Sieninis kalendorius

Prieš pradedant spręsti uždavinį, reikia pastebėti, kad pro iškirptą langelį ne visuomet matysis skaičius. Gali matytis ir tuščias langelis.

Kaip ir daugelyje uždavinių su datomis bei kalendoriais, vienas pirmųjų darbų būna sužinoti kuria savaitės diena prasideda metai.

Žinodami tai galėsime dirbti toliau. Per metus sausio 1-moji pasislenka arba viena (jei metai paprastieji), arba dviem (jei metai keliamieji) dienomis. Pavyzdžiui, 1979 sausio 1-moji buvo pirmadienis. Kadangi 1979 metai nėra keliamieji, tai 1980 m. sausio 1-moji yra antradienis. Šie metai jau yra keliamieji, todėl 1981 m. sausio 1-moji pasislenka per dvi dienas: yra ketvirtadienį.

Atskaitos pradžia galime pasirinkti šio šimtmečio paskutiniuosius metus. Žinome, kad kitų (1999) metų sausio 1-moji bus penktadienį. Eidami atgal ligi duotųjų metų, sužinome, kuria savaitės diena prasideda metai.

```
type metai = 1583..1999;
     mėnuo = 1..12;
     diena = 0..31;  { langelis gali būti tuščias, tad turi būti ir 0 }
     data = record
              mt: metai;
              mn: mėnuo;
              dn: diena
            end;

function kel (met: metai): boolean;
{ ar metai keliamieji }
begin
  kel := (met mod 400 = 0) or
         (met mod 100 <> 0) and (met mod 4 = 0)
end;  { kel }
```

```
function pirma_diena (met: metai): diena;
{ randa kokia savaitės diena prasideda metai }
 var dien: diena;
     m, i: metai;
     posl: integer;
begin
  { 1999 metus pasirinkome atskaitos metais }
  dien := 5; m := 1999;
  for i := m-1 downto met do
    begin
      if kel (i) { randame poslinkį }
         then posl := 2
         else posl := 1;
      if dien - posl < 1
         then dien := dien + 7 - posl
         else dien := dien - posl;
    end;
  pirma_diena := dien
end; { pirma_diena }
```

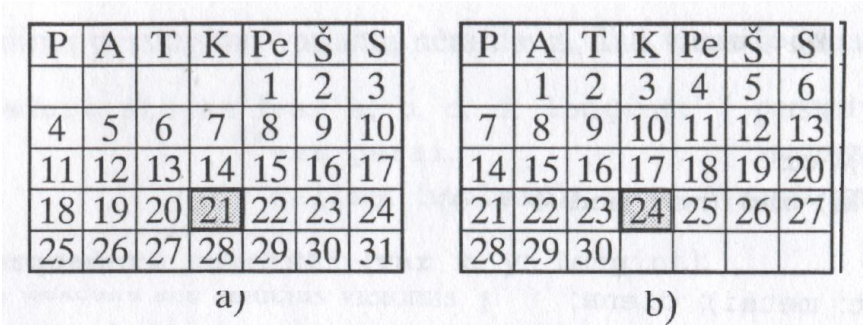
Žinodami kuria savaitės diena prasideda metai, sužinome kuria savaitės diena prasideda pirmais (duotasis) mėnuo ir kuria savaitės diena prasideda antrasis mėnuo, (t. y. mėnuo, kurio diena matysis pro iškirptą skylutę).

```
function dmėn (met: metai; mėn: mėnuo): diena;
{ kiek dienų turi met metų mėnuo mn }
begin
  case mėn of
    4, 6, 9, 11 : dmėn := 30;
    2: if kel(met) then dmėn := 29
                   else dmėn := 28;
    else dmėn := 31
  end
end; { dmėn }


function pir_mėn_dien (met: metai; mėn: mėnuo;
                       die: diena): diena;
{ randa kuria savaitės diena prasideda met metų mėn mėnuo }
{ kai žinoma, kuria savaitės diena prasideda metai met }
  var dien, i: integer;
begin
  dien := die;
```

```
  for i := 1 to mėn-1 do
    dien := dien + dmėn (met, i);
  if dien mod 7 = 0
     then pirma_mėn_dien := 7
     else pirma_mėn_dien := dien mod 7;
end; { pir_mėn_dien }
```

Sakykime, kad pirmasis mėnuo prasideda penktadienį, o antrasis antradienį. Iš žemiau pateikto pavyzdžio, matosi, kad visos antrojo mėnesio dienos bus pasislinkę į priekį per 5-2 = 3 dienas (žr. pav.).



P	A	T	K	Pe	Š	S
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

a)

P	A	T	K	Pe	Š	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

b)

Taigi, vietoj 21-os dienos matysis 24-oji.

```
function iškirptas_langelis (d: data; skyl: integer)
                                                  : diena;
{ randa skaičių, kuris matosi pro iškirptą langelį }
  var met_dien, mėn1_dien, mėn2_dien: diena;
      post: integer;
begin
  { rasime kokia savaitės diena prasideda duotieji metai }
  met_dien := pirma_diena (d.mt);
  { rasime kokia savaitės diena prasideda duotasis mėnuo }
  mėn1_dien := pir_mėn_dien (d.mt, d.mn, met_dien);
  { rasime kokia savaitės diena prasideda mėnuo, kurio diena }
  { matysis pro iškirptą langelį }
  mėn2_dien := pir_mėn_dien (d.mt, d.mn+skyl, met_dien);
  { rasime postūmį }
  post := mėn1_dien - mėn2_dien;
  if ((d.dn + post) <= 0) or
     (d.dn + post > dmėn (d.mt, d.mn+skyl))
     then iškirptas_langelis := 0
     else iškirptas_langelis := d.dn + post
end; { iškirptas_langelis }
```