Reiškinys

Kiekvieną reiškinį sudaro skaičiai, kintamieji bei tarp jų surašytos aritmetinės operacijos.

Pradinis duomuo – simbolių eilutė. Tikriname jos pirmąjį simbolį – jei skaitmuo, vadinasi, reiškinys prasideda skaičiumi, jei raidė – kintamuoju. Tuomet nustatome objekto (skaičiaus ar kintamojo) ilgį, patį objektą pašaliname iš eilutės.

Po skaičiaus ar kintamojo eina aritmetinė operacija. Ji sudaryta arba iš vieno simbolio (+, -, *, /), arba iš trijų (div, mod). Pašaliname iš simbolių eilutės operaciją ir toliau kartojame veiksmus – tikriname, ar likęs reiškinys prasideda skaičiumi, ar kintamuoju.

Reikia nepamiršti keleto dalykų:

1) reiškinio pradžioje esantis minusas rodo neigiamą skaičių, o ne aritmetinę operaciją;
2) pradinis duomuo baigiasi lygybe;
3) po paskutiniojo objekto (kintamojo ar skaičiaus) nebebus aritmetinės operacijos;
4) reiškinyje gali būti tarpų, todėl iš simbolių eilutės pašalinus objektą, reikia pašalinti ir po jo einančius tarpus;

```
procedure tarpai (var s: string);
{ pašalina tarpus iš reiškinio s pradžios }
begin
  while (s[1] = ' ') do
    delete (s, 1, 1)
end; { tarpai }

procedure skaičiuoti (s: string; { pradinis reiškinys }
                      var sk, kint, op: integer);
                     { kiek reiškinyje yra skaičių, kintamųjų ir operacijų }
{ suskaičiuoja kiek reiškinyje yra skaičių, kintamųjų ir operacijų }
  var ilgis: integer;
begin
  tarpai(s); { pašalinami tarpai reiškinio priekyje }
  { numetamas minusas reiškinio priekyje }
  if s[1] = '-'
     then delete (s, 1, 1);
  tarpai (s);
  sk := 0;
```

```
  kint := 0;
  op := 0;
  while s <> '=' do    { kol neišnagrinėjome viso reiškinio }
    begin
      { reiškinys prasidės arba skaičiumi, arba kintamojo vardu }
      if s[1] in ['0'..'9']
      then begin { randamas skaičiaus ilgis }
             skaičius (s, ilgis);
             sk := sk + 1;
           end
      else begin { randamas kintamojo ilgis }
             kintamasis (s, ilgis);
             kint := kint + 1;
           end;
      { pašaliname tą skaičių ar kintamąjį }
      delete (s, 1, ilgis);
      tarpai(s);
      { jei tai ne reiškinio pabaiga, randame aritmetinę operaciją }
      if s <> '='
      then begin
             if s[1] in ['+', '-', '*', '/']
                then ilgis := 1
                else ilgis := 3;
             { operacijos div arba mod }
             delete (s, 1, ilgis);
             op := op + 1
           end;
      tarpai(s);
    end
end;  { skaičiuoti }
```

Reiškinys gali prasidėti skaičiumi. Sveikąjį skaičių sudaro tik skaitmenys, realųjį – gali sudaryti skaitmenys ir simboliai „+“, „-“, bei raidė E (arba e):

```
0.15
0.89e-10
0.578E+3
7e-8
48.3e6
```

Skaičiaus pabaigos ieškome taip: kol nagrinėjami simboliai yra skaitmenys arba ‘.’, einame toliau. Jei aptikome kitokį simbolį ir tas simbolis nėra raidė E (e), radome skaičiaus pabaigą. Jei radome simbolį E (e), tuomet praleidžiame vieną simbolį (tai bus arba laipsnio rodiklio ženklas, arba paties rodiklio pirmasis skaitmuo) bei visus po to einančius skaitmenis.

```
procedure skaičius (s: string; var ilgis: integer);
{ s – reiškinys, prasidedantis skaičiumi }
{ procedūra randa jo ilgį }
begin
  ilgis := 2;
  while s[ilgis] in ['0'..'9', '.'] do
    ilgis := ilgis + 1;
  { gal būt tai dar ne skaičiaus pabaiga, o realaus skaičiaus laipsnis }
  if s[ilgis] in ['e', 'E']
     then begin
            ilgis := ilgis + 2;
            while s[ilgis] in ['0'..'9'] do
              ilgis := ilgis + 1
          end;
  ilgis := ilgis - 1
end;  { skaičius }
```

Jei reiškinys pasideda kintamuoju, rasti reiškinio pabaigą nesunku – tereikia rasti pirmąjį simbolį, kuris nėra nei raidė, nei skaitmuo. Mat tarp kintamojo ir aritmetinių operacijų `div` ar `mod` būtinai turi būti paliktas tarpas (kitaip bus laikoma kintamojo vardu).
Reikia nepamiršti, kad kintamojo vardą gali sudaryti ne tik mažosios, bet ir didžiosios raidės.

```
procedure kintamasis (s: string; var ilgis: integer);
{ s – reiškinys, prasidedantis kintamuoju }
{ procedūra randa jo ilgį }
begin
  ilgis := 1;
  while s[ilgis] in ['A'..'Z', 'a'..'z', '0'..'9'] do
    ilgis := ilgis + 1;
  ilgis := ilgis - 1
end;  { kintamasis }
```