44. (2010eko apirila #2) bitak(E(1..q)) eta biratuta(F(1..r), (f_1, f_2, …, f_r), G(1..r), (g_1, g_2, …, g_r), H(1..r), (h_1, h_2, …, h_r), Q(1..r), pos) predikatuak eta A(1..n), B(1..n), C(1..n) eta zeroak eta batekoak bakarrik dituen D(1..n) bektoreak emanda, C(i) 0 bada A(i) → B(i) → C(i) → A(i) biraketa eta C(i) 1 bada A(i) → C(i) → B(i) → A(i) biraketa burutzen duen programa. -- #

a) **bitak(E(1..r))** $\equiv \forall k\, (1 \le k \le r \rightarrow E(k) = 0 \vee E(k) = 1)$

b) **biratuta(F(1..r), (f_1, f_2, …, f_r), G(1..r), (g_1, g_2, …, g_r), H(1..r), (h_1, h_2, …, h_r), Q(1..r), pos)** $\equiv$

$\{(0 \le pos \le r) \wedge$
$bitak(Q(1..r)) \wedge$
$\forall k\, ((1 \le k \le pos \wedge Q(k) = 0) \rightarrow (F(k) = h_k \wedge G(k) = f_k \wedge H(k) = g_k)) \wedge$
$\forall k\, ((1 \le k \le pos \wedge Q(k) = 1) \rightarrow (F(k) = g_k \wedge G(k) = h_k \wedge H(k) = f_k))\}$

b) atala egiteko beste aukera bat:

biratuta(F(1..r), (f_1, f_2, …, f_r), G(1..r), (g_1, g_2, …, g_r), H(1..r), (h_1, h_2, …, h_r), Q(1..r), pos) $\equiv$

$\{(0 \le pos \le r) \wedge$
$bitak(Q(1..r)) \wedge$
$\forall k\, (1 \le k \le pos \rightarrow ((Q(k) = 0 \rightarrow (F(k) = h_k \wedge G(k) = f_k \wedge H(k) = g_k)) \wedge$
$(Q(k) = 1 \rightarrow (F(k) = g_k \wedge G(k) = h_k \wedge H(k) = f_k))))\}$

c) Asertzioak ematerakoan egokiena edo naturalena den ordena jarraituko da eta ez zenbakizko ordena:

(1) {Hasierako baldintza} $\equiv \{n \ge 1 \wedge$
$\forall k\, (1 \le k \le n \rightarrow (A(k) = a_k \wedge B(k) = b_k \wedge C(k) = c_k)) \wedge bitak(D(1..n))\}$

Hasierako baldintzak honako hau dio: A, B, C eta D bektoreek gutxienez elementu bat izango dute, A, B eta C tauletako hasierako balioak *a*, b eta c minuskulen bidez eta dagozkien azpiindizeak erabiliz adieraziko dira eta gainera D bektoreko posizio bakoitzean 0 edo 1 dago.

(2) {Tarteko asertzioa} $\equiv \{(1) \wedge i = 0\}$

(10) {Bukaerako baldintza} $\equiv$
{biratuta(A(1..n), (a_1, a_2, …, a_n), B(1..n), (b_1, b_2, …, b_n), C(1..n), (c_1, c_2, …, c_n), D(1..n), n)}

Bukaerako baldintzaren bidez tauletan egin beharreko aldaketa denak egin direla adierazten da, hau da, taulak n posizioraino zeharkatu dira eta egin beharreko biraketak burutu dira.

(3) {Inbariantea} ≡
$\{(0 \le i \le n) \wedge \text{biratuta}(A(1..n), (a_1, a_2, \ldots, a_n), B(1..n), (b_1, b_2, \ldots, b_n), C(1..n), (c_1, c_2, \ldots, c_n), D(1..n), i)\}$

Inbariantearen bidez i posizioraino egin beharreko aldaketa denak eginda daudela adierazten da.

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

While-an sartu garenez badakigu i ez dela n-ren berdina izango.

(5) {Tarteko asertzioa} ≡
$\{(1 \le i \le n) \wedge \text{biratuta}(A(1..n), (a_1, a_2, \ldots, a_n), B(1..n), (b_1, b_2, \ldots, b_n), C(1..n), (c_1, c_2, \ldots, c_n), D(1..n), i - 1)\}$

i aldagaiaren balioa handitu egin da eta ondorioz i aldagaiaren eremuaren goiko eta beheko muga ere handitu egin dira. Orain aldaketak i – 1 posizioraino daude eginda eta ez i posizioraino.

