

LATVIJAS 21. INFORMĀTIKAS OLIMPIĀDES
IV POSMA UZDEVUMU APSKATS
Otrā diena (2008. gada 4. aprīlis)



Uzdevuma nosaukums:	TRĪS TARAKĀNI	GALDS	KAKTUSA GRAFS
Ievaddatu faila nosaukums:	tristar.dat	galds.dat	kaktuss.dat
Izvaddatu faila nosaukums:	tristar.rez	galds.rez	kaktuss.rez
Izpildes laika ierobežojums vienam testpiemēram (laiks tiek mērīts uz testēšanas servera):	0,1 sekunde	0,2 sekundes	2 sekundes
Atmiņas ierobežojums:	64MB	64MB	64MB
Maksimāli iespējamais punktu skaits par uzdevumu:	100	100	100
Nosacījums, lai testēšanas serveris atzītu programmu par derīgu testēšanai:	Programmai jākompilējas bez kļūdām un jāizdod pareizs rezultāts uzdevuma formulējumā minētajiem piemēriem.	Programmai jākompilējas bez kļūdām un jāizdod pareizs rezultāts uzdevuma formulējumā minētajiem piemēriem.	Programmai jākompilējas bez kļūdām un jāizdod pareizs rezultāts uzdevuma formulējumā minētajiem piemēriem.

Ievaddatu un izvaddatu failus norādiet **bez** pilnā ceļa (uzskatiet, ka tie atrodas tekošajā katalogā) un tieši tā, kā norādīts uzdevuma formulējumā (**ar mazajiem burtiem**)!

Kompilējot programmas uz servera, tiks lietoti šādi kompilatori:

Valodai PASCAL:

- FreePascal (versija 2.2.0) ar parametriem `-O2 -XS`

Valodai C:

- GNU C (versija 3.4.2) ar parametriem `-std=c99 -O2 -s -static -lm`
- Microsoft Visual C 2005 ar parametriem `/TC /O2`

Valodai C++:

- GNU C++ (versija 3.4.2) ar parametriem `-O2 -s -static`
- Microsoft Visual C++ 2005 ar parametriem `/TP /O2`

Katra uzdevuma pēdējo akceptēto programmas kodu pēc nosūtīšanas uz serveri saglabāriet arī darba datorā un neizdzēsiet pēc sacensību beigām!

Faila operāciju (datu ielasīšanas un izvadīšanas) ātrdarbības testēšanai uz servera var izmantot šim mērķim speciāli izveidoto uzdevumu “**Ātrdarbības tests**”. Vienīgā piemēra ievaddatu fails `tests.dat` satur 10^6 rindu, i -tajā faila rindā dots skaitlis i ($1 \leq i \leq 10^6$). Atbilde tiek uzskatīta par pareizu, ja iesūtītā programma izveido identisku izvaddatu failu `tests.rez`. Laika limits testa izpildei – 5 sekundes.

LATVIJAS 21. INFORMĀTIKAS OLIMPIĀDES
IV POSMA UZDEVUMI
Otrā diena (2008. gada 4. aprīlis)



1. “TRĪS TARAKĀNI”

Gara gaiteņa vienā galā atrodas trīs tarakāni. Pirmais tarakāns līdz gaiteņa otram galam var aizskriet T_1 sekundēs, otrais – T_2 sekundēs, trešais – T_3 sekundēs. Tarakāni sāk skriet vienlaicīgi un bez apstāšanās skrien līdz gaiteņa otram galam. Sasnieguši gaiteņa galu, tarakāni momentāli apgriežas un skrien pa gaiteni atpakaļ. Šādi tie skraida pa gaiteni turp un atpakaļ līdz brīdim, kad tie visi vienlaicīgi nonāk kādā gaiteņa punktā un apstājas.

Uzrakstiet programmu, kas dotām T_1 , T_2 un T_3 vērtībām nosaka, cik sekundes tikšanās brīdī būs pagājušas kopš skrējiena sākuma!

Ievaddati

Teksta faila `tristar.dat` vienīgajā rindā dotas trīs naturālu skaitļu T_1 , T_2 un T_3 vērtības. Blakusesoši skaitļi atdalīti ar tukšumzīmi. Zināms, ka neviens no skaitļiem nepārsniedz 1 000 000 un starp skaitļiem T_1 , T_2 un T_3 var atrast vismaz divus atšķirīgus.

