



# Tecnologías de Programación

Paradigma Lógico  
ProLog



# Programación Lógica

## Programación Imperativa

Términos de máquina, no humanos

Programar es: codificar COMO

Traducir conocimientos a la máquina

Es imposible describir el problema

## Programación Declarativa

Paradigmas funcional y lógico

Programar es: describir el QUE

Expresando y organizando el conocimientos

Ejecutando lo expresado



# Programación Lógica

## Programa

- Conocimientos acerca del problema

- Expresado como conjunto de axiomas lógicos

## Como invocar el programa

- Proveyendo un nuevo axioma lógico (el cual se tratará de probar)

## Ejecución

- Tratar de probar la sentencia dada en la invocación



# Programación Lógica

## **Campos de Aplicación**

Sistemas Expertos

Demostración automática de Teoremas

Inteligencia artificial

Cualquier otro campo de acción en donde la naturaleza del problema se pueda expresar como un cuerpo de predicados lógicos



# Programación Lógica

Supongamos las siguientes declaraciones:

- a) Toda madre ama a sus hijos
- b) maría es madre y juan es hijo de maría

Aplicando algo de razonamiento podemos inferir que:

- c) maría ama a juan



# Programación Lógica

Las sentencias en lenguaje natural expresadas en “a” y “b” describen un universo de personas y relaciones entre dichos individuos.

El ejemplo refleja la idea principal de la programación lógica

Describir universos de objetos y relaciones sobre los mismos, y aplicar un sistema de programación para inferir conclusiones como la “c”



# Programación Lógica

## Alfabeto

### Propósito

Brindar un conjunto de símbolos con los que trabajar

Los símbolos pueden denotar

Individuos: constantes

Ej.: `juan` es usado para denotar a la persona Juan

Relaciones: predicados

Aridad: número de individuos involucrados

Relación: conjunto de n-tuplas de individuos

Ej.: `madre_de(juan, maría)`, `mujer(maría)`, etc



# Programación Lógica

## Formalización de sentencias

Se debe definir en forma precisa las sentencias que conformarán el programa

## Lenguaje formal

- Elimina ambigüedad

- Factible de ser interpretado por un computador





# Programación Lógica

## Alfabeto

### Más símbolos

Referencias a individuos no específicos: variables

Ej.: si X es madre de juan entonces X ama a juan

Referencias a todos los individuos (Como decir “**TODOS**”?): existe el cuantificador universal  $\forall$  (se lee “para todo”)

Ej.: para todo X, si X es madre de Y, entonces X ama a Y



# Programación Lógica

## Alfabeto

### Más símbolos

La sentencia se cumple al menos para algún X:  
Cuantificador existencial  $\exists$

Ej.: existe al menos un X tal que X es madre de Juan

Conectores lógicos: combinan elementos para  
crear nuevos elementos más complejos

$\wedge$  (conjunción),  $\vee$  (disyunción),  $\neg$  (negación),  $\rightarrow$   
(implicación) y  $\leftrightarrow$  (equivalencia lógica).



# Programación Lógica

## Términos

Las sentencias en lenguaje natural consisten de palabras donde los objetos son representados por sustantivos (nombres). En el lenguaje de lógica de predicados, los objetos serán representados por términos.

Denotan objetos del dominio del problema



# Programación Lógica

## Términos

## Ejemplos

Constantes: juan, maría, 1, etc...

Variables: X, Fecha, Lista1, etc...

Funtores: lista(cabecera, cola)



# Programación Lógica

## Fórmulas Lógicas

### Ejemplo

- a) Todas las madres aman a sus hijos
- b) maría es madre y juan es hijo de maría
- c) maría ama a juan

Las sentencias podrían formalizarse de la siguiente forma:

$$\forall X (\forall Y ((\text{madre}(X) \wedge \text{hijo\_de}(Y, X)) \rightarrow \text{ama}(X, Y)))$$
$$\text{madre}(\text{maría}) \wedge \text{hijo\_de}(\text{juan}, \text{maría})$$
$$\text{ama}(\text{maría}, \text{juan})$$



# Programación Lógica

## Fórmulas Lógicas

Fórmulas de la forma  $p(t_1, \dots, t_n)$  son llamadas fórmulas atómicas o simplemente átomos

Fórmulas en la forma  $(F \rightarrow G)$  serán escritas en la forma  $(G \leftarrow F)$  para adoptar una sintaxis similar a la de Prolog



# Programación Lógica

## Fórmulas Lógicas

Fuerza de vinculación de los conectores lógicos (de más a menos):

$\neg$  (negación),  $\forall$  (universal),  $\exists$  (existencial)

$\vee$  (disyunción)

$\wedge$  (conjunción)

$\rightarrow$  (o como acordamos  $\leftarrow$ )

$\leftrightarrow$  (equivalencia lógica)



# Programación Lógica

## Semántica

Las fórmulas hacen referencia a algún universo y en ese marco pueden ser V o F

El significado de una fórmula se define en relación a un modelo del mundo real





# Programación Lógica

## Inferencia Lógica

En las fórmulas ya vistas

$$\forall X (\forall Y ((\text{madre}(X) \wedge \text{hijo\_de}(X, Y)) \rightarrow \text{ama}(X, Y)))$$
$$\text{madre}(\text{maría}) \wedge \text{hijo\_de}(\text{maría}, \text{juan})$$
$$\text{ama}(\text{maría}, \text{juan})$$

se puede apreciar que de un conjunto dado de fórmulas (1 y 2) se produjo una nueva fórmula (3) a través de conclusiones lógicas. Al proceso que lleva a cabo éstos descubrimientos lo denominamos inferencia lógica. Las fórmulas 1 y 2 son las premisas, y la 3 es la conclusión.



