**43. (Abril 2010 #1) Predicados par(x), impares(D(1..p)) y parpar(E(1..r), ($e_1, e_2, \ldots, e_r$), F(1..r), ($f_1, f_2, \ldots, f_r$), G(1..r), ($g_1, g_2, \ldots, g_r$), pos) y programa que, dados tres vectores de enteros A(1..n), B(1..n) y C(1..n) y sabiendo que n ≥ 1 y que al principio todos los elementos de C(1..n) son impares, intercambia los elementos de B(1..n) y C(1..n) que ocupan la misma posición siempre que los elementos de esa posición en A(1..n) y B(1..n) sean números pares. -- #**

**a)** **par(x)** $\equiv \{x \bmod 2 = 0\}$

**b)** **impares(D(1..r))** $\equiv \forall k\ (1 \le k \le r \to \neg par(D(k)))$

**c)** **parpar(E(1..r), ($e_1, e_2, \ldots, e_r$), F(1..r), ($f_1, f_2, \ldots, f_r$), G(1..r), ($g_1, g_2, \ldots, g_r$), pos)** $\equiv$

$\{(0 \le pos \le r) \wedge$
$\forall k\ ((1 \le k \le pos \wedge par(E(k))) \to \neg par(F(k))) \wedge$
$\forall k\ ((1 \le k \le pos \wedge par(e_k) \wedge par(f_k)) \to (E(k) = e_k \wedge F(k) = g_k \wedge G(k) = f_k)) \wedge$
$\forall k\ ((1 \le k \le pos \wedge (\neg par(e_k) \vee \neg par(f_k))) \to (E(k) = e_k \wedge F(k) = f_k \wedge G(k) = g_k))\}$

**d)** Las aserciones se darán en el orden en el que es más natural formularlas:

(1) {Precondición} $\equiv \{n \ge 1 \wedge$
$\forall k\ (1 \le k \le n \to (A(k) = a_k \wedge B(k) = b_k \wedge C(k) = c_k)) \wedge impares(C(1..n))\}$

En la precondición se indica que los vectores A, B y C tendrán por lo menos un elemento, que los valores iniciales de A, B y C los representaremos mediante *a*'s, b's y c's minúsculas con los correspondientes subíndices y que todos los elementos de C son impares.

(2) {Aserción intermedia} $\equiv \{(1) \wedge i = 0\}$

(9) {Postcondición} $\equiv$
{parpar(A(1..n), ($a_1, a_2, \ldots, a_n$), B(1..n), ($b_1, b_2, \ldots, b_n$), C(1..n), ($c_1, c_2, \ldots, c_n$), n)}

En la postcondición se indica que se han hecho todos los cambios hasta la posición n incluida, es decir, se ha recorrido todo el vector y se han hecho todos los cambios necesarios.

(3) {Invariante} $\equiv$
$\{(0 \le i \le n) \wedge$ parpar(A(1..n), ($a_1, a_2, \ldots, a_n$), B(1..n), ($b_1, b_2, \ldots, b_n$), C(1..n), ($c_1, c_2, \ldots, c_n$), i)}

En el invariante se indica que se han hecho todos los cambios hasta la posición i incluida, es decir, estamos situados en la posición i y ya se ha analizado lo que ocurre en la posición i.

(4) {Aserción intermedia} ≡
$\{(0 \le i \le n - 1) \wedge \text{parpar}(A(1..n), (a_1, a_2, \ldots, a_n), B(1..n), (b_1, b_2, \ldots, b_n), C(1..n), (c_1, c_2, \ldots, c_n), i)\}$

Por haber entrado en el while sabemos que se ha cumplido la condición del while y que i no es n.

(5) {Aserción intermedia} ≡
$\{(0 \le i \le n - 1) \wedge \text{parpar}(A(1..n), (a_1, a_2, \ldots, a_n), B(1..n), (b_1, b_2, \ldots, b_n), C(1..n), (c_1, c_2, \ldots, c_n), i) \wedge \text{par}(A(i + 1)) \wedge \text{par}(B(i + 1)) \wedge A(i + 1) = a_{i+1} \wedge B(i + 1) = b_{i+1}\}$

Por haber entrado en la rama then del if sabemos que los elementos A(i + 1) y B(i + 1) son pares. Además podemos asegurar que en A(i + 1) y B(i + 1) se tienen los valores iniciales $a_{i+1}$ y $b_{i+1}$.

