1. Expresa los siguientes enunciados en **LPO**:

   a) Hay tantos números naturales pares como impares menores que *i*.

   b) El producto de los números mayores que *i* y menores que *j* tiene *z* divisores.

2. Decide cuáles de las siguientes afirmaciones son ciertas y razona aquellas que no los sean:

   a) $1 \leq i \leq n \wedge A[i] \neq 0 \rightarrow \mathrm{def}(A[i]==0)$

   b) $x \, mod \, i = 0 \rightarrow \exists i (1 < i < x \wedge x \, mod \, i = 0)$

   c) $\mathrm{num\_ceros} = \aleph i (1 \leq i < k \wedge A[i] = 0) \wedge A[k] = 0$
   $\rightarrow \mathrm{num\_ceros} = \aleph i (1 \leq i < k \wedge A[i] = 0)$

   d) $\mathrm{def}(A[i] = 0) \rightarrow A[i] = 0$

   e) $x = a/b \wedge b \neq 0 \rightarrow x - 1 = (a - b)/b$

3. Descompón la siguiente representación esquemática de un programa:

$/* \Phi */$

$while(B_1)$
$\{$
$if\,(B_2)\,A_1;$
$else\; A_2;$

$A_3;$
$\}$

$/* \Psi */$

4. Deduce **el invariante y la expresión cota E** del siguiente bucle:

$/* \, i = n \wedge \text{num\_pares} = 0 \wedge \text{num\_impares} = 0 \, */$
$while\,(i > 0)$
$\{$
$if\,(A[i] \ \% \ 2 == 0)$
$\quad \text{num\_pares} = \text{num\_pares} + 1\,;$
$else$
$\quad \text{num\_impares} = \text{num\_impares} + 1\,;$

$i = i - 1\,;$
$\}$
$/* \, \text{num\_pares} = \aleph\, i\,(1 \le i \le n \wedge A[i]\, mod\, 2 = 0) \wedge$
$\text{num\_impares} = \aleph\, i\,(1 \le i \le n \wedge A[i]\, mod\, 2 \ne 0) \, */$

5. **Verifica la corrección parcial** del siguiente programa:

$/* \, i = 0 \wedge x > 0 \wedge s = 0 \, */$
$while\,(i < x)$
$\{$
$i = i + 1\,;$
$s = s + A[i]\,;$
$s = s + 1\,;$
$\}$
$/* \, s = x + \sum_{j=1}^{x} A[j] \, */$

$$/* \, INV \, */ \equiv /* \, 0 \le i \le x \wedge s = i + \sum_{j=1}^{i} A[j] \, */$$

6. **Verifica la corrección parcial** del siguiente programa:

function sqrt(int x) return int res
$PRE \equiv$ /* x>=0 */
$POST \equiv$ /* res= $\sqrt{x}$ */

$/* \, z = a \, */$
$if\,(z < 0)$
$\quad z = -z\,;$

$y = \text{sqrt}(z)\,;$
$/* \, y = \sqrt{|a|} \, */$

7. **Deriva** una función recursiva que, dado un vector de números enteros *A[1..n]* y unos índices *i* y *j*, nos devuelva el producto de los elementos entre las posiciones *i* y *j*.