# Programación Lógica

El programa se escribe como conjunto finito de fórmulas en lenguaje formal

La sintaxis usada se restringe por cuestiones de eficiencia

Sentencias de éste tipo expresan:

- Que una relación se da entre individuos (hechos)

- Que una relación se da entre individuos provistos por otras relaciones (reglas)



# Programación Lógica

## Cláusulas Definidas

Llamaremos cláusulas definidas, a las fórmulas con el siguiente formado (sintaxis):

$$A0 \leftarrow A1 \wedge \dots \wedge A_n$$

O equivalentemente

$$A0 \vee \neg A1 \vee \dots \vee \neg A_n$$

donde  $A0, \dots, A_n$  son formulas atómicas, y todas las variables que ocurran en una fórmula son (implícitamente) cuantificadas en forma universal.

Los hechos son cláusulas definidas en donde  $n = 0$ .

$A0$  es llamado cabecera de la cláusula y  $(A1 \wedge \dots \wedge A_n)$  cuerpo de la misma.



# Programación Lógica

## Ejemplo

Tomás es hijo de Juan

Ana es hija de Tomás

Juan es hijo de Marcos

Alicia es hija de Juan

El nieto de una persona es el hijo de un hijo de dicha persona



# Programación Lógica

## Ejemplo

Formalizando el ejemplo tenemos los siguiente hechos:

hijo(tomás, juan)

hijo(ana, tomas)

hijo(juan, marcos)

hijo(alicia, juan)



# Programación Lógica

## Ejemplo

Para la última sentencia:

Para todo X e Y, nieto(X, Y) si existe un Z tal que hijo(X, Z) e hijo(Z, Y)

Formalizando y transformando:

$$\forall X \forall Y (\text{nieto}(X, Y) \leftarrow \exists Z (\text{hijo}(X, Z) \wedge \text{hijo}(Z, Y)))$$

$$\forall X \forall Y (\text{nieto}(X, Y) \vee \neg \exists Z (\text{hijo}(X, Z) \wedge \text{hijo}(Z, Y)))$$

$$\forall X \forall Y (\text{nieto}(X, Y) \vee \forall Z \neg (\text{hijo}(X, Z) \wedge \text{hijo}(Z, Y)))$$

$$\forall X \forall Y \forall Z (\text{nieto}(X, Y) \vee \neg (\text{hijo}(X, Z) \wedge \text{hijo}(Z, Y)))$$

$$\forall X \forall Y \forall Z (\text{nieto}(X, Y) \leftarrow (\text{hijo}(X, Z) \wedge \text{hijo}(Z, Y)))$$



# Programación Lógica

## Ejemplo

Entonces tenemos:

hijo(tomás, juan)

hijo(ana, tomas)

hijo(juan, marcos)

hijo(alicia, juan)

$\text{nieto}(X, Y) \leftarrow \text{hijo}(X, Z) \wedge \text{hijo}(Z, Y)$



# Programación Lógica

## Programa Definido

Un programa definido es un conjunto finito de cláusulas definidas

El programador trata de describir el modelo previsto por medio de cláusulas definidas: Hechos (no hay cuerpo) y Reglas

Estas reglas son usadas por el computador para arribar a conclusiones acerca del modelo

El conjunto de consecuencias lógicas de un programa puede ser infinito, sin embargo se espera que el usuario indague al programa sobre aspectos del modelo en forma selectiva (objetivos)





# Programación Lógica

## Objetivos Definidos

La forma en la que el usuario indagará al programa también es por medio de cláusulas, las cuales en principio van a tener el siguiente formato:

$$\forall(\neg(A1 \wedge \dots \wedge A_m))$$

Estas fórmulas son denominadas objetivos definidos, y usualmente se escriben como:

$$\leftarrow A1, \dots, A_m$$

En donde las  $A_i$  son fórmulas atómicas denominadas sub-objetivos



# Programación Lógica

## Ejemplos de consultas al programa y las correspondientes definición de Objetivos

Es Ana hija de Tomás:  $\leftarrow$  hijo(ana, tomás)

Como no hay variables se obtiene simplemente “Si”

Quien es nieto de Ana:  $\leftarrow$  nieto(X, ana)

Como el programa no contiene información acerca de nietos de Ana, la respuesta es “Nadie”



# Programación Lógica

Fin de la Introducción