Izvaddati

Laika posma garums līdz satikšanās brīdim jāaprēķina kā nesaīsināma daļa P/Q (P un Q – naturāli skaitļi). Teksta faila `tristar rez` vienīgajā rindā jāizvada skaitļi P un Q , kas atdalīti ar tukšumzīmi.

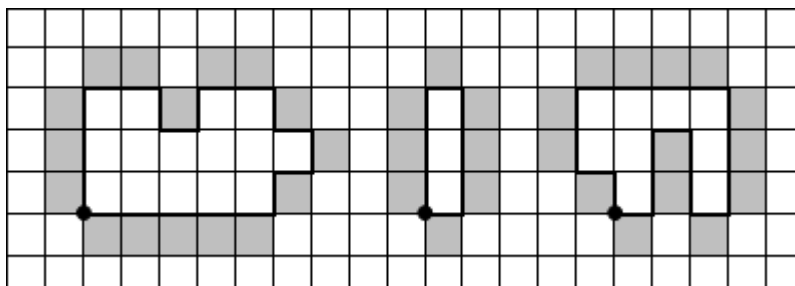
Piemēri

Ievaddati (fails <code>tristar.dat</code>)	Izvaddati (fails <code>tristar rez</code>)
1 1 2	4 3
4 1 2	8 3

2. “GALDS”

Gatavojoties Lielajai konferencei, pasākuma rīkotāji ir sagatavojuši lielu galdu. Lai izplānotu dalībnieku izvietojumu pie galda, uz rūtiņu lapas ir izveidots galda plāns, kurā galds attēlots kā daudzstūris, kura malas iet pa rūtiņu līnijām. Izmantojot šo plānu, nepieciešams noteikt, cik dalībnieku iespējams izvietot apkārt galdam. Zināms, ka katrs dalībnieks jānosēdina pie galda kontūra posma, kam plānā atbilst viena rūtiņas mala, un pie šī posma jābūt vietai, kur novietot krēslu, kam plānā atbilst viena rūtiņa. Attēlā redzamajām galdu konfigurācijām attēlots maksimāli iespējamais dalībnieku skaits (pelēkā krāsā iekrāsotas krēsliem atbilstošās rūtiņas).

Uzrakstiet programmu, kas dotai galda konfigurācijai aprēķina lielāko iespējamo dalībnieku skaitu, ko iespējams nosēdināt ap to!



Ievaddati

Galds tiek uzdots kā tā perpendikulāro malu secīga virkne, aprakstu sākot ar horizontālo malu no tā galda stūra, kurš atrodas uz galda apakšējās malas un starp visiem tiem – visvairāk pa kreisi (zīmējumos atzīmēts ar aplīti).

Teksta faila `galds.dat` pirmajā rindā dots galda malu skaits N ($4 \leq N \leq 10^5$).

Faila otrajā rindā doti N veseli skaitļi, kas atšķirīgi no 0, – malu garumi, apstaigājot daudzstūri no sākotnējā stūra. Pozitīvs skaitlis nozīmē, ka kārtējā mala ir vērsta uz augšu vai pa labi, bet negatīvs – uz leju vai pa kreisi. Skaitļa modulis norāda attiecīgās malas garumu. Zināms, ka galds ir korekts – tā malas nekrustojas un jebkurām divām vienādi vērstām (horizontālām vai vertikālām) malām nav neviena kopīga punkta.

Izvaddati

Teksta faila `galds.rez` vienīgajā rindā jāizvada naturāls skaitlis – lielākais iespējamais dalībnieku, ko iespējams nosēdināt apkārt galdam, skaits. Zināms, ka tas ir mazāks par 2^{31} .

Piemēri (atbilst zīmējumiem)

Ievaddati (fails <code>galds.dat</code>)	Izvaddati (fails <code>galds.rez</code>)
12 5 1 1 1 -1 1 -2 -1 -1 1 -2 -3	16
4 1 3 -1 -3	8
10 1 2 1 -2 1 3 -4 -2 1 -1	14

3. “KAKTUSA GRAFS”

