LATVIJAS 22. INFORMĀTIKAS OLIMPIĀDES III POSMA UZDEVUMU APSKATS JAUNĀKAJAI (8.-10. KLAŠU) GRUPAI Pirmā diena (2009. gada 11. marts)

Uzdevuma nosaukums:	**LOKOMOTĪVES**	**SLIEŽU CEĻI**	**MONĒTU STABIŅI**
Ievaddatu datnes nosaukums:	`loko.dat`	`celi.dat`	`stabini.dat`
Izvaddatu datnes nosaukums:	`loko.rez`	`celi.rez`	`stabini.rez`
Izpildes laika ierobežojums vienam testpiemēram (laiks tiek mērīts uz testēšanas servera):	1 sekunde	1 sekunde	0,2 sekundes
Atmiņas ierobežojums:	64MB	64MB	64MB
Maksimāli iespējamais punktu skaits par uzdevumu:	100	100	100

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

Lai iesūtītais risinājums tiktu pieņemts tālākai testēšanai, tam pareizi jāstrādā uz **visiem** uzdevuma formulējumā dotajiem testpiemēriem. Testēšanas serverī noklikšķinot uz iesūtījuma, parādās rezultāts katram testpiemēram tādā pašā secībā, kā tie doti uzdevuma formulējumā.

Kompilējot programmas uz servera, tiks lietoti šādi kompilatori:

Valodai PASCAL:
- FreePascal (versija 2.2.0) ar parametriem `–O2 –Sg`

Valodai C:
- GNU C (versija 3.4.2) ar parametriem `-std=c99 -O2 -s -static –lm`
- Microsoft Visual C 2008 ar parametriem `/TC /O2`

Valodai C++:
- GNU C++ (versija 3.4.2) ar parametriem `-O2 -s -static`
- Microsoft Visual C++ 2008 ar parametriem `/TP /O2`

Programmas tiks testētas uz datora ar *Intel® Core™ 2 2* GHz procesoru.

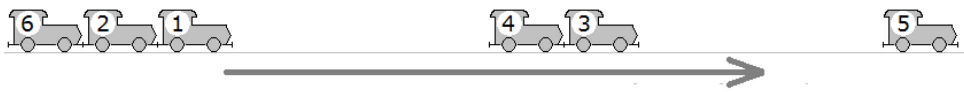
LATVIJAS 22. INFORMĀTIKAS OLIMPIĀDES
III POSMA UZDEVUMI
JAUNĀKAJAI (8.-10. KLAŠU) GRUPAI
Pirmā diena (2009. gada 11. marts)

1. "LOKOMOTĪVES"

Uz bezgalīgi gara taisna sliežu ceļa vienādos attālumos viena no otras atrodas N lokomotīves. Katrai lokomotīvei ir unikāls numurs – naturāls skaitlis no 1 līdz N. Lokomotīves numurs norāda tās maksimālo kustības ātrumu kilometros stundā. Visas lokomotīves vienlaicīgi sāk kustēties uz priekšu (zīmējumā – pa labi) ar maksimālo iespējamo ātrumu. Ja kāda lokomotīve panāk priekšā braucošu lēnāku lokomotīvi, tā samazina kustības ātrumu un turpina braukt ar priekšā braucošās lokomotīves ātrumu. Pēc kāda laika visas lokomotīves ir sadalījušās grupās, kas turpmāk vairs nemainās.

Piemēram, sākumā sešas lokomotīves var būt izvietotas šādi:

Tad pēc kāda laika pēc kustības uzsākšanas tās sadalīsies trīs grupās, kas katra pārvietojas ar vienu noteiktu ātrumu:



Ievērojiet, ka grupā var būt arī tikai viena lokomotīve!

Uzrakstiet programmu, kas dotam sākotnējam lokomotīvju novietojumam nosaka, cik pastāvīgās grupās tās sadalīsies!

Ievaddati

Teksta datnes `loko.dat` pirmajā rindā dots naturāls skaitlis N (lokomotīvju skaits, N ≤ 1 000 000). Otrajā datnes rindā doti N dažādi naturāli skaitļi robežās no 1 līdz N – lokomotīvju numuri tādā secībā, kādā tās atrodas uz sliežu ceļa, sākot ar braukšanas virzienā pēdējo. Starp katriem diviem blakusesošiem skaitļiem ir viena tukšumzīme.

Izvaddati

Teksta datnes `loko.rez` vienīgajā rindā jāizvada viens naturāls skaitlis – grupu skaits, kādā sadalīsies lokomotīves.

