

**Instituto Tecnológico y de Estudios Superiores de Monterrey**



**Tecnológico  
de Monterrey**

## Implementación de Métodos Computacionales

Gamaliel Abisay Palomo Briones

Tarea 5.2. Reto Prolog

Santos Alejandro Arellano Olarte // A01643742

Arturo Ramos Martínez // A01643269

Jorge Javier Blasquez Gonzalez // A01637706

Fernando Antonio López García // A01643685

Carlos Ivan Armenta Naranjo // A016453070

20/Mayo/2024

## 1. ¿Qué Respuestas Se Obtienen De Los Siguietes Programas Y Consultas En Prolog?

Programa :

```
Vegetarian(Jose).  
Vegetarian(James).  
Vegetable(Carrot).  
Vegetable(Egg_plant).  
Likes(Jose, X) :- Vegetable(X).  
Loves(Who, Egg_plant) :- Vegetarian(Who).
```

Consultas :

```
?- Vegetable(X). : X = Carrot  
?- Vegetable(Potato). : Potato = carrot  
?- Vegetarian(_). : True  
?- Likes(Jose, What). : False  
?- Likes(Who, Egg_plant). : False  
?- Loves(Who, Egg_plant). : Who = james.
```

## 2. Diseña Un Programa Que Contenga Predicados Y Consultas Para Los Siguietes Enunciados:

- A. María Lee El Libro Harry Potter Cuyo Autor Es J.K. Rowling
- B. A Una Persona Le Gusta Ir De Shopping Si Es Una Mujer.
- C. Jorge Detesta Cualquier Ciudad Que Sea Grande Y Con Mucha Gente.
- D. Si Una Persona Es Vegetariana, Entonces Le Gustan Las Zanahorias.

```
% Hechos  
actividad(leer).  
actividad(shopping).  
  
autor(harry_potter, j_k_rowling).  
autor(leer, murakami).  
  
persona(maria, mujer).  
persona(ana, mujer).  
persona(jorge, hombre).
```

```

ciudad(nueva_york, grande).
ciudad(nueva_york, con_mucha_gente).
ciudad(tokio, grande).
ciudad(tokio, con_mucha_gente).
ciudad(colima, pequeño).
ciudad(colima, con_poca_gente).

vegetariano(ana).
vegetariano(jorge).

lee(maria, harry_potter).

% Reglas
le_gusta_ir_de_shopping(Persona) :- persona(Persona, mujer).

detesta(jorge, Ciudad) :-
    ciudad(Ciudad, grande),
    ciudad(Ciudad, con_mucha_gente).

le_gustan(Persona, zanahorias) :- vegetariano(Persona).

lee_libro_con_autor(Persona, Libro, Autor) :-
    lee(Persona, Libro),
    autor(Libro, Autor).

```

```

28 ?- le_gustan(ana, zanahorias).
true.

29 ?- le_gustan(jorge, zanahorias).
true.

30 ?- le_gustan(maria, zanahorias).
false.

```

```

26 ?- detesta(jorge, Ciudad).
Ciudad = nueva_york .

27 ?-
detesta(jorge, colima).
false.

```

```
22 ?- le_gusta_ir_de_shopping(maria).  
true.
```

```
23 ?- le_gusta_ir_de_shopping(Persona).  
Persona = maria .
```

```
24 ?- detesta(jorge, nueva_york).  
true .
```

```
20 ?- le_gusta_ir_de_shopping(jorge).  
false.
```

```
17 ?- lee_libro_con_autor(maria, harry_potter, j_k_rowling).  
true.
```

```
18 ?- lee_libro_con_autor(maria, harry_potter, murakami).  
false.
```

```
19 ?- lee_libro_con_autor(ana, harry_potter, j_k_rowling).  
false.
```

3. Encontrar Los Predicados Que Ayuden A Resolver Los Sigüentes Ejercicios, Además De Implementarlos En Un Archivo \*.PI:

- **Encontrar El Último Elemento De Una Lista.**

- Ejemplo:  
?- My\_last(X,[A,B,C,D]).  
X = D

**my\_last(X, [X]).**

**my\_last(X, [\_|Tail]) :- my\_last(X, Tail).**

**Uso:**

**?- my\_last(X, [a, b, c, d]).**

**X = d.**

- **Encontrar El K-Ésimo Elemento De Una Lista.**

- Ejemplo:  
?- Element\_at(X,[A,B,C,D,E],3).  
X = C

```
element_at(X, [X|_], 1).  
element_at(X, [_|Tail], K) :-  
    K > 1,  
    K1 is K - 1,  
    element_at(X, Tail, K1).
```

Uso: ?- element\_at(X, [a, b, c, d, e], 3).  
X = c.