org/download/stable>

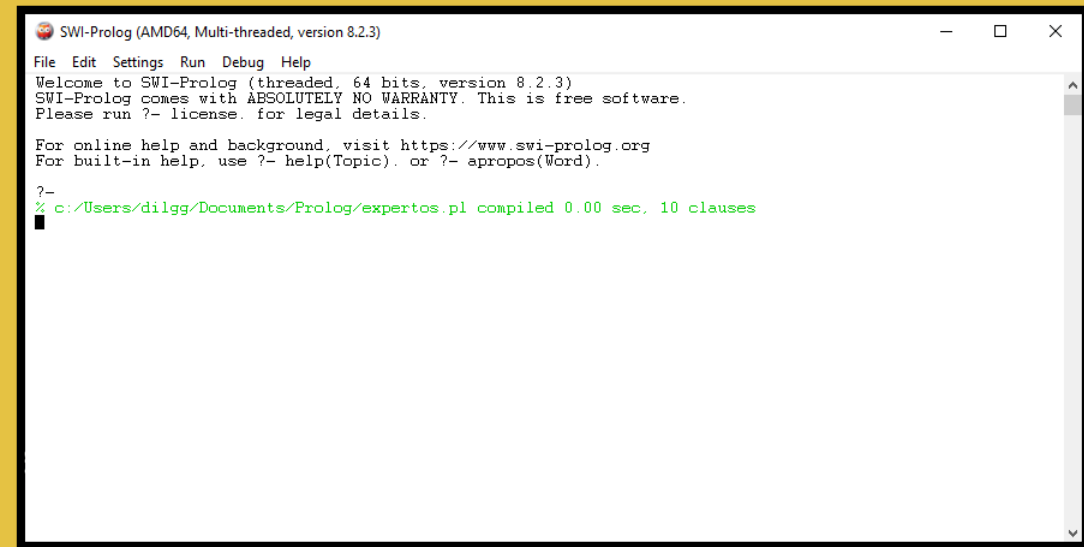
Binaries		
	11,825,869 bytes	SWI-Prolog 8.2.3-1 for Microsoft Windows (64 bit) Self-installing executable for Microsoft's Windows 64-bit editions. Requires at least Windows 7. See the reference manual for deciding on whether to use the 32- or 64-bits version. This binary is linked against GMP 6.1.1 which is covered by the LGPL license. SHA256: b4297d1d60ce050daf0d07764487370ef475cad70b54a96b074f83138d792cd6
	11,468,354 bytes	SWI-Prolog 8.2.3-1 for Microsoft Windows (32 bit) Self-installing executable for MS-Windows. Requires at least Windows 7. Installs swipl-win.exe and swipl.exe . This binary is linked against GMP 6.1.1 which is covered by the LGPL license. SHA256: a05901d850c8e3138dab176ab10fab6b44ba452d35c7076b65d213dfcd89c209
	27,660,126 bytes	SWI-Prolog 8.2.3-1 for MacOSX 10.12 (Sierra) and later on intel Installer with binaries created using Macports . Installs /opt/local/bin/swipl. Needs xquartz (X11) and the Developer Tools (Xcode) installed for running the development tools SHA256: 235a08a3fcc4eecfa023a8e4f89a78ce54f7723120883f9944efaaaf89930f444

Tenemos aqui nuestra interfaz descargada ,
la una para modificar, y la otra Ventana
para correr nuestro programa



A screenshot of a Prolog editor window titled 'expertos.pl'. The window has a menu bar with 'File', 'Edit', 'Browse', 'Compile', 'Prolog', 'Pce', and 'Help'. The main text area contains the following Prolog code:

```
evaluar:-  
    hipotesis(Enfermedad),  
    write('Creo que el paciente tiene'),  
    write(Enfermedad),  
    nl,  
    write('Tener cuidado!'),
```



A screenshot of the SWI-Prolog console window, titled 'SWI-Prolog (AMD64, Multi-threaded, version 8.2.3)'. The window shows the following text:

```
File Edit Settings Run Debug Help  
Welcome to SWI-Prolog (threaded, 64 bits, version 8.2.3)  
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.  
Please run ?- license. for legal details.  
  
For online help and background, visit https://www.swi-prolog.org  
For built-in help, use ?- help(Topic). or ?- apropos(Word).  
  
?-  
% c:/Users/dilgg/Documents/Prolog/expertos.pl compiled 0.00 sec, 10 clauses
```

- **Ventajas**

- El código es mucho más sencillo y fácil de modificar
- Es fácil de programar

- **Desventajas**

- Como es Amplio prolog , si nos toma un poco de tiempo en aprender
- Se debe aclarar los hechos por que pueden haber soluciones erroneas.

Conclusiones

- En conclusion Podemos ver que el lenguaje de programacion Prolog es facil de implementar y nos va ayudar en muchas ambitos profesionales, como finanzas, Comunicaciones, medicina, ingenieria, etc.

Bibliografia

- <http://ferestrepoca.github.io/paradigmas-de-programacion/proglogica/tutoriales/prolog-gh-pages/index.html>