

Guía de Laboratorio - Programación Declarativa

Curso de Programación Lógica y Funcional

Ejercicios en Haskell

1. Determinar si un número es par

```
esPar :: Int -> Bool
esPar n = n `mod` 2 == 0

-- Ejemplo:
-- esPar 4  => True
-- esPar 5  => False
```

2. Calcular el cuadrado de un número

```
cuadrado :: Int -> Int
cuadrado x = x * x

-- Ejemplo:
-- cuadrado 5 => 25
```

3. Obtener el factorial de un número

```
factorial :: Int -> Int
factorial 0 = 0
factorial 1 = 1
factorial n = n * factorial (n - 1)

-- Ejemplo:
-- factorial 5 => 120
```

4. Sumar todos los elementos de una lista

```
sumaLista :: [Int] -> Int
sumaLista [] = 0
sumaLista (x:xs) = x + sumaLista xs

-- Ejemplo:
-- sumaLista [1,2,3,4] => 10
```

5. Obtener el máximo de una lista

```
maximo :: [Int] -> Int
maximo [x] = x
maximo (x:xs) = max x (maximo xs)

-- Ejemplo:
-- maximo [3,8,2,5] => 8
```

Ejercicios en Prolog

1. Definir hechos de animales y clasificarlos como mamíferos o aves

```
mamifero(perro).
mamifero(gato).
ave(gallina).
ave(paloma).

es_mamifero(X) :- mamifero(X).
es_ave(X) :- ave(X).

% Consultas:
% ?- es_mamifero(perro).
% ?- es_ave(paloma).
```

2. Regla que determine si un número es par o impar

```
par(N) :- 0 is N mod 2.
impar(N) :- 1 is N mod 2.

% Consultas:
% ?- par(4).
% ?- impar(7).
```

3. Reglas para identificar si alguien es hijo de otra persona

```
padre(juan, maria).
madre(ana, maria).

hijo(H, P) :- padre(P, H).
hijo(H, M) :- madre(M, H).

% Consultas:
% ?- hijo(maria, juan).
% ?- hijo(maria, ana).
```

4. Modelar una familia pequeña con relaciones de padre, madre e hijo

```
padre(carlos, pedro).
madre(luisa, pedro).
padre(carlos, ana).
madre(luisa, ana).

hijo(H, P) :- padre(P, H).
```

```
hijo(H, M) :- madre(M, H).
```

```
% Consultas:
```

```
% ?- hijo(pedro, carlos).
```

```
% ?- hijo(ana, luisa).
```

5. Representar una lista de números y consultar el primero y el último elemento

```
lista([1,2,3,4,5]).
```

```
primero([H|_], H).
```

```
ultimo([X], X).
```

```
ultimo([_|T], X) :- ultimo(T, X).
```

```
% Consultas:
```

```
% ?- lista(L), primero(L, P).
```

```
% ?- lista(L), ultimo(L, U).
```