



The background is a dark blue field with a faint green grid. A complex network of thin white lines connects numerous small blue and green dots, creating a web-like structure. In the center, there is a detailed, grayscale image of a computer circuit board, showing various components like chips and connectors. The text "PROLOG (Teoría)" is centered over this image in a white, sans-serif font.

PROLOG (Teoría)

SBC



SWI Prolog

PROLOG

Proviene del francés ***Programation et Logique***. Es un lenguaje de programación para inteligencia artificial que hace uso de la lógica de predicados. Fue desarrollado por **Philippe Roussel** y **Alain Colmerauer** en la Universidad de Marsella en 1972. No obstante, el primer compilador lo construyó el matemático **Robert Kowalski**, del Imperial College de Londres. Al principio era sólo un lenguaje interpretado; luego, a mediados de los 70's, **David Warren** desarrolló un compilador que traducía Prolog a un conjunto de instrucciones de una máquina abstracta denominada ***Warren Abstract Machine*** (WAM).

SBC

PROLOG

Las primeras versiones del lenguaje diferían en muchos aspectos de sus sintaxis, empleándose mayormente como forma normalizada el dialecto propuesto por la **Universidad de Edimburgo**¹¹³ hasta que en 1995 se estableció el estándar llamado ISO-Prolog (ISO/IEC 13211-1). Prolog se enmarca en el paradigma de los lenguajes lógicos y declarativos, lo que le diferencia enormemente de otros lenguajes más populares tales como Fortran, Pascal, C o Java. Prolog, Es un lenguaje de programación centrado alrededor de un pequeño conjunto de mecanismos básicos, incluyendo la unificación de patrones, estructuración de datos basado en árboles y retroceso automático. Es apropiado para problemas que involucran objetos estructurados y relaciones entre ellos.

SBC

Programación en Prolog

La programación está basada en la **lógica de primer orden** LPO (o **lógica de predicados**). Consiste en definir relaciones y preguntar acerca de dichas relaciones. El programa está conformado por cláusulas. Las cláusulas pueden ser de 3 tipos: hechos, reglas y preguntas.

Una relación puede ser especificada por medio de hechos o por reglas que definen la relación. Un procedimiento es un conjunto de cláusulas acerca de una misma relación.

Preguntar acerca de relaciones en Prolog es similar a interrogar a una base de datos (*query*). Prolog responde con un conjunto de objetos que satisfacen la pregunta.



SBC

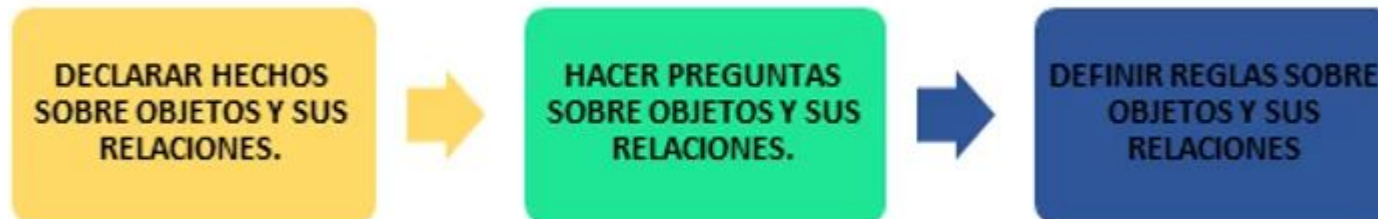
Un programa PROLOG consiste en un conjunto de proposiciones (o sentencias), de la forma:

$A :- B_1, \dots, B_n$. Donde $n \geq 0$.

- Cuando $n > 0$, la sentencia se escribe $A :- B_1, \dots, B_n$. Se denomina regla.
- Si $n = 0$, la sentencia se escribe A . Se denomina hecho.

Un programa PROLOG consiste en un conjunto de hechos (afirmaciones simples) y de reglas que afirman:

El hecho A es cierto si son ciertos los hechos B_1 , y B_2 , y ... y B_n .



SBC

Las reglas sirven para deducir nuevos hechos a partir de otros. A un programa PROLOG se le hacen preguntas. Una pregunta se escribe como:

?A1, A2, ..., Am. Siendo $m > 0$.

Informalmente, dicha pregunta se leerá:

¿Son ciertos los hechos A1 y A2 y ... y Am?

