



## Lenguajes de Inteligencia Artificial

*Segundo curso. Primer cuatrimestre*

Ingeniería Técnica en Informática de Gestión  
e Ingeniería Técnica en Informática de Sistemas  
Escuela Politécnica Superior



*Universidad de Córdoba  
Curso académico: 2009 - 2010*

---

### Práctica número 5: Introducción al lenguaje Prolog

1. Escribe un fichero denominado "datos.pl" que contenga los siguientes hechos

- *ama(juan, ana).*
- *ama(ana, miguel).*
- *ama(luis, isabel)*
- *ama(miguel, ana).*
- *ama(laura, juan).*
- *ama(isabel, luis).*

donde el predicado *ama(X, Y)* indica que *X ama a Y*.

- Escribe en PROLOG las siguientes preguntas
  - *¿A quién ama "Juan"?*
  - *¿Quién ama a "Ana"?*
  - *¿Quién ama a alguien?*
  - *¿Quién es amado por alguien?*
  - *¿Quiénes se aman mutuamente?*
  - *¿Quién ama sin ser correspondido?*
- Añade al fichero datos.pl una regla que permita describir a los "*amantes*", es decir, aquellas personas que se aman mutuamente.

2. Escribe un fichero denominado "familia.pl" que contenga los siguientes hechos:

- *hombre(antonio).*
- *hombre(juan).*
- *hombre(luis).*
- *hombre(rodrido).*
- *hombre(ricardo).*
- *mujer(isabel).*
- *mujer(ana).*
- *mujer(marta).*
- *mujer(carmen).*
- *mujer(laura).*
- *mujer(alicia).*

- Define hechos en los que se afirmen los siguientes enunciados:
  - *Antonio y Ana son matrimonio*

- *Juan y Carmen son matrimonio.*
  - *Luis e Isabel son matrimonio*
  - *Rodrigo y Laura son matrimonio.*
  - *Juan, Rodrigo y Marta son hijos de Antonio y Ana.*
  - *Carmen es hija de Luis e Isabel.*
  - *Ricardo es hijo de Juan y Carmen.*
  - *Alicia es hija de Rodrigo e Isabel.*
  - Define una regla que indique que "*matrimonio*" es reflexivo, es decir, si *X* e *Y* forma un matrimonio entonces *Y* y *X* también lo forman.
  - Define reglas para obtener:
    - *los nietos de una persona*
    - *los abuelos de una persona*
    - *los hermanos de una persona*
    - *los tíos de una persona*
    - *las tías de una persona*
    - *los primos de una persona*
    - *las primas de una persona*
    - *los suegros de una persona*
3. Escribe un programa que:
- Declare los siguientes hechos relativos a trabajadores de una empresa.
  - Utiliza el predicado *encargado\_de\_tarea(trabajador,tarea)*
    - *Miguel está encargado de las tareas de admisión, control y vigilancia.*
    - *Ricardo está encargado de las tareas de planificación y asesoramiento.*
    - *Alicia están encargada de la dirección y control.*
  - Defina reglas que permitan comprobar los siguientes hechos:
    - Si una tarea ha sido encargada a alguna persona. Utiliza el predicado *encargada(Tarea)*.
    - Si dos personas comparten alguna tarea, es decir, *comparten\_tarea(Persona1,Persona2)*.
4. Predicados sobre deportistas:
- Utiliza el predicado *juega(Persona,Deporte)* para declarar lo siguientes hechos relativos a deportistas:
    - Héctor juega al baloncesto.
    - Miguel juega al balonmano y rugby
    - Alicia juega al tenis, baloncesto y ajedrez.
  - Define un predicado que permita comprobar si dos personas juegan al mismo deporte.
5. Escribe un programa que permita realizar las siguientes operaciones aritméticas:
- *Máximo de dos números.*
  - *Mínimo de dos números.*
  - *Producto de los números comprendidos entre dos dados.*

- Área de un círculo dado el radio.
  - Área de un rombo dadas las diagonales mayor y menor.
  - Área de un trapecio dados la base mayor, la base menor y la altura.
6. Escribe un fichero denominado "lectores.pl" que contenga los siguientes hechos que utilizan la estructura nombre y el predicado lector:
- lector(nombre(ana,garrido,aguirre),mujer,31).
  - lector(nombre(marta,cantero,lasa),mujer,20).
  - lector(nombre(rodrigo,duque,soto),hombre,30).
  - Etc.
- Escribe en Prolog las siguientes preguntas:
    - ¿Hay lectores?
    - ¿Quiénes son lectores?
    - ¿Qué lectores son mujeres? y ¿hombres?
    - ¿Hay lectores con el mismo nombre y diferentes apellidos?
  - Escribe en Prolog una regla para comprobar si unos apellidos están repetidos.
  - Nota: Utiliza el predicado bagof y un predicado auxiliar para contar los elementos de una lista.
7. Utiliza el predicado "prestado" para definir algunos hechos en los que se indique que un libro (con título y autor) ha sido prestado a una persona (con nombre y apellidos).

