

Lietuvos mokinių informavimo ir techninės kūrybos centras
Matematikos ir informatikos institutas

Devynioliktoji moksleivių informatikos olimpiada

TREČIOJO ETAPO I DALIES UŽDAVINIAI

Vyresniųjų grupė

(X–XII klasės)

Vilnius, 2008

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	Paiškinimas
PROC VARDAS	PROC MAIN	Antraštė, kuria prasideda kiekviena paprogramė. Pavyzdyje aprašyta paprogramė pavadinta MAIN.
END	END	Paprogramės pabaigą nurodanti lexema.
CALL VARDAS	CALL ANTROJI	Kitos paprogramės iškvietimo komanda. Pavyzdyje pateikta komanda iškviečia paprogramę ANTROJI. Baigus vykdyti iškviestą paprogramę, tęsiamas iškviestusios 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

Pradiniai duomenys pateikti faile **sustojimas.in**. Pirmoje eilutėje įrašytas programos eilučių skaičius K ($1 < K \leq 1000$).

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 rezultatų failo **sustojimas.out** pirmąją eilutę reikia išvesti žodį BAIGIA. Priešingu atveju į pirmąją failo eilutę turi būti išvedamas žodis NEBAIGIA.

Į antrąją eilutę reikia išvesti sėkmingai įvykdytų programos eilučių skaičių.

Pavyzdžiai

Pradiniai duomenys	Rezultatai	Paaiškinimai
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.	

<i>Pradiniai duomenys</i>	<i>Rezultatai</i>	<i>Paaškinimai</i>
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.

Pastabos ir vertinimas

Sprendimas vertinamas iki 100 taškų:

- 90 taškų už programą;
- 10 taškų už programavimo stilių.

Bus vertinami tik tie darbai, kuriuos priims *Programavimo varžybų aptarnavimo sistema*. Programavimo stilius vertinamas tik tuomet, jei programa už testus surenka 50% ar daugiau taškų. C kalba rašomos programos turi būti užbaigiamos **return 0**.

Sistemai reikia pateikti failą, kuriame būtų:

- 1) identifikuojantis komentaras (žr. toliau pateiktą lentelę);
- 2) komentaras, kuriame nurodoma Jūsų vardas, pavardė, mokykla, klasė;
- 3) programa, sprendžianti duotąjį uždavinį.

Failas, kuriame įrašytas sprendimas, **BŪTINAI turi prasidėti tokiu komentaru:**

<i>Jei programuojama</i>		
<i>Paskalio kalba</i>	<i>C kalba</i>	<i>C++ kalba</i>
{ TASK: SUSTOJIMAS LANG: PASCAL }	/* TASK: SUSTOJIMAS LANG: C */	/* TASK: SUSTOJIMAS LANG: C++ */

Vykdymo laikas – 1 sek., procesorius – Celeron 1,3 GHz.

Kavos algebra

Linas yra informatikas ir, nors sako, kad mėgsta rytą, iš tiesų beveik visus darbus atlieka naktimis. O naktimis, kad neimtų miegas, jis geria... kavą.

Kaip ir dauguma informatikų, Linas yra tingus, todėl įprato pirkti tirpią „Nescafe“ kavą, nes ją lengviausia pasigaminti. Jų yra įvairių rūšių. Pavyzdžiui, «2 in 1» pakelyje yra kavos ir pieno miltelių, o «3 in 1» dar be to yra cukraus.

„«2 in 1», «3 in 1»... Hmm, sumaišius išeitų «5 in 2» kava“, – apsidžiaugė Linas! Ir apimtas įkvėpimo samprotavo toliau:

Tegu turime kavas « p in r » ir « s in t ». Tuomet jų mišinys bus kava « $p+s$ in $r+t$ ». Kuriuos iš turimų N pakelių kavų « a_1 in 1», « a_2 in 1», ..., « a_N in 1» reikia sumaišyti, kad gautume kavą « b in c »?

Užduotis. Parašykite programą, sprendžiančią šį uždavinį.

Pradiniai duomenys

Pradiniai duomenys pateikiami faile **kava.in**. Pirmoje eilutėje įrašyti trys sveikieji skaičiai b , c ir N . Tolesnėse eilutėse pateikti turimus kavos pakelius apibūdinantys skaičiai a_1 , a_2 , ..., a_N , po vieną eilutėje.

Visi skaičiai yra teigiami. Galioja apribojimai: $a_i \leq 1\,000$, $b \leq 10\,000$, $c \leq N \leq 100$.

Rezultatai

Į rezultatų failą **kava.out** programa turi bet kokia tvarka išvesti numerius pakelių, kurių turinį sumaišius būtų gaunama kava « b in c ». Kiekvienas pakelis gali būti panaudojamas tik vieną kartą. Jei kavos « b in c » niekaip neįmanoma gauti, tuomet rezultatų failo pirmoje eilutėje turi būti įrašytas skaičius 0 (nulis). Jei galimi keli sprendiniai, programa turi išvesti bet kurį iš jų.

Pavyzdžiai

Pradiniai duomenys	Rezultatai	Paaiškinimai
5 2 3 2 3 1	1 2	Sumaišę pirmo ir antro pakelio turinį gauname kavą «5 in 2». Sprendinį galima išvesti ir atvirkščia tvarka: pirma 2, po to 1.



<i>Pradiniai duomenys</i>	<i>Rezultatai</i>	<i>Paaškinimai</i>
6 2 3 2 3 1	0	Galima gauti kavą «6 in 3», bet «6 in 2» gauti neįmanoma.
12 3 6 6 7 5 2 3 2	2 4 5	Kitas galimas sprendinys būtų gaunamas sumaišius antro, penkto ir šešto pakelio turinius, kadangi šeštame pakelyje yra tokia pati kava kaip ir ketvirtame. Programa gali pateikti bet kurį sprendinį.

Pastabos ir vertinimas

50% testų galioja ribojimas $N \leq 25$.

Sprendimas vertinamas iki 100 taškų:

- 80 taškų už programą;
- 20 taškų už idėjos aprašymą.

Bus vertinami tik tie darbai, kuriuos priims *Programavimo varžybų aptarnavimo sistema*. C kalba rašomos programos turi būti užbaigiamos **return 0**. Sistemai reikia pateikti failą, kuriame būtų:

- 1) identifikuojantis komentaras (žr. toliau pateiktą lentelę);
- 2) komentaras, kuriame nurodoma Jūsų vardas, pavardė, mokykla, klasė;
- 3) komentaras, kuriame yra trumpas uždavinio sprendimo idėjos aprašas;
- 4) programa, sprendžianti duotąjį uždavinį.

Failas, kuriame įrašytas sprendimas, **BŪTINAI turi prasidėti tokiu komentaru:**

<i>Jei programuojama</i>			<i>Jei pateikiamas tik idėjos aprašymas</i>
<i>Paskalio kalba</i>	<i>C kalba</i>	<i>C++ kalba</i>	
{ TASK: KAVA LANG: PASCAL }	/* TASK: KAVA LANG: C */	/* TASK: KAVA LANG: C++ */	/* TASK: KAVA LANG: NO */

Vykdymo laikas – 1sek., procesorius – Celeron 1,3 GHz.

Olimpiados rezultatai bus paskelbti iki vasario 23 d.

Informatikos olimpiadų svetainėje : <http://ims.mii.lt/olimp>