

**LATVIJAS 18. INFORMĀTIKAS OLIMPIĀDES III POSMA
UZDEVUMU APSKATS
Pirmā diena (2005.gada 14.marts)**



Uzdevuma nosaukums:	ŠŪNAS	SKUDRAS	ARITMĒTISKĀ PROGRESIJA
Ievaddatu faila nosaukums:	sunas.dat	skudras.dat	progr.dat
Izvaddatu faila nosaukums:	sunas.rez	skudras.rez	progr.rez
Izpildes laika ierobežojums vienam testam (laiks tiek mērīts uz testēšanas servera):	1 sekunde	2 sekundes	3 sekundes
Maksimāli iespējamais punktu skaits par uzdevumu:	100	100	100
Komentārs, kas jāiekļauj programmas koda sākumā (pirmajās četrās rindās), ja programma rakstīta valodā PASCAL :	{ task: sunas lang: pascal }	{ task: skudras lang: pascal }	{ task: progr lang: pascal }
Komentārs, kas jāiekļauj programmas koda sākumā (pirmajās četrās rindās), ja programma rakstīta valodā C :	/* task: sunas lang: c */	/* task: skudras lang: c */	/* task: progr lang: c */
Komentārs, kas jāiekļauj programmas koda sākumā (pirmajās četrās rindās), ja programma rakstīta valodā C++ :	/* task: sunas lang: c++ */	/* task: skudras lang: c++ */	/* task: progr lang: c++ */
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

Datu un rezultātu failus norādiet **bez** pilnā ceļa (uzskatiet, ka datu un rezultātu faili atrodas tekošajā katalogā) un tieši tā, kā norādīts uzdevuma formulējumā (**ar mazajiem burtiem**)!

C vai C++ valodā rakstītajām programmām jābeidzas ar atgriešanās kodu 0.

Jautājumus par uzdevumu formulējumiem centieties uzdot sacensību pirmās stundas laikā!

Katra uzdevuma labāko programmas kodu pēc nosūtīšanas uz serveri saglabājiet arī darba datorā un neizdzēsiet pēc sacensību beigām!

LATVIJAS 18. INFORMĀTIKAS OLIMPIĀDES
III POSMA UZDEVUMI
Pirmā diena (2005.gada 14.marts)



1. "ŠŪNAS"

Visa plakne sadalīta regulāros sešstūros, veidojot "bišu šūnu" rakstu. Šūnas ir sanumurētas pēc kārtas pa spirāli pulksteņrādītāja virzienā. Plaknes numurēšanas sākuma fragments redzams zīmējumā.

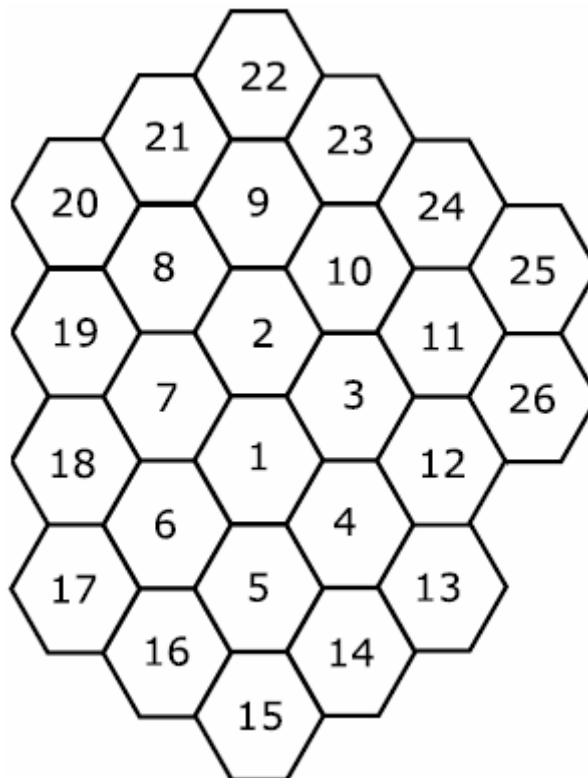
Uzrakstiet programmu, kas ievadītam šūnas numuram N nosaka, kādi ir to sešu šūnu, kurai ar doto ir kopīga mala, numuri!

Ievaddati

Teksta faila `sunas.dat` pirmajā rindā dota naturāla skaitļa N ($1 \leq N \leq 2 \cdot 10^9$) vērtība.

Izvaddati

Teksta faila `sunas.rez` vienīgajā rindā augošā secībā jāizvada seši naturāli skaitļi - dotās šūnas kaimiņu šūnu numuri. Starp katriem diviem blakus skaitļiem jābūt vienam tukšumsimbolam.