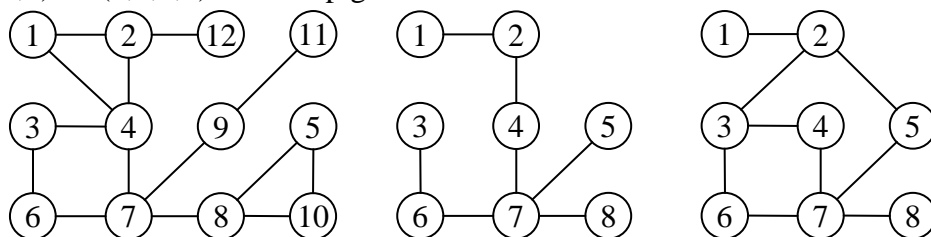
Dots grafs, kas sastāv no N virsotnēm un M šķautnēm. Grafa virsotnes ir sanumurētas pēc kārtas ar naturāliem skaitļiem no 1 līdz N . Katra šķautne savieno divas atšķirīgas grafa virsotnes. Jebkuras divas virsotnes grafā savieno ne vairāk kā viena šķautne.

Grafu sauc par *sakarīgu*, ja no jebkuras tā virsotnes iespējams aiziet līdz jebkurai citai tā virsotnei, ejot pa grafa šķautnēm.

Par *vienkāršu ciklu* grafā sauc atšķirīgu virsotņu virkni (v_1, v_2, \dots, v_k) , kur $k > 2$, ka katram i ($1 \leq i < k$) virsotnes v_i un v_{i+1} ir savienotas ar šķautni, kā arī virsotnes v_1 un v_k ir savienotas ar šķautni. Uzskatīsim, ka divi cikli ir vienādi, ja ir vienādas šķautņu kopas, kas veido šos ciklus.

Grafu sauc par *kaktusa grafu*, ja tas ir sakarīgs un jebkuriem diviem tā atšķirīgiem vienkāršiem cikliem ir ne vairāk kā viena kopīga virsotne.

Piemēram, attēlā pa kreisi redzams kaktusa grafs ar trīs vienkāršiem cikliem (1,2,4), (3,4,7,6) un (8,5,10). Vidējā attēlā redzams kaktusa grafs bez vienkāršiem cikliem. Attēlā pa labi redzamais grafs nav kaktusa grafs, jo tā vienkāršiem cikliem (3,2,5,7,4) un (6,7,4,3) ir trīs kopīgas virsotnes.



Uzrakstiet programmu, kas dotam sakarīgam grafam noskaidro, vai tas ir kaktusa grafs!

Ievaddati

Teksta faila `kaktuss.dat` pirmajā rindā doti veseli skaitļi N un M ($1 \leq N \leq 100\,000$, $0 \leq M \leq 300\,000$).

Nākamajās M faila rindās dots grafa šķautņu apraksts. Katrā rindā dots vienas šķautnes apraksts – divi atšķirīgi naturāli skaitļi, kas nepārsniedz N , – virsotņu numuri, ko savieno šķautne.

Izvaddati

Ja dotais grafs ir kaktusa grafs, teksta faila `kaktuss.rez` pirmajā rindā jāizvada dotā grafa vienkāršo ciklu skaits. Nākamajās faila rindās jāizvada visi dotā grafa vienkāršie cikli patvaļīgā secībā, pa vienam ciklam rindā.

Ja dotais grafs nav kaktusa grafs, faila pirmajā rindā jāizvada vārds “NAV”, bet nākamajās divās rindās – jebkuri divi tā atšķirīgi vienkārši cikli, kuriem ir vairāk par vienu kopīgu virsotni.

Informācija par katru ciklu jāizvada jaunā rindā, izvadot cikla virsotņu virknes numurus. Diviem blakusesošiem numuriem jābūt atdalītiem ar tukšumzīmi.

Piemēri (atbilst zīmējumiem)

Ievaddati (fails <code>kaktuss.dat</code>)	Izvaddati (fails <code>kaktuss.rez</code>)
<pre> 12 14 1 2 2 4 4 1 7 6 7 4 7 9 7 8 3 4 3 6 9 11 10 8 5 10 5 8 2 12 </pre>	<pre> 3 1 2 4 3 4 7 6 8 5 10 </pre>
<pre> 8 7 7 4 4 2 2 1 7 6 6 3 5 7 8 7 </pre>	<pre> 0 </pre>
<pre> 8 9 1 2 2 5 5 7 7 6 6 3 3 4 4 7 8 7 3 2 </pre>	<pre> NAV 3 2 5 7 4 6 7 4 3 </pre>