

LATVIJAS REPUBLIKAS 19. INFORMĀTIKAS OLIMPIĀDES
II POSMA UZDEVUMI
JAUNĀKAJAI (7.-9. klašu) GRUPAI



1. "IEKAVAS"

(50 punkti)*

Korektu iekavu izteiksmi definēsim šādi:

- () ir korekta iekavu izteiksme;
- [] ir korekta iekavu izteiksme;
- {} ir korekta iekavu izteiksme;
- ja A ir korekta iekavu izteiksme, tad (A) arī ir korekta iekavu izteiksme;
- ja A ir korekta iekavu izteiksme, tad [A] arī ir korekta iekavu izteiksme;
- ja A ir korekta iekavu izteiksme, tad {A} arī ir korekta iekavu izteiksme;
- ja A un B ir korektas iekavu izteiksmes, tad AB arī ir korekta iekavu izteiksme.

Tā, piemēram, ([(){}), [[]]] un ()((())) ir korektas iekavu izteiksmes, bet)(, [[]]} un () - nav. Uzrakstiet programmu, kas dotai simbolu virknei nosaka, vai tā ir korekta iekavu izteiksme!

Ievaddati

Teksta failam `iekavas.dat` ir trīs rindas. Katrā no tām dota viena simbolu virkne, kas sastāv tikai no simboliem '(', ')', '[', ']', '{', '}''. Katrā simbolu virknē ir vismaz viens un ne vairāk kā 250 simboli.

Izvaddati

Teksta failam `iekavas.rez` jāsaturs tieši trīs rindas. Faila *i*-tajā rindā ($1 \leq i \leq 3$) jāizvada 'IR', ja ievaddatu faila *i*-tajā rindā ir korekta iekavu izteiksme, vai 'NAV', ja izteiksme nav korekta.

Piemēri

| Ievaddati (fails <code>iekavas.dat</code>) | Izvaddati (fails <code>iekavas.rez</code>) |
|---|---|
| () | IR |
| {[][][]][[]][[]][[]] | NAV |
| {{{{}}}} | IR |

| Ievaddati (fails <code>iekavas.dat</code>) | Izvaddati (fails <code>iekavas.rez</code>) |
|---|---|
| ([]) | NAV |
| [[]] | IR |
| []() | NAV |

*) Katra testa izpildei tiks dota viena sekunde.
Katram uzdevumam būs 10 testi. Par pilnīgi pareizu laikā izpildītu testu tiks piešķirti 5 punkti.

2."LUTAUSIS"

(50 punkti)*

Lutausim failā `lutaussis.dat` bija jāpieraksta augoša naturālu skaitļu virkne, katru skaitli pierakstot savā rindā. Ja kāds skaitlis bija tikpat garš kā iepriekšējais un bija izsakāms kā "iepriekšējā skaitļa sākuma fragments + atšķirīgā astes daļa", tad Lutausis (slinkuma dēļ, protams!) to pierakstīja kā "...aste", kur punktu skaits pieraksta sākumā atbilda neuzrakstīto ciparu skaitam. Faila beigās Lutausis apzīmēja ar skaitli 0.

Piemēram:

| Bija jāpieraksta: | Lutausis pierakstīja: |
|-------------------|-----------------------|
| 9 | 9 |
| 27 | 27 |
| 29 | .9 |
| 303 | 303 |
| 309 | ..9 |
| 311 | .11 |
| 319 | ..9 |
| 3192 | 3192 |
| | 0 |

Diemžēl, pārsūtot failu Sprīdītīm pa elektronisko pastu, punkti rindu sākumā tika pazaudēti un Sprīdītis saņēma šāda satura failu:

| Sprīdītis saņēma: |
|-------------------|
| 9 |
| 27 |
| 9 |
| 303 |
| 9 |
| 11 |
| 9 |
| 3192 |
| 0 |

Uzrakstiet programmu, kas no Sprīdīša saņemtā faila aprēķina sākotnējo skaitļu summu!

Ievaddati

Teksta fails `lutaussis.dat`. Zināms, ka sākotnējā virknē bija ne vairāk kā 65535 skaitļi un visu sākotnējo skaitļu summa nepārsniedza 2000000000.

Izvaddati

Teksta faila `lutaussis.rez` vienīgajā rindā jāizvada naturāls skaitlis - visu sākotnējo skaitļu summa.

