

## Hechos, Reglas y Resolución

Este taller tiene por objetivos:

- Entender el proceso de resolución, utilizado por el [Prolog](#) para la solución de consultas.
- Entender y utilizar el operador *no es probable*, como mecanismo para resolver consultas negadas.
- Aprender a definir y utilizar reglas, las cuales posibilitan hacer programas haciendo uso de inferencias lógicas.

**Ejemplo:** Consideremos la siguiente base de datos:

```
mamifero(perro).  
mamifero(gato).  
mamifero(equidna).  
mamifero(ornitorrinco).  
oviparo(gallina).  
oviparo(ornitorrinco).  
oviparo(equidna).
```

Preguntas en forma afirmativa las puede resolver el Prolog,

```
?- mamifero(X).  
X = perro ;  
X = gato.  
  
?- oviparo(Y).  
Y = gallina ;  
Y = ornitorrinco.
```

pero preguntas en forma negativa no:

```
?- \+mamifero(X).  
false.  
  
?- \+oviparo(Y).  
false.
```

Esto ocurre porque el Prolog no puede particularizar variables libres, pues no tiene conocimiento del universo del discurso para los predicados que se le han definido.

Una solución parcial a este problema, para dominios finitos, es hacer explícito el universo del discurso en la base de datos. Podemos por ejemplo definir hechos de la siguiente forma:

```
animal(perro).  
animal(gato).  
animal(gallina).  
animal(ornitorrinco).  
animal(equidna).
```

De esta forma, podemos hacer una primera meta en forma positiva para particularizar la variable con un elemento del dominio y la segunda meta para la condición en forma negativa, por ejemplo:

```
?- animal(Y), \+oviparo(Y).  
Y = perro ;  
Y = gato ;  
false.  
  
?- animal(X), \+mamifero(X).  
X = gallina ;  
X = ornitorrinco.
```

## Ejercicio 1

Retomar la base de datos de la práctica anterior con los hechos `autorLibro/2`.

- Agregar los hechos necesarios para poder resolver consultas en forma negativa.
- Resolver la consulta: Quiénes no son los autores de un determinado libro?
- Resolver la consulta: Qué libros no ha escrito un determinado autor?

## Reglas en Prolog

Una regla es la representación en Prolog de una implicación lógica, es decir un condicional que siempre es verdadero.

Para construir una regla se indica primero la conclusión, el símbolo :- y las condiciones para que la conclusión sea verdadera. Por ejemplo para decir que X y Y son hermanos si son hijos del mismo padre, lo indicamos:

```
hermanos(X,Y) :- padre(Z,X), padre(Z,Y), X\=Y.
```

## Ejercicio 2

Extender la base de datos para incluir los siguientes hechos y reglas:

- Definir hechos personajeLibro/2 que relacionan un personaje, con el libro al que pertenece.
- Definir la regla personajesDelMismoLibro/2 que toma dos personajes y retorna true si aparecen en el mismo libro.
- Definir la regla autorDelPersonaje/2 que tomando como primer argumento el nombre del personaje y como segundo argumento el nombre de un autor, arroja un resultado true si el personaje fue creado por el autor.

## Ejercicio 3

Con base en los hechos y reglas definidos en el ejercicio 2 realizar las siguientes consultas:

- Cuáles son los personajes de un libro?
- Comprobar que dos personajes pertenezcan al mismo libro. Hacerlo tanto para el caso afirmativo como el negativo.
- Comprobar si un personaje es de un autor. Hacer una consulta para un caso afirmativo y otra para un caso negativo.
- Hacer una consulta que dado el nombre del autor, determine todos los personajes de ese autor.
- Hacer una consulta que dado el nombre de un personaje, encuentre los autores que no crearon el personaje.

## Informe

Nombrar la base de datos y las consultas siguiente el estándar **ApellidoNombre-id-BD6.pl** y **ApellidoNombre-id-consultas6.txt**. Remitir los 2 archivos al concluir la práctica. (No comprimir, ni subir a lugares distintos al aula digital).

No olvidar incluir los comentarios documentando el programa:

- Autor y fecha
- Descripción de cada uno de los predicados utilizados y sus argumentos.