Piemēri

Ievaddati (fails <code>sunas.dat</code>)	Izvaddati (fails <code>sunas.rez</code>)
2	1 3 7 8 9 10

Ievaddati (fails <code>sunas.dat</code>)	Izvaddati (fails <code>sunas.rez</code>)
27	12 13 26 28 47 48

2. "SKUDRAS"

Dziļā meža skudras pārvietojas garās kolonnās viena aiz otras. Ilgu gadu gaitā mazie meža pētnieki ir iemācījušies atšķirt dažāda veida skudras un piešķirušī katram skudru veidam savu numuru - naturālu skaitli.



Jaunākie pētījumi rāda, ka katru skudru kolonnu var sadalīt vienā vai vairākās skudru vienībās. Ir zināms, ka:

- katra skudra ietilpst kādā vienībā,
- vienā vienībā ietilpst tikai viena veida skudras,
- ja starp divām viena veida skudrām kolonnā ir ne vairāk kā $K-1$ (K - naturāls skaitlis, kas dažādām kolonnām var atšķirties) cita veida skudras, tad abas skudras ietilpst vienā vienībā,
- ja starp divām viena veida skudrām kolonnā pēc kārtas ir vismaz K cita veida skudras, tad šīs divas skudras ir dažādās vienībās.

Piemēram, ja $K=3$, tad zīmējumā redzamajā kolonnā visas pirmā veida skudras ir vienā vienībā, bet trešā veida skudras katra ir savā vienībā.

Mazos meža pētniekus katrā skudru kolonnā visvairāk interesē lielākais skudru skaits vienā vienībā, bet, tā kā skudru vienā kolonnā var būt ļoti daudz, tad šis skaitīšanas darbs ir gauži nogurdinošs.

Uzrakstiet programmu, kas dotai skudru kolonnai un attālumam K nosaka lielākās skudru vienības izmēru!

Ievaddati

Ievaddatu faila `skudras.dat` vienīgajā rindā ir ar tukšumsimbolu atdalīti skaitļi, no kuriem pirmais ir naturāls skaitlis K ($1 \leq K \leq 100$), otrais - naturāls skaitlis N ($1 \leq N \leq 10^6$), kas apzīmē skudru kolonnas garumu, un pēc tiem seko N naturāli skaitļi, ne lielāki par 10^6 , un apzīmē attiecīgās skudras veidu. Skudru veidu numuri ir doti tieši tādā secībā, kādā skudras atrodas kolonnā.

Izvaddati

Izvaddatu failam `skudras.rez` jāsaturs viens vienīgs naturāls skaitlis – lielākās skudru vienības izmērs.

Piemēri

Ievaddati (fails <code>skudras.dat</code>)	Izvaddati (fails <code>skudras.rez</code>)
2 6 10 10 2 2 10 10	2
Ievaddati (fails <code>skudras.dat</code>)	Izvaddati (fails <code>skudras.rez</code>)
1 6 1 1 2 1 1 1	3
Ievaddati (fails <code>skudras.dat</code>)	Izvaddati (fails <code>skudras.rez</code>)
2 6 1 1 2 1 1 1	5

3. "ARITMĒTISKĀ PROGRESIJA"

Kādu dienu Edgars uz tāfeles uzrakstīja kādas aritmētiskās progresijas locekļus (vismaz 2) pēc kārtas vienā rindā bez atstarpēm vai kādiem atdalošiem simboliem. Zināms, ka visi uzrakstītie progresijas locekļi bija nenegatīvi.

Kaspars pēc šīs virknes gribētu noteikt aritmētiskās progresijas pirmo locekli un diferenci tā, lai tā būtu mazākā iespējamā difference (pēc absolūtās vērtības).

Uzrakstiet programmu, kas šo darbu veic Kaspara vietā!

Ja vienāda lieluma (pēc absolūtās vērtības) diferenci var iegūt vairākos veidos, programmai jāatrod tas variants, kurā progresijas pirmais loceklis ir mazākais iespējamais.

Ievaddati

Teksta faila `progr.dat` pirmajā rindā dota naturāla skaitļa N ($1 < N < 5000$) vērtība - ciparu skaits Edgara uzrakstītajā virknē. Faila otrajā rindā doti N cipari - Edgara uzrakstītā ciparu virkne.

Izvaddati

Teksta faila `progr.rez` vienīgajā rindā izvadīt divus ar tukšumsimbolu atdalītus veselus skaitļus – aritmētiskās progresijas pirmo locekli un diferenci, kas atbilst iepriekšaprakstītajiem nosacījumiem.