SBC

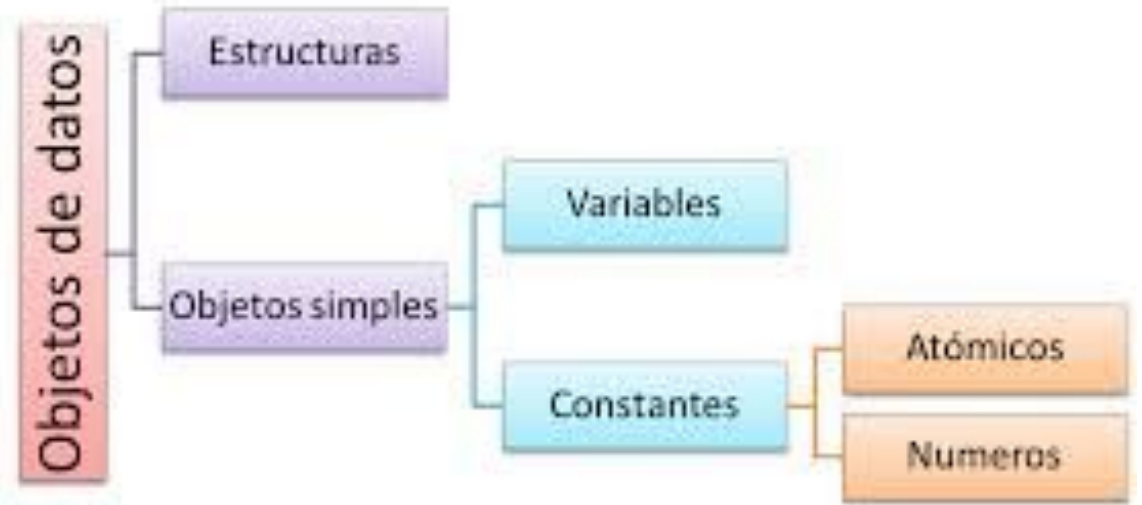
Prolog, para establecer si un objeto definido satisface una pregunta, sigue un proceso complejo que involucra inferencia lógica y un mecanismo de retroceso (*backtracking*). Esto es hecho automáticamente y está oculto al usuario. Un programa en Prolog tiene dos tipos de significados:

- Declarativo.- Asociado con las relaciones definidas en el programa.
- Procedimental.- Asociado con la forma en que Prolog evalúa las relaciones definidas, para determinar las respuestas.

SBC

Objetos de Datos

Prolog tiene como datos: objetos simples y estructuras. Esto se ilustra en la siguiente Figura.



SBC

Átomos.- Sus nombres inician con letra minúscula, o son cadenas de caracteres encerrados entre comillas:

```
juan ana x_25 'Juan' 'Ana' 'Facultad de Sistemas'
```

Números.- Ya sean estos Enteros o Reales:

```
4 1356 0 -98
```

```
3.14 -0.0035 100.2
```

Los nombres de los objetos Variables, siempre inician con una letra mayúscula o con una subraya:

```
X Juan Ana Resultado _x_25 _123
```

SBC

•**Estructuras**

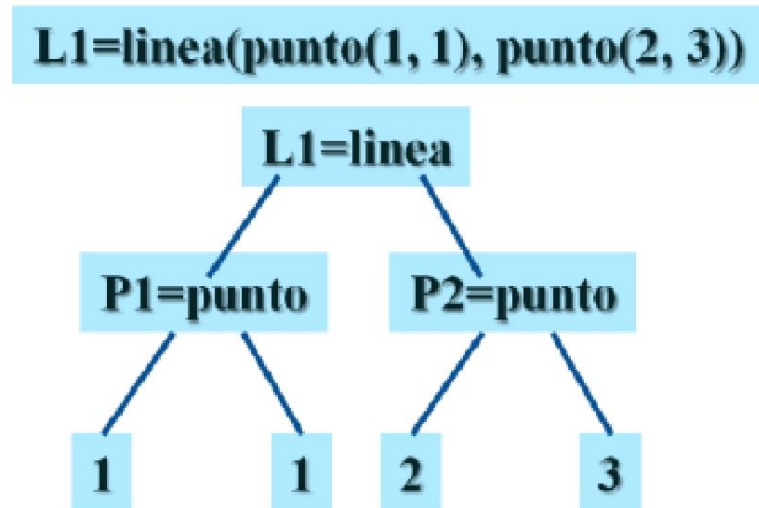
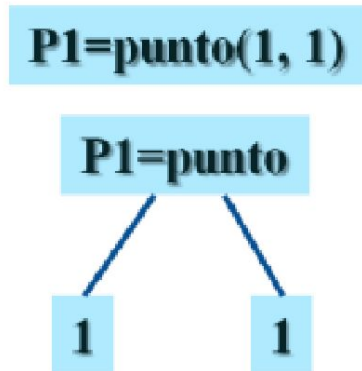
• Las estructuras son objetos que tienen varios componentes. Sintácticamente, en Prolog, todos los objetos de datos se denominan términos. Los componentes pueden ser objetos simples o estructuras. A pesar de estar conformados por varios componentes, las estructuras en Prolog son tratadas como objetos simples.

