Siųstuvai ir imtuvas

Kiekvieną reiškinį sudaro skaičiai, kintamieji bei tarp jų surašytos aritmetinės operacijos. Pradžioje tarkime, kad turime tik vieną siųstuvą ir imtuvą, pirmasis signalas išsiunčiamas laiko momentu 0. Siųstuvas skleidžia signalus kas T_1 sekundžių, o šie imtuvą pasiekia po L_1 sekundžių. Pirmasis signalas siųstuvą pasieks po L_1 sekundžių, o po to kas T_1 sekundžių imtuvą pasieks nauji signalai.

T. y. signalai imtuvą pasieks laiko momentais $L_1 + mT_1$, kur $m = 0, 1, ...$

Pavyzdžiui, $L_1 = 5$, $T_1 = 2$:

Laiko momentai	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	13	...
Siųstuvas	1		2		3		4		5		6		7		8
Imtuvas						1		2		3		4		5	

1 lentelė

Įjunkime dar vieną siųstuvą. Sakykime, kad $L_2 = 2$, $T_2 = 3$. Dabar pirmojo siųstuvo signalus žymėsime pliusais, antrojo – minusais.

Laiko momentai	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*	*19*	...
Pirmas siųstuvas	+		+		+		+		+		+		+		+		+		+		+
Antras siųstuvas	–			–			–			–			–			–			–		
Imtuvas			–			+, –		+	–	+		+, –		+	–	+		+, –		+	–

2 lentelė

Matome, kad nuo penkto laiko momento gauname vienodus periodus (jie lentelėje pažymėti kita spalva). Periodo ilgis – 6 laiko vienetai, kiekviename periode yra po vieną blogą signalą, taigi imtuvą pasieks 16.7% blogų signalų.

Bendru atveju blogi signalai gali ateiti dar neprasidėjus pirmajam periodui (šiame pavyzdyje laiko momentais 1, 2, 3, 4). Tačiau blogų signalų skaičius iki prasidedant pirmajam periodui yra fiksuotas (baigtinis), o pačių periodų (ir blogų signalų juose) skaičius – begalinis, nes siųstuvai veikia be perstojo. Todėl skaičiuojant blogų signalų procentą reikia nekreipti dėmesio į signalus pasiekiančius imtuvą ligi pirmojo periodo.

Periodiškai pasikartojančius signalus gavome panagrinėję vieną pavyzdį. Įrodysime kad tai teisinga bendru atveju.

Pradžioje sakykime, kad turime du siųstuvus ir laikykime kad $L_1 = L_2 = L$. Tuomet pirmasis abiejų siųstuvų signalas (blogas) imtuvą pasieks laiko momentu L. Antrasis blogas signalas pasieks imtuvą laiko momentu $L + mbk(T_1, T_2)$, čia *mbk* – mažiausias bendrasis kartotinis, t. y. kai galios lygybė $m_1T_1 = m_2T_2$. Trečiasis blogas signalas pasieks $L + 2mbk(T_1, T_2)$ ir t. t. Periodas prasidės laiko momentu L ir jo ilgis bus $mbk(T_1, T_2)$.

Imkime aukščiau pateiktą pavyzdį ir sakykime, kad $L_1 = L_2 = 2$, $T_1 = 2$, $T_2 = 3$.

Laiko momentai	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*	*19*	...
Pirmas siųstuvas	+		+		+		+		+		+		+		+		+		+		+
Antras siųstuvas	–			–			–			–			–			–			–		
Imtuvas			+ –		+	–	+		+ –		+	–	+		+ –		+	–	+		

3 lentelė

Analogiška situacija bus ir tada kai turėsime *n* siųstuvų. Laiko momentu L imtuvą pasieks visų siųstuvų pirmasis signalas. Blogi signalai pasieks imtuvą laiko momentais $m \times mbk(T_i, T_j)$, čia $m = 0, 1, 2...$ o *i*, *j* – bet kurių dviejų siųstuvų numeriai. Laiko momentu $mbk(T_1, T_2, ... T_n)$ imtuvą pasieks visų *n* siųstuvų signalai. Taigi periodas prasidės laiko momentu L ir jo ilgis bus $mbk(T_1, T_2, ... T_n)$.

Jei turėtume tik du imtuvus, blogų signalų ieškotume skaičiuodami *mbk*. Tačiau turint *n* signalų, taip ieškoti sunku, nes gali sutapti ne tik dviejų bet trijų ar daugiau skaičių *mbk*.

