

### Ejercicio 1

Se tiene el predicado `dobleLength/2` que debería relacionar una lista con el doble de su longitud y está definido de la siguiente manera:

```
dobleLength(L,N):-
    length(L,N),
    N is N * 2.
/* el predicado length relaciona una lista con su longitud */
```

1. Al hacer la consulta `dobleLength([a,b,c],X)` la respuesta obtenida es "No". ¿A qué se debe esta respuesta? Señale el error en el predicado.
2. Corrija el programa para que funcione adecuadamente. Explique los cambios que realizó.
3. ¿El predicado es nada, parcial o totalmente inversible? Dé todos los ejemplos de consultas necesarios para poder demostrarlo.

### Ejercicio 2

Se necesita:

- hacer la función que me dice si una lista de fechas es creciente.
- hacer la función que me dice si una lista de personas está ordenada alfabéticamente por nombre.

Sabiendo que existe la función `esMenor` que dadas dos fechas indica si la primera es menor a la segunda y la función `esAnterior` que, dados dos strings, indica si el primero es anterior al segundo considerando el orden alfabético.

Dos programadores lo resolvieron de la siguiente forma:

#### Programador 1

```
sonFechasCrecientes [_] = True
sonFechasCrecientes (x:y:xs) = esMenor x y && sonFechasCrecientes (y:xs)

estanOrdenadosAlfabeticamente [_] = True
estanOrdenadosAlfabeticamente (x:y:xs) =
    esAnterior x y && estanOrdenadosAlfabeticamente (y:xs)
```

#### Programador 2

```
cumple f [_] = True
cumple f (x:y:xs) = f x y && cumple f (y:xs)

sonFechasCrecientes fechas = cumple esMenor fechas
estanOrdenadosAlfabeticamente personas = cumple esAnterior personas
```

a) ¿Ambas soluciones funcionan correctamente? En caso negativo, hacer los arreglos necesarios en cada una de ellas.

b) Indicar la presencia o ausencia de los siguientes conceptos en cada solución (en caso de presencia, indicar en qué parte de la solución se ven los conceptos).

1. Aplicación parcial
2. Orden superior
3. Pattern matching

c) Definir el tipo de la función `cumple`

### Ejercicio 3

Se tiene el siguiente código Smalltalk:

```
Persona (vi: rutinaDeEjercicios)
    totalDeCaloriasQuemadasAlHacerRutina
        ^(rutinaDeEjercicios collect:
```

```

[:ejercicio | ejercicio caloríasQuemadasPor: self]) sum

EntrenamientoEnBicicleta (vi: minutosTotales, velocidad)
    caloríasQuemadasPor: unaPersona
        ^velocidad / 800 * minutosTotales * unaPersona pesoEnKg

CaminataEnCinta (vi: minutosTotales, kmRecorridos)
    caloríasQuemadasPor: unaPersona
        ^kmRecorridos * (1000/minutosTotales) * (80 / unaPersona pesoEnKg)

```

1. Se requiere agregar otros ejercicios, que son similares a las caminatas en cinta pero con pendiente, lo que hace que la persona queme calorías como en la caminata en cinta normal, más la décima parte de la pendiente (se supone un valor medido en grados, que indica el ángulo de inclinación).
  - a. Incorporar este agregado codificando la solución.
  - b. ¿Qué concepto que antes no estaba presente decidió utilizar en su solución? ¿Por qué?
2. Otro ejercicio que se desea incorporar es una sesión de pilates, que quema siempre 100 calorías sin importar qué persona la realice. ¿Qué concepto ya estaba presente en la solución original que resultaría útil para incorporar este agregado? ¿Por qué?
3. Completar (y corregir, si es necesario) la siguiente solución en Prolog que es similar a la desarrollada en el punto 1.

```

%persona(Persona, Peso, Ejercicios).
totalDeCaloriasQuemadasAlHacerRutina(Persona, CaloriasQuemadas):-
    persona(Persona, _, Ejercicios),
    findall(Caloria, (member(Ejercicio,Ejercicios),
        caloríasQuemadasPor(Ejercicio,Persona,Caloria), Calorias),
        sumlist(Calorias, CaloriasQuemadas)).

```

4. Mencione al menos dos conceptos que aparecen en la solución completa (es decir, considerando tanto el código dado en el enunciado como lo agregado o corregido) y justifique cuáles fueron las ventajas de haberlos aplicado.
5. Si se necesita pedirle a la persona que efectivamente realice su rutina de ejercicios, y por lo tanto disminuya su peso en función de las calorías quemadas (si después le pedimos su peso a la persona vamos a ver que disminuyó), ¿se puede hacer tanto en la solución hecha en Smalltalk como en la hecha en Prolog? ¿Qué concepto(s) son necesarios para poder incorporar este cambio? Justifique.