

# Clase de Inteligencia Artificial: Tarea de Prolog

## **Profesores:**

Luis A. Pineda Cortés, IIMAS, UNAM  
Arturo Rodríguez García, PCIC, UNAM

Septiembre 3, 2015

## **Condiciones de Entrega**

**Fecha de entrega** Jueves 17 de septiembre, justo al inicio de la clase.

**Formato:** Trabajo impreso con las respuestas y/o código de cada ejercicio. En cada inciso, reporta el código en tu base lógica (archivo .pl) y muestra algunos ejemplos significativos de consultas realizadas desde el listener de Prolog.

**Observaciones:** La tarea es individual. Utilizar SWI Prolog versión 7.1.20 o superior.

## **Descripción**

Realizar los siguientes ejercicios:

1. Cargar los siguientes hechos:

```
hombre(bart).  
mujer(lisa).  
mujer(maggie).  
hombre(homero).  
mujer(marge).  
hombre(abraham).  
mujer(selma).  
mujer(patty).  
mujer(mona).  
hombre(herb).  
hombre(clancy).  
mujer(jackeline).  
mujer(ling).  
padre(bart,homero).  
padre(lisa,homero).  
padre(maggie,homero).  
padre(homero,abraham).  
padre(herb,abraham).  
padre(marge,clancy).
```

```

padre(patty,clancy).
padre(selma,clancy).
madre(bart,marge).
madre(lisa,marge).
madre(maggie,marge).
madre(homero,mona).
madre(herb,mona).
madre(marge,jackeline).
madre(patty,jackeline).
madre(selma,jackeline).
madre(ling,selma).

```

La interpretación del predicado padre(X,Y) es Y es el padre de X.  
 La interpretación del predicado madre(X,Y) es Y es la madre de X.

Agregar las siguientes reglas:

- a) abuelo(X,Y) que es verdadera si Y es abuelo(forzosamente varón) de X.
  - b) abuela(X,Y) que es verdadera si Y es abuela de X.
  - c) nieto(X,Y) que es verdadera si Y es nieto(forzosamente varón) de X.
  - d) nieta(X,Y) que es verdadera si Y es nieta de X.
  - e) hermano(X,Y) que es verdadera si Y es hermano(forzosamente varón) de X.
  - f) hermana(X,Y) que es verdadera si Y es hermana de X.
  - g) tia(X,Y) que es verdadera si Y es tía de X.
  - h) primo(X,Y) que es verdadera si Y es primo(forzosamente varón) de X.
  - i) sobrino(X,Y) que es verdadera si Y es sobrino(forzosamente varón) de X.
  - j) pareja(X,Y) que es verdadera si X y Y han tenido por lo menos un hijo o hija en común.
2. Usando los mismos hechos del inciso anterior, realizar las siguientes consultas desde el listener. Explica el orden de los resultados obtenidos con ayuda de un diagrama de árbol.
- a) ?- mujer(X), hombre(Y).
  - b) ?- mujer(X),!, hombre(Y).
  - c) ?- mujer(X), hombre(Y),!.
3. Crear un predicado que cuente el número de elementos que tiene una lista. Ejemplo:  
 ?- cuenta([a,b,e,j,a],X).  
 X = 5.
4. Crear un predicado que obtenga el i-ésimo elemento de la sucesión de Fibonacci (0,1,1,2,3,5,8,13,21,...).  
 Ejemplo:  
 ?- fib(7,X).  
 X = 13.
5. Crear un predicado que cuente el numero de veces que un elemento aparece en una lista.  
 Ejemplo:  
 ?- cuenta(a,[b,a,n,a,n,a],X).  
 X = 3.

6. Crear un predicado que elimine todas las apariciones de un elemento en una lista. Ejemplo:  

```
?- elimina(a,[b,a,n,a,n,a],X).  
X = [b,n,n].
```
7. Crear un predicado que determine si una lista de letras es una palabra palíndroma. Ejemplo:  

```
?- palindroma([r,e,c,o,n,o,c,e,r]).  
true.  
?- palindroma([p,e,l,o,t,a]).  
false.
```
8. Crear un predicado que a partir de una lista atributo-valor, cuyo formato es [ a1=>v1 , a2=>v2 , ... , aN=>vN ] , obtenga el valor de un atributo que se le pida. Ejemplo:  

```
?- valor(forma, [ color=>azul , forma=>esferica , peso=>ligero , material=>plastico ] , X).  
X=esferica.
```

Para ello, tendrás que definir el operador => en tu base lógica.