Piemēri

Ievaddati (fails <code>progr.dat</code>)	Izvaddati (fails <code>progr.rez</code>)
7 1357911	1 2

Ievaddati (fails <code>progr.dat</code>)	Izvaddati (fails <code>progr.rez</code>)
4 1111	1 0

Ievaddati (fails <code>progr.dat</code>)	Izvaddati (fails <code>progr.rez</code>)
6 345234	345 -111

Ievaddati (fails <code>progr.dat</code>)	Izvaddati (fails <code>progr.rez</code>)
3 101	10 -9

**LATVIJAS 18. INFORMĀTIKAS OLIMPIĀDES III POSMA
UZDEVUMU APSKATS
Otrā diena (2005.gada 15.marts)**



Uzdevuma nosaukums:	RINDĀ PĒC SALDĒJUMA	LODĪTES
Ievaddatu faila nosaukums:	sald.dat	lodites.dat
Izvaddatu faila nosaukums:	sald.rez	lodites.rez
Izpildes laika ierobežojums vienam testam (laiks tiek mērīts uz testēšanas servera):	1 sekunde	1 sekunde
Maksimāli iespējamais punktu skaits par uzdevumu:	100	100
Komentārs, kas jāiekļauj programmas koda sākumā (pirmajās četrās rindās), ja programma rakstīta valodā PASCAL :	{ task: sald lang: pascal }	{ task: lodites lang: pascal }
Komentārs, kas jāiekļauj programmas koda sākumā (pirmajās četrās rindās), ja programma rakstīta valodā C :	/* task: sald lang: c */	/* task: lodites lang: c */
Komentārs, kas jāiekļauj programmas koda sākumā (pirmajās četrās rindās), ja programma rakstīta valodā C++ :	/* task: sald lang: c++ */	/* task: lodites lang: c++ */
Nosacījums, lai testēšanas serveris atzītu programmu par derīgu testēšanai:	Programmai jākompilējas bez kļūdām un jāizdod pareizs rezultāts uzdevuma formulējumā minētajiem pirmajiem diviem piemēriem	Programmai jākompilējas bez kļūdām un jāizdod pareizs rezultāts uzdevuma formulējumā minētajiem pirmajiem trim piemēriem

Datu un rezultātu failus norādiet **bez** pilnā ceļa (uzskatiet, ka datu un rezultātu faili atrodas tekošajā katalogā) un tieši tā kā norādīts uzdevuma formulējumā (**ar mazajiem burtiem**)!

C vai C++ valodā rakstītajām programmām jābeidzas ar atgriešanās kodu 0.

Jautājumus par uzdevumu formulējumiem centieties uzdot sacensību pirmās stundas laikā!

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

LATVIJAS 18. INFORMĀTIKAS OLIMPIĀDES
III POSMA UZDEVUMI
Otrā diena (2005.gada 15.marts)



1. "RINDĀ PĒC SALDĒJUMA"

Rindā pie saldējuma kioska stāv N skolēni. Saldējums maksā vienu eiru (1 €). Nevienam no skolēniem sākumā naudas nav, savukārt skolotājam nauda ir un tā visa ir vienas un divu eiru monētās. Skolotājas uzdevums ir izdalīt daļu no šīm monētām tā, lai

- katram skolēnam tiktu iedota ne vairāk kā viena monēta;
- kopā izdalītās naudas daudzums būtu tieši N eiras;
- visi skolēni varētu nopirkt sev saldējumu.

Saldējuma pārdevējam sākotnēji nav nevienas monētas, un atlikumu no 2 € monētas viņš var izdot tikai tad, ja kasē ir 1 € monēta. Ja pārdevējam ir izbeigušās naudas monētas, ko izdot skolēnam, kurš vēlas nopirkt saldējumu, dodot 2 € monētu, viņš kļūst nikns un kiosku aiztaisa ciet, kā rezultātā atlikušie skolēni netiek pie saldējuma. Kiosks tiek aiztaisīts arī tad, ja skolēns vēlas nopirkt saldējumu, bet viņam nav monētas. Skolēnam, kuram ir izdots atlikums, šī 1 € monēta jāatdod pirmajam rindā stāvošajam, kuram tobrīd vēl nav monētas.

Uzrakstiet programmu, kas dotai N vērtībai aprēķina un izvada, cik dažādos veidos skolotāja var izdalīt naudu tā, lai visi skolēni aprakstītajā situācijā varētu iegādāties saldējumu!

Ievaddati

Teksta faila `sald.dat` vienīgajā rindā ir dots naturāls skaitlis $N(N \leq 46)$.

Izvaddati

Teksta faila `sald.rez` pirmajā rindā jāizvada viens vesels skaitlis - dažādo naudas izdalīšanas veidu skaits.