• Para combinar los componentes (argumentos) en un objeto simple, se usa un ***functor***:

- fecha(1, marzo, 2010).
- fecha(Dia, Mes, 2010).
- punto(6, 4).
- punto(X, Y).

SBC

Todas las estructuras pueden ser visualizadas como árboles, en el que la raíz es el *functor* y sus ramificaciones son sus componentes.



Un *functor* está definido por:

El nombre, cuya sintaxis es igual a la de los átomos.

El *arity*, esto es el número de argumentos.

SBC

Listas en Prolog

La lista es una estructura de datos que contiene una secuencia de cero o más objetos encerrados entre corchetes y separados por comas. Las listas pueden ser vacías o no.

- [jugar, estudiar, viajar, coleccionar] []

Una lista no vacía tiene 2 partes el encabezado y la cola. El encabezado puede ser un árbol o una variable y la cola tiene que ser una lista.

- .(Encabezado, Cola)

SBC

La lista puede ser vista como representación de un árbol.

- `.(jugar, .(estudiar, .(viajar, .(coleccionar, []))))`



SBC

Operaciones con Listas

Prolog proporciona la notación de la barra vertical, para facilitar la operación con listas.

- $[a, b, c] = [a \mid [b, c]] = [a, b \mid [c]]$

Un objeto X es miembro de una lista, si se cumple que:

- X encabeza la lista; o,
- X se encuentra en la Cola de la lista.

`miembro(X,[X | Cola]).`

`miembro(X,[Cabeza | Cola]):-`

`miembro(X,Cola).`

SBC

La Unificación

Para determinar la veracidad de sentencias compuestas por predicados y conectivos lógicos, es necesario evaluar la veracidad de cada uno de sus términos y luego aplicar los operadores lógicos a los resultados.

Un predicado componente resuelve a verdadero si se identifica con un axioma de la base de conocimiento. La unificación es un proceso que computa las sustituciones apropiadas para determinar si dos objetos coinciden o se identifican.

SBC

El Proceso de Unificación se realiza en base a las siguientes condiciones:

- Todo predicado que no contenga variables en sus argumentos, debe tener un axioma que se identifique totalmente, para considerarlo como verdadero.
- Si un predicado tiene una variable, ésta debe ser asociada a un valor determinado. Se seleccionan todos los axiomas de la base que se identifiquen con el patrón en todo excepto por la variable. La variable es asociada con el valor del argumento de la posición correspondiente del axioma.
- Si más de un axioma se identifica con el predicado dado, todos los valores asociados son considerados y tratados separadamente.
- El proceso de identificación continúa asumiendo que el valor de la variable es el valor asociado, en cualquier lugar que ésta aparezca.
- Los conectivos lógicos son aplicados a todos los predicados, para determinar la veracidad de la sentencia dada.

SBC

Problema de Ejemplo en PROLOG

Dado un conjunto de personas, hombres y mujeres, que tienen ciertos *hobbies*, se trata de establecer cuáles pueden conformar una pareja ideal, suponiendo que para ello deben compartir al menos uno de los *hobbies* declarados.

Para resolver este problema se utilizará **SWI-Prolog**¹¹⁴, que es una versión de software libre (*Lesser GNU Public License*). Además, para usar **SWI-Prolog**¹¹⁵ bajo **MS-Windows**, se recomienda también instalar el entorno de desarrollo **SWI-Prolog-Editor**¹¹⁶.

<http://www.swi-prolog.org/>

<http://www.swi-prolog.org/versions.html>

<http://lakk.bildung.hessen.de/netzwerk/faecher/informatik/swiprolog/indexe.html>

SBC

En primer lugar se definen como axiomas al grupo de personas, donde cada una está representada por un nombre, su género y una lista de hobbies.

Problemas Propuestos

A continuación se codifica la regla de inferencia: Se puede decir que es una pareja ideal si existe un hombre y una mujer que comparten al menos uno de sus hobbies (interés común). Para determinar si el hombre y la mujer tienen algún interés común, se analiza el contenido de las dos listas de hobbies, buscando alguno que sea miembro tanto de la lista del hombre, como de la lista de la mujer. Finalmente, se imprime en la salida los resultados de las parejas que se consideran ideales.



SBC

Dilema del Granjero

Un granjero desea cruzar un río llevando consigo un lobo, una oveja y una carga de col. El bote disponible solo permite el transporte de hasta dos objetos a la vez. Si no puede dejar solos a la oveja y al lobo o a la cabra y la carga de col, ¿cómo puede hacer para cruzar el río sin contratiempos?