(6) {Tarteko asertzioa} ≡
$\{(1 \le i \le n) \wedge \text{biratuta}(A(1..n), (a_1, a_2, \ldots, a_n), B(1..n), (b_1, b_2, \ldots, b_n), C(1..n), (c_1, c_2, \ldots, c_n), D(1..n), i - 1) \wedge D(i) = 0\}$

If aginduko *then* bidetik sartu garenez, D(i) posizioan 0 balioa daukagula badakigu.

(7) {Tarteko asertzioa} ≡
$\{(1 \le i \le n) \wedge \text{biratuta}(A(1..n), (a_1, a_2, \ldots, a_n), B(1..n), (b_1, b_2, \ldots, b_n), C(1..n), (c_1, c_2, \ldots, c_n), D(1..n), i - 1) \wedge D(i) = 0 \wedge lag = C(i) = c_i\}$

Tras ejecutarse la asignación lag : = C(i); esleipena burutu ondoren, lag aldagaian eta C(i) posizioan balio bera daukagu eta balio hori gainera posizio horretako hasierako balioa da, c_i. Une honetan aldaketak i – 1 posizioraino daude eginda, i posizioan erdizka gaude, aldaketak egiten hasita baina bukatzeke eta horregatik *biratu* predikatuan i – 1 ipini behar da.

(8) {Tarteko asertzioa} ≡

$\{(1 \le i \le n) \wedge$ biratuta$(A(1..n), (a_1, a_2, \ldots, a_n), B(1..n), (b_1, b_2, \ldots, b_n), C(1..n), (c_1, c_2, \ldots, c_n), D(1..n), i - 1) \wedge D(i) = 0 \wedge lag = c_i \wedge C(i) = B(i) = b_i\}$

C(i) : = B(i); esleipena burutu ondoren C(i) posizioan eta B(i) posizioan balio bera dugu eta gaitera balio hori B(i) posizioko hasierako balioa da, b_i. Orain lag aldagaian ez daukagu C(i) posizioan dagoen balio bera, C(i) posizioko hasierako balioa dago, hau da, c_i. Une honetan aldaketak i – 1 posizioraino daude eginda, i posizioan erdizka gaude, aldaketak egiten ari gara baina bukatzeke dago eta horregatik *biratu* predikatuan i – 1 ipini behar da.

(9) {Tarteko asertzioa} ≡

$\{(1 \le i \le n) \wedge$ biratuta $(A(1..n), (a_1, a_2, \ldots, a_n), B(1..n), (b_1, b_2, \ldots, b_n), C(1..n), (c_1, c_2, \ldots, c_n), D(1..n), i - 1) \wedge D(i) = 0 \wedge lag = c_i \wedge C(i) = b_i \wedge B(i) = A(i) = a_i\}$

B(i) : = A(i); esleipena burutu ondoren B(i) posizioan eta A(i) posizioan balio bera daukagu, A(i) posizioko hasierako balioa, a_i. Orain lag aldagaian ez daukagu C(i) posizioan dagoen balio bera, C(i) posizioko hasierako balioa dago, hau da, c_i. Bestalde C(i) posizioan ez dago B(i) posizioko balio bera, B(i) posizioko hasierako balioa dago, hau da, b_i Une honetan aldaketak i – 1 posizioraino daude eginda, i posizioan erdizka gaude, aldaketak egiten ari gara baina bukatzeke dago eta horregatik *biratu* predikatuan i – 1 ipini behar da.

(11) E = n – i

Inbariantea betetzen den puntuan gauden bakoitzean E espresioak while-a bukatzeko zenbat buelta falta diren adierazten digu. Beraz, "i aldagaiak hartuko duen azkeneko balioa" ken "i" izango da. Azken batean E espresioak n eta i-ren arteko distantzia adierazten du. Horrela, i-ren balioa handitzen denean distantzia txikitu egiten da eta gelditzen den buelta-kopurua ere txikiagoa izango da.

Asertzio batetik bestera zer aldatzen den hobeto ikusteko, aldaketak kolorez ipini dira.