



## H. Ledgardo uždavinys

Šį uždavinį sugalvojo žinomas programuotojas Henris Ledgargas (*Henry Ledgard*). Jis rašė, kad algoritmą siūlė patobulinti maždaug šimtui žmonių. Ir nė vienas nieko geresnio nepasiūlė.

Olimpiadai pateiktame uždavinyje sąlyga palengvinta – nurodyta, kad reikia atsisakyti kintamojo  $k$  ir vieno (iš dviejų) sąlyginio sakinio.

Pateiktame algoritme lyginami du gretimi masyvo elementai:  $i$ -asis ir  $(i-1)$ -asis. Tačiau kai žinoma, jog iki šiol rasta  $sk$  vienodų skaičių, galima iš karto nustatyti, ar toliau bus  $sk+1$  lygių skaičių, ar ne: nepamirškime, kad duotas surikiuotas masyvas. Jei  $i$ -asis ir  $(i-sk)$ -asis masyvo elementai lygūs, tai lygūs ir visi tarp jų esantys elementai. Tuomet vienodų skaičių bus nemažiau kaip  $sk+1$ . Taigi kintamojo  $k$  nebereikia.

```
function daug2 (a: array [m..n] of integer): integer;  
{ patobulinta funkcija }  
  var i,                { elemento indeksas }  
      r,                { ir elemento reikšmė }  
      sk: integer; { vienodų skaičius }  
begin  
  r := a[m];  
  sk := 1;  
  for i := m+1 to n do  
    if a[i] = a[i - sk]    { ar vienodi elementai, }  
      then begin          { nutolę per sk pozicijų }  
        r := a[i];  
        sk := sk + 1 { rasta ilgesnė lygių }  
      end;              { skaičių eilė }  
  daug2 := r  
end;
```