

1. a) Derivar una definición recursiva para la función especificada por:

$$f.xs = \langle \forall as, bs : xs = as ++ bs : prod.as > sum.bs \rangle$$

- b) Dar una lista de 3 elementos xs que cumpla $f.xs = True$. Justificar.
 c) Dar otra lista de 3 elementos xs que cumpla $f.xs = False$. Justificar.
2. Considere el problema de, dado un arreglo de **números enteros**, calcular la cantidad de pares de elementos cuyo producto da 1, especificado de la siguiente manera:

Const $N : Int, A : array[0, N)$ of Int ;

Var $r : Int$;

{ $P : N \geq 0$ }

S

{ $Q : r = \langle N p, q : 0 \leq p < q < N : A.p * A.q = 1 \rangle$ }

- a) Calcular el resultado para $A = [-1, 2, 1, -1, 1]$. Justificar, enumerando todos los elementos del rango.
 b) Derivar un programa imperativo que resuelva este problema. El programa **debe recorrer una sola vez el arreglo** (sin ciclos anidados).
Ayuda: $a * b = 1$ sí y sólo sí $(a = 1 \wedge b = 1) \vee (a = -1 \wedge b = -1)$.

3. **Especificar con pre y post condición** los siguientes problemas. Declarar constantes y variables. **No derivar.**

- a) Dados dos arreglos A y B , ambos de $N \geq 0$ números, calcular la suma de los elementos de B que son mayores a todos los de A .
 b) Dado un arreglo A de $N \geq 0$ números, calcular si se cumple que en todas las posiciones pares hay elementos pares y en todas las posiciones impares hay elementos impares.

4. **(Ejercicio para libres:)** Derivar un programa imperativo que satisfaga la siguiente especificación.

Const $N : Int, A : array[0, N)$ of Int ;

Var $r : Bool$;

{ $P : N \geq 0$ }

S

{ $Q : r = \langle \forall i : 0 \leq i < N : A.i = i! \rangle$ }