Programación Declarativa

Ingeniería Informática
Especialidad de Computación
Cuarto curso. Primer cuatrimestre

Escuela Politécnica Superior de Córdoba
Universidad de Córdoba

Curso académico: 2017 – 2018

Práctica número 6.- Introducción al lenguaje Prolog

Se deben presentar los ejercicios indicados con (*).

1. **Amantes**
   - Escribe un fichero denominado "**amantes.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 amantes.pl una regla que permita describir a los ***"amantes"***, es decir, aquellas personas que se aman mutuamente.

2. **Familia**
   - Escribe un fichero denominado "**familia.pl**" que contenga los siguientes hechos:
     - *hombre(antonio).*
     - *hombre(juan).*
     - *hombre(luis).*
     - *hombre(rodrigo).*
     - *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 el predidcado "*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. (*) Declara 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 está encargada de la dirección y control.*
   - Define 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. (*) Lectores
   - 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 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.

5. (*) Escribe un programa que permita realizar las siguientes operaciones aritméticas:
   - Área de un círculo.
   - Área de un trapecio.
   - Producto de los números comprendidos entre dos dados.

6. (*) Codifica el predicado **crear(N,L)** que permite crear una lista a partir de un número natural
   - Por ejemplo:
     ?crear(10,L)
     L = [0,1,...,10].

7. (*) **Números primos**
   - Define el predicado **primo(N)** para comprobar si el número N es primo o no
     - Nota: un número es primo si no tiene divisores propios menores o iguales que su raíz cuadrada.
   - Define el predicado **crear_primos(N,L)** para crear una lista compuesta por los números primos menores o iguales que el número N.
     - Por ejemplo:
       ?- crear_primos(10,L).
       L = [2,3,5,7]

8. (*) Codifica un predicado denominado, **invertir**, para invertir todos los elementos de una lista que puede contener **sublistas**:
   - Por ejemplo
     *?- **invertir**([1,2,3,4,5],R).*
     R = [5, 4, 3, 2, 1].

     *?- **invertir**([1,[2,3],[4,5]],R).*
     R = [[5, 4], [3, 2], 1].

   - Observación: codifica los siguientes predicados auxiliares
     - ***es_lista(X)***: comprueba si X es una lista
     - ***concatenar(L1,L2,L)***: L es el resultado de concatenar L1 y L2.

9. (*) **Monumentos**
   - Utiliza el predicado **monumento(Nombre,Localidad,Estilo)** para definir hechos asociados a los siguientes monumentos:
     - *Mezquita, Córdoba, Árabe*
     - *Medina Azahara, Córdoba, Árabe*
     - *Catedral, Santiago de Compostela, Románico*
   - Define el predicado **contar_monumentos(Localidad,N)** para contar los monumentos que hay en una localidad.
     - Por ejemplo
       *?- **contar_monumentos**("Córdoba",N).*
       N = 2.
   - Observación:
     - Utiliza el predicado ***bagof*** o ***findall***
     - Define un predicado auxiliar para **contar** para contar los elementos de una lista.

10. (*) Método de ordenación **mergesort**.
    - Codifica un predicado que permita ordenar una lista de números utilizando el método *mergesort*.
      - Ejemplo
        - Lista original: 5 4 1 3 2
        - División
          - ✓ Primera: 5 1 2 ; 4 3 ;
          - ✓ Segunda: 5 2 ; 1 ; ; 4 ; 3 ; ;
          - ✓ Tercera: 5 ; 2 ; ; 1 ; ; 4; 3 ; ;
        - Fusión:
          - ✓ Primera: 2 5 ; 1 ; ; 3 4 ;
          - ✓ Segunda: 1 2 5 ; 3 4 ;
          - ✓ Tercera: 1 2 3 4 5
    - Observación
      - Utiliza predicados auxiliares para
        - la "división" (*split*): reparte los elementos de una lista en dos listas, dependiendo de que ocupen un "lugar" par o impar
        - y para la fusión (*merge*): une de forma ordenada dos listas ordenadas

11. (*) Codifica los siguientes predicados sobre **listas numéricas**.
    - Media de una lista
    - Máximo de una lista.
    - Mediana de una lista ordenada
      - Si la lista no está ordenada entonces se debe ordenar.

12. (*) **Donantes de sangre**
    - Declara los hechos relativos a una base de datos de donantes que contiene la siguiente información:
      - *donante(persona(juan,campos,ruiz),a,positivo).*
      - *donante(persona(ana,lara,silva),ab,negativo).*
      - *donante(persona(luis,luna,pachecho),ab,negativo).*

    - Nota: ***persona*** es una estructura
- Escribe los hechos y las reglas que permitan comprobar si una persona ***puede donar*** sangre a otra teniendo en cuenta el grupo sanguíneo y el factor RH.
- 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 para ***contar*** 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.

13. (*) 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.
    - Define 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?

14. (*) **Ficheros y números primos**
    - Escribe un programa que lea los números contenidos en un fichero y que escriba los números **primos** en otro fichero.