Pereikime prie atvejo su dviem siųstuvais, kai $L_1 \neq L_2$. Tarkime, kad $L_1 \geq L_2$. Tuomet periodas prasidės tuo laiko momentu L_1 (t. .y kai abiejų siųstuvų signalai jau bus pasiekę imtuvą po vieną kartą). Mat nuo to momento imtuvą pasiekiantys signalai nebepriklausys nuo L_i, tik nuo T_i. Aukščiau pateiktame pavyzdyje (2 lentelė) laiko momentu L_1 imtuvą pasiekia ne tik pirmojo, bet ir antrojo siųstuvo signalas. Todėl situacija bus lygiai tokia pati kaip ir tada, kai $L_1 = L_2$. Panagrinėkime kitokį pavyzdį. Sakykime, kad $L_1 = 3$, $T_1 = 2$, $L_2 = 1$, $T_2 = 3$

Laiko momentai	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*	*19*	...
Pirmas siųstuvas	+		+		+		+		+		+		+		+		+		+		+
Antras siųstuvas	–			–			–			–			–			–			–		
Imtuvas		–		+	–	+		+		+	–	+		+		+	–	+		+	
								–						–						–	

4 lentelė

Periodas prasidės laiko momentu 3, jo ilgis – 6.

Analogiškai, kai turime *n* siųstuvų, periodas prasidės tuo laiko momentu, kai imtuvą pasieks pirmasis siųstuvo, kurio signalai eina ilgiausiai (didžiausias L) signalas. Periodo ilgis bus *mbk* (T_1, T_2, ... T_n).

Sprendžiant uždavinį patogiausia modeliuoti kiekvieną laiko momentą iš pirmojo periodo, t. .y iš intervalo [L_{max}, L_{max} + *mbk* (T_1, T_2, ... T_n)[.

Na, o blogų signalų procentas lygus:

blogų_signalų_skaičius × 100 / *gautų_signalų_skaičius_viename_periode*

```
const MAX_N = 20;  { maksimalus siųstuvų skaičius }
type lentelė = array [1..MAX_N] of integer;

function dbd (x, y: integer): integer;
{ didžiausias bendras daliklis }
begin
  if x = 0 then dbd := y
           else dbd := dbd (y mod x, x)
end;  { dbd }

function bmk (x, y: integer): integer;
{ mažiausias bendrasis kartotinis }
begin
  bmk := x div dbd (x, y) * y
end;  { bmk }
```

```
function blogi_signalai (n: integer; t, l: lentelė)
                                                  : integer;
{ apskaičiuoja blogų signalų, kuriuos gauna imtuvas, procentą }
  var blogi, { blogų signalų skaičius }
      viso, sign , l_max, kart, i, j: integer;
      pas: lentelė; { lentelė, rodanti kuriais laiko momentais }
                    { atitinkamo siųstuvo signalas pasiekia imtuvą }
begin
  { rasime l_max }
  l_max := 0;
  for i := 1 to n do
    if l_max < l[i]
       then l_max := l[i];
  { rasime bmk(T1, T2, ... Tn) }
  kart := 1;
  for i := 1 to n do
    kart := bmk (kart, t[i]);
  { rasime kada pirmasis kiekvieno siųstuvo signalas pasiekia imtuvą }
  pas := l;
  { rasime kada kiekvieno siųstuvo pirmasis signalas siekia imtuvą }
  { nagrinėjamame periode }
  for i := 1 to n do
    while pas[i] < l_max do
      pas[i] := pas[i] +  t[i];
  { modeliuosime kiekvieną laiko momentą pirmajame periode }
  blogi := 0;
  viso := 0;
  for i := l_max to l_max + (kart-1) do
    begin
      sign := 0; { momentu i imtuvą pasiekiančių signalų skaičius }
      for j := 1 to n do
        if pas[j] = i
           then sign := sign + 1;
      if sign > 1
         then blogi := blogi + 1;
      if sign > 0
         then viso := viso + 1;
      { randame tolesnius laiko momentus kada signalai pasieks imtuvą }
      for j := 1 to n do
        if pas[j] <= i
           then pas[j] := pas[j] + t[j]
    end;
  blogi_signalai := round (blogi/viso*100)
  { tokia veiksmų tvarka nesukels perpildymo }
end; { blogi_signalai }
```