Piemēri.

Ievaddati (fails <code>sald.dat</code>)	Izvaddati (fails <code>sald.rez</code>)
2	1

Piezīme:

Abi skolēni saldējumu varēs nopirkt tikai tad, ja katram no viņiem būs iedota 1 € monēta.

Ievaddati (fails <code>sald.dat</code>)	Izvaddati (fails <code>sald.rez</code>)
5	8

Piezīme:

Skolotāja rindā stāvošajiem skolēniem naudu var izdalīt šādos astoņos veidos ("-" nozīmē, ka šim skolēnam sākumā nauda nav iedota):

- | | |
|----|----|
| 1) | 5) |
| 2) | 6) |
| 3) | 7) |
| 4) | 8) |

Ievaddati (fails <code>sald.dat</code>)	Izvaddati (fails <code>sald.rez</code>)
45	1933803956425574035

2. "LODĪTES"

Uz diviem riņķiem izvietotas baltas un melnas lodītes. Uz katra riņķa atrodas Q lodītes. Melno un balto lodīšu kopskaits ir vienāds. Abi riņķi ir novietoti tā, ka divas lodītes atrodas gan uz viena, gan otra riņķa. Katra riņķa iekšpusē atrodas W otra riņķa lodītes. Vienā gājienā atļauts pārbīdīt visas uz viena riņķa esošās lodītes par vienu pozīciju pulksteņrādītāja kustības virzienā vai pret to.

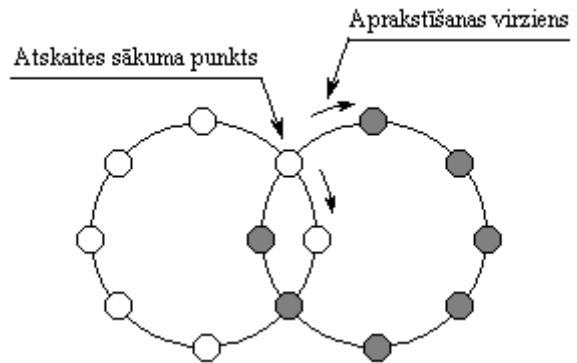
Uzrakstiet programmu, kas nosaka mazāko gājienu skaitu, ar kādu iespējams sakārtot lodītes tā, ka uz katra no riņķiem visas lodītes, izņemot vienu, ir vienā krāsā! Ievērojiet, ka šādi sakārtojumi pavisam ir četri!

Ievaddati

Teksta faila `lodites.dat` pirmajā rindā ir doti divi naturāli skaitļi $4 \leq Q \leq 12$ un $0 < W < Q/2$, kas atdalīti ar tukšuma simbolu.

Faila otrajā rindā dots kreisā riņķa apraksts kā Q burtus gara simbolu virkne. Katrs simbols virknē apzīmē vienas lodītes krāsu un var būt tikai "M" (apzīmē melnu lodīti) vai "B" (apzīmē balto lodīti). Lodīšu izvietojuma apraksts sākas ar "augšējo" abu riņķu kopīgo lodīti un turpināts pulksteņrādītāja virzienā.

Faila trešajā rindā dots labā riņķa apraksts analogiskā formā.



Izvaddati

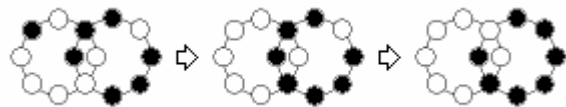
Teksta faila `lodites rez` vienīgajā rindā ir jāizvada vesels skaitlis - mazākais nepieciešamais gājienu skaits. Ja minētā veida lodīšu sakārtojumu iegūt nav iespējams, faila vienīgajā rindā jāizvada skaitlis -1.

Piemēri

Ievaddati (fails <code>lodites.dat</code>)	Izvaddati (fails <code>lodites rez</code>)
8 1 MBBBBBMB MMBMMBM	4

Piezīme

Pareizo sakārtojumu var iegūt, pagriežot kreiso riņķi divas reizes pulksteņrādītāja kustības virzienā, un pēc tam labo riņķi divas reizes pretēji pulksteņrādītāja virzienam (skat.zīmējumu).



Ievaddati (fails <code>lodites.dat</code>)	Izvaddati (fails <code>lodites rez</code>)
4 1 MMBB MBBM	-1

Ievaddati (fails <code>lodites.dat</code>)	Izvaddati (fails <code>lodites rez</code>)
5 2 MMMMBM MBBBB	0

Ievaddati (fails <code>lodites.dat</code>)	Izvaddati (fails <code>lodites rez</code>)
12 4 BBMMMMMBBBB BBMBMMMMBMB	19