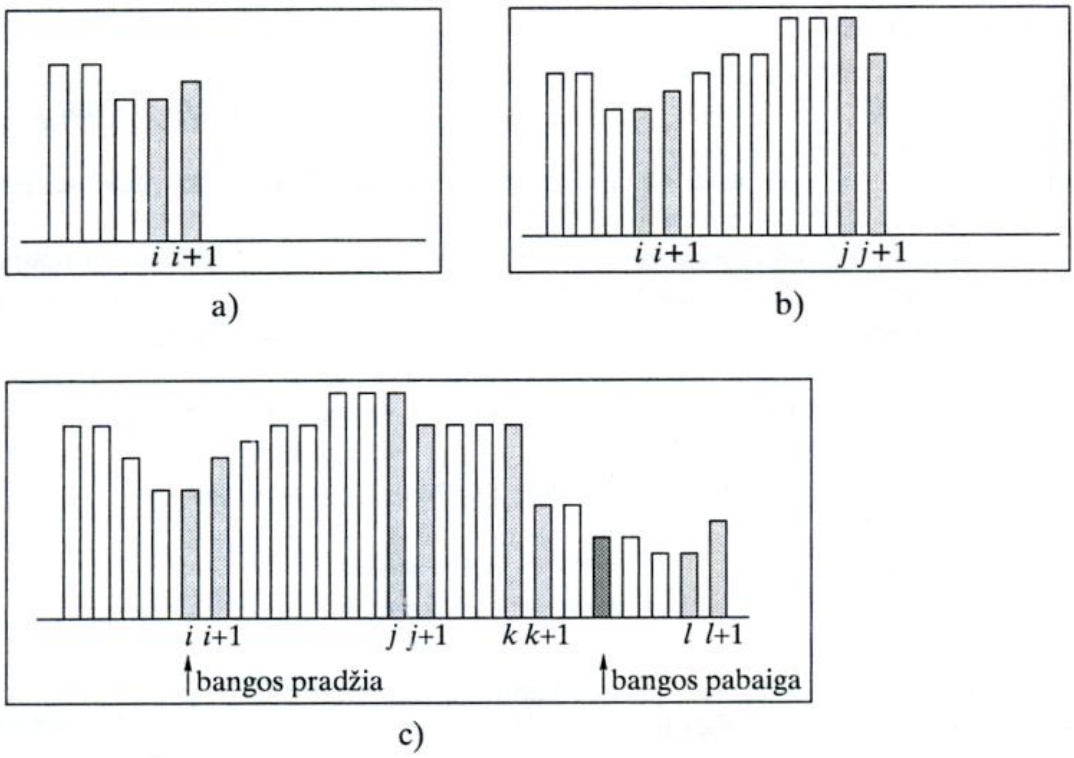
Kuoliukai

Šiame uždavinyje galima numatyti tris išskirtinius momentus, kuriuos būtina apgalvoti rašant algoritmą:

1) išgaubtos ir įgaubtos bangos „persikloja“;
2) gali iš viso nebūti išgaubtos ar įgaubtos bangos;
3) bangos galuose negali būti vienodo ilgio kuoliukų (žr. bangos apibrėžtį sąlygoje).

Kuoliukus numeruosime nuo 1 iki *n*. Panagrinėkime, kaip reikėtų ieškoti išgaubtos bangos. Paeiliui skaitome kuoliukų ilgius, kol randame du iš eilės einančius kuoliukus, kurių pirmasis būtų trumpesnis už antrąjį, t. y. $h_i < h_{i+1}$. Tai bus išgaubtos bangos pradžia. Įsimename šio kuoliuko numerį *i* (.a pav.).



90.2a–c pav.

Toliau skaitome kuoliukų ilgius. Bangos viršūnė bus pasiekta, kai surasime du iš eilės einančius kuoliukus, kurių pirmasis ilgesnis už antrąjį, t. y. $h_j > h_{j+1}$.

Jei šitokių kuoliukų nepavyktų rasti iki tvorelės pabaigos, tai tiriamoji banga nebūtų išgaubta.

