Skaičių lentelės atstatymas

Trūkstamą lentelės elementą galima mėginti atstatyti pagal toje eilutėje ar stulpelyje esančius skaičius, t. y. iš eilutės ar stulpelio sumos atimti visus kitus elementus ir ieškoti tų elementų sumos. Taip galima atstatyti trūkstamą skaičių, jei eilutėje ar stulpelyje nėra daugiau nežinomų skaičių. Jei yra, reikia mėginti atstatyti tolesnį trūkstamą skaičių.

Vieną kartą patikrinę visas eilutes ir stulpelius, uždavinį retai kada išspręsime. Panagrinėkime tokį pavyzdį:

222	333	–	1	–
–	–	4	–	14
224	–	–	6	1014

Patikrinę eilutes ir stulpelius pirmą kartą gausime:

222	333	–	1	1000
2	–	4	5	14
224	–	–	6	1014

Matome, kad atstatyti ne visi skaičiai, nors akivaizdu, kad dar galima juos atstatyti. Patikrinę antrą kartą gauname:

222	333	444	1	1000
2	3	4	5	14
224	336	448	6	1014

Taigi jei patikrinus lentelę nors vieną skaičių pavyko atstatyti, reikia viską kartoti iš pradžių. Taip darome, kol atstatome skaičius arba kol tokiu būdu nebegalime atstatyti skaičių. Jei pavyko atstatyti, būtina patikrinti, ar atstatytoje lentelėje nėra klaidų. Mat atstatinėjant galime nepamatyti, kad pradinėje lentelėje įrašyti nekorektiški skaičiai.

```
const max = 10;  { maksimalus lentelės dydis }
type lentelė = array [1..max, 1..max] of integer;
     masyvas = array [1..max] of integer;

{ žemiau surašyti globalieji kintamieji }
var eil, st: integer;  { eilučių ir stulpelių skaičius lentelėje }
      lent: lentelė;  { pradinė lentelė }
```

```
procedure trūksta (var nutrinta: integer;  { kiek nutrintų skaičių }
                   var e, s: masyvas);
{ suskaičiuojame kiek yra trūkstamų elementų kiekviename stulpelyje }
{ bei eilutėje ir kiek iš viso yra nutrintų skaičių }
  var i, j: integer;
begin
  nutrinta := 0;
  for i := 1 to eil do
    e[i] := 0;
  for i := 1 to st do
    s[i] := 0;
  for i := 1 to eil do
    for j := 1 to st do
      if lent[i, j] = -maxint
         then begin
                e[i] := e[i] + 1;
                s[j] := s[j] + 1;
                nutrinta := nutrinta + 1
              end
end;  { trūksta }

  function teisinga (atst: lentelė): boolean;
  { ar nėra klaidų atstatytoje lentelėje }
    var sum_eil, sum_st: masyvas;  { atitinkamų eilučių ir stulpelių sumos }
        gerai: boolean;
        i, j: integer;
  begin
    gerai := true;
    for i := 1 to eil do
      sum_eil[i] := 0;
    for i := 1 to st do
      sum_st[i] := 0;
    for i := 1 to eil do
      for j := 1 to st do
         begin
           if j = st   { jei susumavome eilutę }
              then gerai := gerai and (sum_eil[i] = atst[i, j])
              else sum_eil[i] := sum_eil[i] + atst[i, j];
           if i = eil  { jei susumavome stulpelį }
              then gerai := gerai and (sum_st[j] = atst[i, j])
              else sum_st[j] := sum_st[j] + atst[i, j]
         end;
    teisinga := gerai
  end;  { teisinga }
```

```
procedure atstatyti (var atst: lentelė; { atstatyta lentelė }
                     var klaida, { ar yra lentelėje klaidų }
                     per_daug: boolean { ar per daug prarasta skaičių });
  var e, s: masyvas;
      { trūkstamų skaičių kiekis atitinkamoje eilutėje ar stulpelyje }
      nutrinta, i, j, k, suma: integer;
      atstatyta: boolean; { ar peržiūrėję lentelę atstatėme nors vieną skaičių }
begin
  { suskaičiuojame, kiek yra trūkstamų elementų kiekviename stulpelyje }
  { ir eilutėje ir kiek viso nutrinta skaičių }
  trūksta (nutrinta, e, s);
  atst := lent;
  { bandysime atstatyti skaičius }
  repeat
    atstatyta := false;
    for i := 1 to eil do
       for j := 1 to st do
         if (atst[i, j] = -maxint) then { jei skaičius nutrintas }
            if (e[i] = 1) or (s[j] = 1) then
              begin
                nutrinta := nutrinta - 1;
                atstatyta := true;
                suma := 0;
                if e[i] = 1 then
                   begin { jei eilutėje trūksta vieno skaičiaus }
                     for k := 1 to st-1 do
                       if k <> j then
                         suma := suma + atst[i, k];
                         if j = st then
                           atst[i, j] := suma
                         else atst[i, j] := atst[i, st] - suma
                   end
                else begin { jei stulpelyje trūksta vieno skaičiaus }
                       for k := 1 to eil - 1 do
                         if k <> i then
                           suma := suma + atst[k, j];
                         if i = eil then
                            atst[i, j] := suma
                         else atst[i, j] := atst[eil, j] - suma
                     end;
                e[i] := e[i] - 1;
                s[j] := s[j] - 1
              end;
  until not atstatyta or (nutrinta = 0);
```

```
   { tikrinama, ar nėra klaidų }
   if nutrinta <> 0
      then per_daug := true
      else begin
             per_daug := false;
             klaida := not teisinga (atst)
           end
end;  { atstatyti }
```