Por ejemplo:

prestado(libro('Misericordia',autor('Benito','Pérez','Galdós'),  
persona('Almudena','Alegría','Sol'))).

donde libro y persona son "estructuras".

- Construye preguntas en las que se indique:
    - Si un lector tiene prestado algún libro.
    - Si un libro está prestado a alguien.
    - Si una persona es un escritor.
    - Si un escritor es leído.
    - Si existen autores leídos.
  - Define una regla para comprobar que un escritor es leído si alguno de sus libros está prestado.
8. Escribe un programa denominado "listas.pl" que contenga los predicados para realizar las siguientes operaciones:
- Crear una lista a partir de un número. Por ejemplo: crear(N,L) crearía la lista  $L = [0, 1, \dots, N]$ .
  - prefijo(P,L): comprueba si la lista P es el prefijo de L.
  - sufijo(S,L): comprueba si la lista S es el sufijo de L.
  - Media, máximo o mínimo de una lista.

Nota: utilícese "el corte" para mejorar la eficiencia de los predicados.

9. Escribe las reglas de un predicado denominado *lugar(X,L,P)* de forma que calcule la posición *P* en la que se encuentra *X* dentro de la lista *L*. Si *X* no está en *L* entonces *P* valdrá -1. La primera posición es la que ocupa el lugar cero. Por ejemplo:  
 $?lugar(a, [e, d, c, b, a], P).$   
 $P = 4$
10. Define una regla que permita identificar si un número es primo o no. Nota: utiliza el hecho de que un número es primo si no tiene divisores propios menores o iguales que su raíz cuadrada.
11. Crea una lista compuesta por los números primos menores que uno dado. Por ejemplo:  
 $?crear\_primos(10, L).$   
 $L = [2, 3, 5, 7]$
12. Crea una lista que contenga sólo los elementos que ocupan un lugar impar. Por ejemplo:  
 $?lugar\_impar([a, b, c, d, e], R).$   
 $R = [a, c, e]$
13. Escribe un predicado que permita obtener una lista compuesta por todos los elementos comunes a otras dos listas. Por ejemplo:  
 $?comunes([a, b, c, d], [b, e, d, f], R).$   
 $R = [b, d]$
14. Escribe un predicado que permita obtener una lista compuesta por la diferencia simétrica de dos listas. Por ejemplo:  
 $?diferencia\_simetrica([a, b, c, d, e], [c, d, e, f, g], R).$   
 $R = [a, b, f, g]$
15. Escribe un programa que:
- Declare los siguientes hechos relativos a monumentos y ciudades turísticas. Utiliza el predicado "situado(monumento, ciudad)":
    - *Puente Romano, Mérida.*
    - *Puente Romano, Córdoba.*
    - *Mezquita, Córdoba.*
    - *Alhambra, Granada.*
  - Defina reglas que permitan comprobar los siguientes hechos:
    - Si una ciudad posee un monumento entonces es turística. Utiliza el predicado *turistica(Ciudad)*.
    - Use el comando *setof* para contar el número de monumentos de una ciudad.
16. Escribe un programa que:
- Declare los hechos relativos a una base de datos de donantes: nombre, grupo sanguíneo, factor rh.

- *donante*(persona(juan, campos, ruiz), a, positivo).
- *donante*(persona(ana, lara, silva), ab, negativo).
- *donante*(persona(luis, luna, pachecho), ab, negativo).

- Declare reglas en las que se indique los donantes que pueden donar a otros donantes según la compatibilidad de sus grupos sanguíneos.
- Define reglas para el predicado *contar\_por\_grupo\_y\_factor* que permita contar todos los donantes de un grupo sanguíneo y factor rh específicos. Por ejemplo:
  - ? *contar\_por\_grupo\_y\_factor* (ab, negativo, N).
  - N = 2
 Nota: utilizar el predicado *bagof* y un predicado auxiliar que cuente los elementos de una lista.
- Escribe una regla que permita hacer las siguientes acciones consecutivas
  1. *Pedir por pantalla un grupo sanguíneo y un factor rh,*
  2. *Pedir por pantalla el nombre de un fichero,*
  3. *Y escribir en dicho fichero los nombres de todos los donantes que tengan el grupo sanguíneo y el factor rh indicados.*

17. Un árbol binario es representado por una lista de la forma

[raíz, hijo izquierdo, hijo derecho]

donde raíz es un átomo e hijo izquierdo e hijo derecho son árboles binarios.

- Constrúyanse predicados para:
  - *Escribir la lista en orden prefijo, sufijo e infijo.*
  - *Determinar la profundidad del árbol.*
  - *Comprobar si un elemento está en el árbol.*
  - *Determinar el número de nodos del árbol.*
  - *Determinar el número de hojas del árbol. Un nodo es una hoja si sus hijos izquierdo y derecho son listas vacías.*
- ¿Cómo se pueden redirigir las salidas de los predicados anteriores hacia un fichero de escritura?

18. Escribe un programa que permita contar los términos contenidos en un fichero y que escriba el resultado en un fichero.

19. Escribe un programa que sume los números de un fichero y que escriba el resultado en un fichero.