Pasiekę bangos viršūnę toliau skaitome kuoliukų ilgius. Kai rasime du gretimus kuoliukus, kurių pirmasis bus trumpesnis už antrąjį, t. y. $h_l < h_{l+1}$, tai reikš, kad prasidėjo nauja banga ir tiriamoji banga jau pasibaigė. Tačiau jei bangos pabaigoje yra vienodo ilgio kuoliukų, tai pagal sąlygą tik pirmasis jų priklauso tiriamajai bangai. Todėl šalia paskutinio kuoliuko numerio l reikia žinoti ir numerį kuoliuko ties kuriuo prasideda vienodų kuoliukų eilė. Jei baigiantis bangai vienodų kuoliukų nebuvo, tai šis numeris bus lygus l, o jei bus vienodų kuoliukų grupė, tuomet jis rodys į pirmąjį kuoliuką iš vienodų grupės.

Aptikus naujos bangos pradžią beliks įvertinti ištirtos bangos ilgį: mat pradžia buvo įsiminta anksčiau.

Dabar panagrinėkime atvejį, kai banga užsibaigia tvorelės pabaiga. Jei tvora baigiasi keliais vienodo ilgio kuoliukais, tuomet bangos pabaiga bus laikomas pirmasis iš vienodų kuoliukas. Taip būtų sąlygoje pateiktame pavyzdyje.

Kitaip yra jei tvora baigiasi skirtingo aukščio kuoliukais (90.2 e pav.), tuomet pagal mūsų algoritmą bangos pabaiga būtų laikomas (j+1)-asis kuoliukas. Norėdami išvengti tokios situacijos, tvorelės gale prijungiame fiktyvų kuoliuką tokio pat ilgio kaip ir paskutinis.

Tyrinėjant įgaubtą bangą skirtumas bus tik tas, kad vietoj gretimų kuoliukų ilgių padidėjimo bus sumažėjimas, o vietoj sumažėjimo – padidėjimas.

```
procedure bangos (var f: text;
                  n: integer;      { n >= 2 }
                  var  max_išg,  max_įg:  integer);
  var ilg1, ilg2,                  { gretimų kuoliukų ilgiai }
      i,                           { kuoliuko numeris [0..n-1] }
      pra_išg, pra_įg,             { išgaubtos ir įgaubtos bangų pradžios }
      vienodu_pr:      integer;    { vienodo ilgio kuoliukų pradžia }
      vir_isg, vir_ig: boolean;    { pasiekta  bangos viršūn, }
```

```
  procedure bangų_tyrimas (l1, l2:  integer;
                           var pra: integer;   { bangos pradžia }
                           var vir: boolean;   { ar rasta viršūnė }
                           var max: integer);  { ilgiausia rasta banga }
  { vartojami globalūs kintamieji: i ir vienodų_pr }
    begin
      if (pra = -1) and (l1 < l2)            { prasideda pirma banga }
         then pra := i - 1
         else if (pra <> -1) and (l1 > l2) { pasiekta bangos viršūnė }
             then vir := true
             else if vir and ((l1 < l2) or (i = n)) { prasideda }
                     then { nauja banga arba baigėsi tvorelė }
                       begin
                         if max < vienodų_pr - pra + 1
                            then max := vienodų_pr - pra + 1;
                         pra := i - 1;
                         vir := false
                       end;
    end;   { bangų_tyrimas }

begin
  max_išg := 0;    max_įg := 0;
  pra_išg := -1;   pra_įg := -1;    { -1, nes kuoliukai numeruojami nuo 0
}
  vir_išg := false;  vir_įg := false;
  read (f, ilg2);
  for i := 1  to  n do    { pridedame dar vieną kuoliuką (su numeriu n) }
     begin                { tokio pat ilgio, kaip ir paskutiniojo  (su  }
       ilg1 := ilg2;      { numeriu n-1). Tuomet bus apdorota ir banga, }
        if  i < n         { kuri baigiasi su tvorelės pabaiga }
          then  read (f, ilg2);
        { išgaubtų bangų tyrimas }
        bangų_tyrimas (ilg1, ilg2, pra_išg, vir_išg, max_išg);
        { įgaubtų bangų tyrimas }
        bangų_tyrimas (ilg2, ilg1, pra_įg, vir_įg, max_įg);
        if ilg1 <> ilg2            { jei iš eilės bus keli vienodo ilgio }
           then  vienodu_pr := i;  { kuoliukai, tai vienodų_pr rodys }
                                   { pirmą iš jų. Jei kuoliukai skirtingo}
     end;                          { ilgio, tai vienodų_pr bus lygus i-1 }
end;   { bangos }
```