



Sustojimo uždavinys

1936 m. anglų matematikas Alanas Turingas įrodė, kad nėra ir negali būti algoritmo, bendru atveju nustatančio, ar duotoji programa su duotaisiais pradiniais duomenimis baigtų darbą, ar dirbtų amžinai. Matematikos pasaulyje šis klausimas vadinamas „Sustojimo uždaviniu“. Nepaisant įrodyto fakto, programos, nagrinėjančios kitas programas, išlieka įdomia tema.

Jums teks spręsti sustojimo uždavinį. Tačiau neišsigąskite.

Nagrinėsime programas, kurias sudaro tam tikras rinkinys paprogramių. Programoje paprogramės surašytos viena po kitos ir viena paprogramė negali būti kitos viduje. Vienintelė galima komanda yra kitos paprogramės iškvietimas.

Žemiau esančioje lentelėje aprašyta nagrinėjamos „programavimo kalbos“ sintaksė:

Kalbos konstrukcija	Pavyzdys	Paaškinimas
PROC <i>VARDAS</i>	PROC MAIN	Antraštė, kuria prasideda kiekviena paprogramė. Pavyzdyje aprašyta paprogramė pavadinta MAIN.
END	END	Paprogramės pabaigą nurodanti leksema.
CALL <i>VARDAS</i>	CALL ANTROJI	Kitos paprogramės iškvietimo komanda. Pavyzdyje pateikta komanda iškviečia paprogramę ANTROJI. Baigus vykdyti iškviestą paprogramę, tęsiamas iškvietusios paprogramės vykdymas.

Programoje visuomet egzistuoja paprogramė, vardu **MAIN**, nuo kurios pradedamas programos vykdymas. Baigus vykdyti paprogramę **MAIN**, programa baigia darbą. Panagrinėkime programos pavyzdį:

```
1:  PROC   P1
2:  CALL   P2
3:  END
4:  PROC   P2
5:  END
6:  PROC   REK
7:  CALL   REK
8:  END
9:  PROC   MAIN
10: CALL   P1
11: CALL   REK
12: END
```

Programa pradedama vykdyti nuo 9 eilutės. Toliau seka paprogramės **P1** iškvietimas – pradedama vykdyti **P1** (1 eilutė). **P1** savo ruožtu iškviečia **P2**, tačiau ši iš karto baigia savo darbą, ir tęsiamas **P1** vykdymas, kuri taip pat nebeturi darbo. Tęsiant **MAIN** vykdymą iškviečiama paprogramė **REK**. Ši paprogramė rekursyviai iškviečia pati save ir tampa aišku, kad programos vykdymas niekada nebus baigiamas.



Užduotis. Parašykite programą, kuri nustatytų, ar pateikta programa baigs savo darbą, ar ne, ir rastų sėkmingai įvykdytų programos eilučių skaičių.

Jei ta pati eilutė įvykdoma n kartų, tai skaičiuojamas kiekvienas įvykdymas ir laikoma, kad įvykdyta n eilučių. Taip pat laikoma, kad vykdomos ir eilutės, kuriose yra paprogramių antraštės, bei eilutės, kurios nurodo paprogramių pabaigą.

Tuo atveju, kai programa nebaigia darbo, „sėkmingomis“ laikomos eilutės, įvykdytos prieš programai iškviečiant dar nebaigtą vykdyti paprogramę. Nagrinėtame pavyzdyje buvo sėkmingai įvykdytos 9 eilutės.

Pradiniai duomenys. Pirmoje eilutėje įrašytas programos eilučių skaičius K .

Kiekvienoje eilutėje yra lygiai viena kalbos konstrukcija, užrašyta didžiosiomis lotyniškėmis raidėmis. Lekšemos PROC ir CALL nuo paprogramės vardo atskirtos lygiai vienu tarpo simboliu. Paprogramės vardas gali būti sudarytas tik iš DIDŽIŲJŲ lotyniškų raidžių ir skaitmenų.

Pateikta programa yra korektiška: atitinka aprašytą sintaksę, kviečiamos tik egzistuojančios paprogramės, nėra dviejų paprogramių tuo pačiu vardu, yra paprogramė MAIN.

Pradiniai duomenys tokie, kad rezultatas neviršys 1 000 000.

Rezultatai. Jei pateiktoji programa baigia darbą, į pirmąją eilutę reikia išvesti žodį BAI-GIA. Priešingu atveju į pirmąją eilutę turi būti išvedamas žodis NEBAIGIA.

Į antrąją eilutę reikia išvesti sėkmingai įvykdytų programos eilučių skaičių.

Pavyzdžiai.

Pradiniai duomenys	Rezultatai	Paaškinimas
9 PROC P1 CALL P2 END PROC MAIN CALL P1 END PROC P2 CALL P1 END	NEBAIGIA 5	Programa nebaigia darbo. Vykdomos tokios programos eilutės (dešinėje surašytos pradėtos ir nepabaigtos vykdyti paprogramės): 4 MAIN 5 MAIN, kviečiama P1 1 MAIN, P1 2 MAIN, P1, kviečiama P2 7 MAIN, P1, P2 8 MAIN, P1, P2, kviečiama P1 1 MAIN, P1, P2, P1 Klaida aptinkama vykdant 8 eilutę, nes paprogramė P2 iškviečia dar nebaigtą vykdyti paprogramę P1. Tad iki rekursyvaus iškvietimo (dėl kurio programa nesustos) buvo įvykdytos penkios eilutės.



Pradiniai duomenys	Rezultatai	Paiškinimas
8 PROC MAIN CALL VEIK END PROC EMPTY CALL EMPTY END PROC VEIK END	BAIGIA 5	Programa baigia darbą. Įvykdomos komandos, esančios 1, 2, 7, 8, 3 eilutėse. Atkreipiame dėmesį, kad paprogramė EMPTY rekursyvi ir ją iškvietus, programa nebaigtų darbo. Tačiau vykdant programą ši paprogramė neiškviečiama, tad programa yra baigtinė.

Ribojimai. $1 < K \leq 1000$.