Piemēri

| Ievaddati (fails <code>lutaussis.dat</code>) | Izvaddati (fails <code>lutaussis.rez</code>) |
|---|---|
| 1 | 6 |
| 2 | |
| 3 | |
| 0 | |

| Ievaddati (fails <code>lutaussis.dat</code>) | Izvaddati (fails <code>lutaussis.rez</code>) |
|---|---|
| 9 | 4499 |
| 27 | |
| 9 | |
| 303 | |
| 9 | |
| 11 | |
| 9 | |
| 3192 | |
| 0 | |

*) Katra testa izpildei tiks dota viena sekunde.

Katram uzdevumam būs 10 testi. Par pilnīgi pareizu laikā izpildītu testu tiks piešķirti 5 punkti.

3."REGATE"

(50 punkti)*

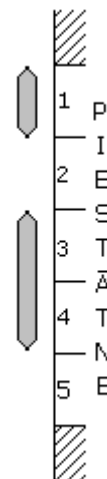
Uz ikgadējo regati ierodas K kuģi. Tiem visiem ir jāpietauvojas pie Galvenās Piestātnes. Galvenā Piestātne ir gara taisna krastmala, kas sadalīta N vienāda garuma *posmos*. Par katru kuģi ir zināms, ka tas pietauvojoties aizņem noteiktu veselu skaitu pēc kārtas esošu posmu. Aizņemto posmu skaitu sauksim par kuģa *garumu*. Pie katra posma drīkst pietauvoties ne vairāk kā viens kuģis. Ir zināms, ka visu kuģu garumu summa nepārsniedz N.

Kuģi pietātnē pienāk pēc kārtas. Kuģis var pietauvoties tikai tad, ja pietātnē var atrast tik daudz brīvus posmus pēc kārtas, ka to skaits nav mazāks par kuģa garumu. Ja pietauvoties iespējams vairākās vietās, tad kuģa kapteinis var izvēlēties, kurā no tām pietauvoties. Kādā brīdī var izveidoties nepatīkama situācija, ka kādam no kuģiem nav kur pietauvoties. Piemēram, ja pietātnes garums ir pieci posmi un kuģis ar garumu 1 ir pietauvojoties pirmajā posmā, bet kuģis ar garumu divi trešajā un ceturtajā posmā, tad kuģis ar garumu 2 pietauvoties vairs nevar (skat.zīm.).

Ja kāds kuģis nevar pietauvoties, tad tas draud ar starptautiska līmeņa nepatīkšanām, un regates organizētāji gribētu par šādu iespēju zināt pēc iespējas agrāk, pie viena uzzinot arī tā pirmā kuģa numuru pēc kārtas, kuram var būt problēmas ar pietauvošanos.

Skaidrs, ka nepatīkšanu nebūtu, ja visi kapteiņi savus kuģus pietauvotu pirmajā brīvajā posmā pēc kārtas. Bet, ko nu par to vairs runāt...

Uzrakstiet programmu, kas nosaka, vai draud starptautiska līmeņa nepatīkšanas!



Ievaddati

Teksta faila *regate.dat* pirmajā rindā dotas divu naturālu skaitļu K (kuģu skaits, $K \leq 500000$) un N (pietātnes posmu skaits, $N \leq 10^9$) vērtības, kas atdalīti ar tukšumsimbolu.

Katrā no nākošajām K rindām dots naturāls skaitlis - viena kuģa garums. Faila $i+1$ -ajā ($1 \leq i \leq K$) rindā dots i -tā kuģa garums. Kuģi doti tādā secībā, kādā tie ienāk ostā.

Izvaddati

Teksta faila *regate.rez* vienīgajā rindā jāizvada vesels skaitlis - tā pirmā kuģa numurs pēc kārtas, kuram ienākot ostā var sākties nepatīkšanas. Ja nepatīkšanas nedraud, failā jāizvada 0.

Piemēri

| Ievaddati (fails <i>regate.dat</i>) | Izvaddati (fails <i>regate.rez</i>) |
|--------------------------------------|--------------------------------------|
| 3 5 1 2 2 | 3 |

| Ievaddati (fails <i>regate.dat</i>) | Izvaddati (fails <i>regate.rez</i>) |
|--------------------------------------|--------------------------------------|
| 3 5 2 2 1 | 0 |

*) Katra testa izpildei tiks dota viena sekunde.

Katram uzdevumam būs 10 testi. Par pilnīgi pareizu laikā izpildītu testu tiks piešķirti 5 punkti.