(6) {Aserción intermedia} ≡
$\{(0 \le i \le n - 1) \wedge \text{parpar}(A(1..n), (a_1, a_2, \ldots, a_n), B(1..n), (b_1, b_2, \ldots, b_n), C(1..n), (c_1, c_2, \ldots, c_n), i) \wedge \text{par}(A(i + 1)) \wedge \text{par}(B(i + 1)) \wedge A(i + 1) = a_{i+1} \wedge B(i + 1) = b_{i+1} \wedge \text{aux} = b_{i+1}\}$

Tras ejecutarse la asignación aux : = B(i + 1); sabemos que ahora en aux tenemos $b_{i+1}$. Pero de momento los cambios completos están hechos hasta la posición i, en la posición i + 1 estamos a medias por lo que en el predicado parpar seguimos poniendo i.

(7) {Aserción intermedia} ≡
$\{(0 \le i \le n - 1) \wedge \text{parpar}(A(1..n), (a_1, a_2, \ldots, a_n), B(1..n), (b_1, b_2, \ldots, b_n), C(1..n), (c_1, c_2, \ldots, c_n), i) \wedge \text{par}(A(i + 1)) \wedge \text{par}(b_{i+1}) \wedge A(i + 1) = a_{i+1} \wedge B(i + 1) = C(i + 1) = c_{i+1} \wedge \text{aux} = b_{i+1}\}$

Tras ejecutarse la asignación B(i + 1) : = C(i + 1); sabemos que ahora en B(i + 1) tenemos lo mismo que en C(i + 1) que es el valor inicial $c_{i+1}$. Ahora lo que es par es $b_{i+1}$ pero no B(i + 1). Todavía los cambios completos están hechos hasta la posición i, en la posición i + 1 estamos a medias por lo que en el predicado parpar seguimos poniendo i.

(8) {Aserción intermedia} ≡
$\{(0 \leq i \leq n - 1) \wedge \text{parpar}(A(1..n), (a_1, a_2, \ldots, a_n), B(1..n), (b_1, b_2, \ldots, b_n),$
$C(1..n), (c_1, c_2, \ldots, c_n), i + 1)\}$

La diferencia entre (7) y (8) es que en (7) sabemos que se ha ido por la rama **then** y por tanto podemos asegurar que $\text{par}(A(i + 1)) \wedge \text{par}(b_{i+1}) \wedge A(i + 1) = a_{i+1} \wedge B(i + 1) = C(i + 1) = c_{i+1} \wedge aux = b_{i+1}$, pero en (8) no sabemos si se ha ido por la rama then o si no se ha cumplido la condición del if y por ello no podemos asegurar que se cumpla $\text{par}(A(i + 1)) \wedge \text{par}(b_{i+1}) \wedge A(i + 1) = a_{i+1} \wedge B(i + 1) = C(i + 1) = c_{i+1} \wedge aux = b_{i+1}$ ya que si no se ha cumplido la condición del if, $\text{par}(A(i + 1)) \wedge \text{par}(b_{i+1}) \wedge B(i + 1) = C(i + 1) = c_{i+1} \wedge aux = b_{i+1}$ no será cierto. Pero ahora ya están hechos todos los cambios correspondientes a la posición i + 1 y por ello en el predicado parpar ponemos i + 1.

(10) $E = n - i$

La expresión cota nos indica, siempre que estemos en el punto donde se cumple el invariante, cuántas vueltas quedan por dar. Es "el último valor que tomará i" menos "i". La expresión E es al fin y al cabo la distancia entre n y la variable i. Cuando i crece, la distancia disminuye y el número de vueltas pendientes decrece.

Las partes coloreadas resaltan los cambios que hay de una aserción a otra.