45. (2010eko ekaina) bikoitia(x) eta mugibi(C(1..r), (c_1, c_2, …, c_r), pos) predikatuak eta A(1..n) bektorean jarraian dauden posizioak (1 eta 2, 3 eta 4, 5 eta 6, eta abar) binaka trukatzen dituen programa. A(1..n) bektoreko elementu kopurua bakoitia bada (n bakoitia), azkeneko elementua ez da lekuz mugituko. -- #

a) bikoitia(x) ≡ {x mod 2 = 0}

b) **mugibi(C(1..r), (c_1, c_2, …, c_r), pos)** ≡

$\{(0 \le pos \le r) \wedge$
$\forall k\ (1 \le k \le pos \rightarrow ((bikoitia(k) \rightarrow (C(k) = c_{k-1} \wedge C(k-1) = c_k)) \wedge$
$\wedge\ ((k = pos \wedge \neg bikoitia(k)) \rightarrow C(k) = c_k)))\}$

b) atalerako beste aukera bat:

mugibi(C(1..r), (c_1, c_2, …, c_r), pos) ≡

$\{(0 \le pos \le r) \wedge$
$\forall k\ ((1 \le k \le pos \wedge bikoitia(k)) \rightarrow (C(k) = c_{k-1} \wedge C(k-1) = c_k)) \wedge$
$(\neg bikoitia(pos) \rightarrow C(k) = c_k)\}$

c) Asertzioak ematerakoan egokiena den ordena jarraituko da:

(1) {Hasierako baldintza} ≡ $\{n \ge 1 \wedge \forall k\ (1 \le k \le n \rightarrow A(k) = a_k)\}$

Hasierako baldintzaren bidez A bektoreak gutxienez elementu bat izango duela eta A bektoreko hasierako balioak *a* minuskulen bidez eta dagozkien azpiindezeak erabiliz adieraziko ditugula esaten da.

(2) {Tarteko asertzioa} ≡ {(1) ∧ i = 1}

(9) {Bukaerako baldintza} ≡
{mugibi(A(1..n), (a_1, a_2, …, a_n), n)}

Bukaerako baldintzaren bidez bektore osoan, hau da, n posizioraino egin beharreko mugimendu denak eginda daudela esaten da.

(3) {Inbariantea} ≡ {(1 ≤ i ≤ n + 1) ∧ mugibi(A(1..n), (a_1, a_2, …, a_n), i – 1)}

Inbariantearen bidez i – 1 posizioraino ahal izan diren aldaketak egin direla adierazten da. Beraz *mugibi* predikatuaren definizioa kontuan hartuz, i – 1 bikoitia baldin bada, 1 eta 2, 3 eta 4, eta gainerako posizioetako balioak binaka lekuz trukatu dira i – 2 eta i – 1 posizioetaraino. Baina i – 1 bakoitia baldin bada, 1 eta 2, 3 eta 4, eta gainerako posizioetako balioak binaka lekuz trukatu dira i – 3 eta i – 2 posizioetaraino eta i – 1 posizioko elementua ez da mugitu.

(4) {Tarteko asertzioa} ≡ $\{(1 \le i \le n) \wedge$
$\text{mugibi}(A(1..n), (a_1, a_2, \ldots, a_n), i - 1)\}$

while-ean sartu garenez badakigu while-aren baldintza bete egin dela eta i ez dela n + 1.

(5) {Tarteko asertzioa} ≡ $\{(1 \le i \le n) \wedge \text{bikoitia}(i) \wedge$
$\text{mugibi}(A(1..n), (a_1, a_2, \ldots, a_n), i - 1)\}$

if aginduaren **then** aukeratik sartu garenez, badakigu i bikoitia dela.

(6) {Tarteko asertzioa} ≡ $\{(1 \le i \le n) \wedge \text{bikoitia}(i) \wedge lag = A(i - 1) = a_{i-1} \wedge$
$\text{mugibi}(A(1..n), (a_1, a_2, \ldots, a_n), i - 1)\}$

lag : = A(i – 1); esleipena burutu ondoren *lag* aldagaiaren balioa eta A(i – 1) berdinak izango dira eta gainera balio hori a_{i-1} hasierako balioa izango da. Baina *mugibi* predikatua i – 1 balioarentzat betetzen da oraindik, izan ere, i posizioari dagokionez erdizka gaude.

(7) {Tarteko asertzioa} ≡
$\{(1 \le i \le n) \wedge \text{bikoitia}(i) \wedge lag = a_{i-1} \wedge A(i - 1) = A(i) = a_i$
$\text{mugibi}(A(1..n), (a_1, a_2, \ldots, a_n), i - 1)\}$

A(i – 1) : = A(i); burutu ondoren lag aldagaiaren balioa eta A(i – 1) ez dira berdinak izango baina lag aldagaian a_{i-1} hasierako balioa mantenduko da. Bestalde A(i – 1) eta A(i) berdinak izango dira orain eta beraien balioa a_i hasierako balioa izango da. Oraindik ere *mugibi* predikatua i – 1 balioarentzat betetzen da, i posizioari dagokionez erdizka baikaude.

(8) {Tarteko asertzioa} ≡ $\{(1 \le i \le n) \wedge \text{bikoitia}(i) \wedge lag = a_{i-1} = A(i) \wedge$
$A(i - 1) = a_i \wedge$
$\text{mugibi}(A(1..n), (a_1, a_2, \ldots, a_n), i)\}$

A(i) : = lag; esleipena burutu ondoren lag aldagaiaren balioa eta A(i)-rena berdinak izango dira, a_{i-1} hasierako balioa. Orain A(i – 1) eta A(i) desberdinak dira: A(i – 1)-ren balioa a_i da eta A(i)-rena a_{i-1}. Trukateta bukatu denez *mugibi* predikatua i balioarekiko beteko da orain.

(9) {Tarteko asertzioa} ≡ $\{(1 \le i \le n) \wedge \text{mugibi}(A(1..n), (a_1, a_2, \ldots, a_n), i)\}$

(8) eta (9) puntuen arteko desberdintasuna honako hau da: (8) puntuan gaudenean badakigu if aginduaren then aukeratik joan garela eta horregatik $\text{bikoitia}(i) \wedge lag = a_{i-1} = A(i) \wedge A(i - 1) = a_i$ betetzen dela ziurtatu dezakegu. Baina (9) puntuan gaudenean ez dakigu then aukeratik joan al garen ala ez eta horregatik ezin dugu esan $\text{bikoitia}(i) \wedge lag = a_{i-1} = A(i) \wedge A(i - 1) = a_i$ betetzen denik, if ainduko baldintza ez bada bete $\text{bikoitia}(i) \wedge lag = a_{i-1} = A(i) \wedge A(i - 1) = a_i$ ez baita egia izango.

(11) $E = n + 1 – i$

Inbariantea betetzen den lekuan gauden bakoitzean E espresioak while agindua bukatzeko zenbat buelta falta diren adierazi behar du. Taula ezkerretik eskuinera zeharkatzen denean E espresioa "i aldagaiak hartuko duen azkeneko balioa" ken "i" izango da. Azken batean E espresioa n + 1 eta i-ren arteko distantzia da. Horrela i-ren balioa handitzen denean, n + 1 eta i-ren arteko distantzia txikiagoa izango da eta eman beharreko buelta-kopurua ere txikiagoa izango da.

Koloreen bidez asertzio batetik bestera dauden aldaketak nabarmendu dira.