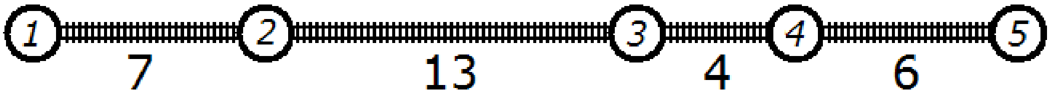
Piemēri

Ievaddati	**Izvaddati**	**Piezīmes**
`6` `6 2 1 4 3 5`	`3`	Piemērs atbilst dotajam zīmējumam.
`6` `1 2 3 4 6 5`	`5`	

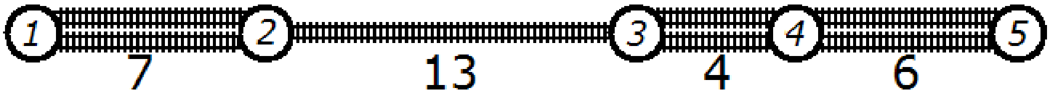
2. “SLIEŽU CEĻI”

Dzelzceļa līnijā ir N stacijas, kas sanumurētas ar naturāliem skaitļiem no 1 līdz N pēc kārtas. Ir zināms attālums kilometros starp katrām divām blakusesošām stacijām. Pagaidām dzelzceļa līnijā ir tikai viens sliežu ceļš, bet ir nolemts sākt būvēt dublējošos sliežu ceļus tā, lai no katras stacijas vismaz uz vienu no kaimiņu stacijām būtu divi paralēli sliežu ceļi un lai dublējošo sliežu ceļu kopgarums būtu pēc iespējas mazāks.

Piemēram, dzelzceļa līnijā var būt piecas stacijas un attālums starp blakusesošām stacijām tāds, kā redzams zīmējumā:



Tad dublējošie sliežu ceļi jābūvē starp 1. un 2., 3. un 4., 4. un 5. staciju:



Šajā gadījumā dublējošo sliežu ceļu kopgarums ir 17 kilometri.

Uzrakstiet programmu, kas dotam staciju skaitam un attālumam starp blakusesošām stacijām aprēķina īsāko iespējamo dublējošo sliežu ceļu kopgarumu!

Ievaddati

Teksta datnes `celi.dat` pirmajā rindā dots naturāls skaitlis N ($1 < N \le 1\,000\,000$). Katrā no nākamajām N−1 datnes rindām dots viens naturāls skaitlis, kura vērtība nepārsniedz 1000, – attālums starp divām blakusesošām stacijām. Katram i ($2 \le i \le N$) attālums starp (i−1)-o un i-to staciju ir dots datnes i-tajā rindā.

Izvaddati

Teksta datnes `celi.rez` vienīgajā rindā jāizvada viens naturāls skaitlis – īsākā iespējamā dublējošā sliežu ceļa kopgarums.

Piemēri

Ievaddati	Izvaddati	Piezīmes
5 7 13 4 6	17	Atbilst uzdevuma formulējumā dotajam piemēram.
4 7 1 13	20	

3. “MONĒTU STABIŅI”

Aplī ir izvietoti N monētu stabiņi. Katrā stabiņā ir vismaz viena monēta. Vairāku monētu gadījumā tās ir uzliktas viena virs otras. Stabiņi ir sanumurēti ar naturāliem skaitļiem no 1 līdz N pulksteņrādītāja virzienā pēc kārtas. Pēc N-tā stabiņa pulksteņrādītāja virzienā seko pirmais stabiņš.

Monētu skaits pa visiem stabiņiem kopā ir N*K. Ar stabiņiem atļauts veikt tikai viena veida darbību – “līdzināšanu”. Tas nozīmē, ka tiek izvēlēts kāds stabiņš un, ja tajā ir vairāk par K monētām, tad šajā stabiņā tiek atstātas tieši K monētas un atlikušās tiek pārceltas uz blakusesošu stabiņu pulksteņrādītāja virzienā. Ja monētu skaits stabiņā nepārsniedz K, tad neko darīt nevar (bet šis tomēr tiek ieskaitīts kā izdarīts gājiens). Tālāk šāda pat operācija tiek atkārtota ar nākamo stabiņu pulksteņrādītāja virzienā, pēc tam ar vēl nākamo utt. Pēc kāda laika monētu skaits visos stabiņos kļūs vienāds ar K, un tikko tas ir noticis, tā process apstājas.

Sākot līdzināšanu no dažādiem stabiņiem, kopējais darbību skaits var būt atšķirīgs. Piemēram, ja N=5, K=4 un sākumā monētu skaits stabiņos ir 3, 1, 6, 4, 6, tad, sākot līdzināšanu ar ceturto stabiņu, monētu skaits stabiņos mainīsies šādi:

Gājiena numurs	Monētu skaits stabiņos		Piezīmes
	pirms gājiena	pēc gājiena	
1	3 1 6 **4** 6	3 1 6 **4** 6	4=4, nekas nemainās
2	3 1 6 4 **6**	5 1 6 4 **4**	6>4, divas monētas ir pārceltas uz pirmo stabiņu
3	**5** 1 6 4 4	**4** 2 6 4 4	5>4, viena monēta ir pārcelta uz otro stabiņu
4	4 **2** 6 4 4	4 **2** 6 4 4	2<4, nekas nemainās
5	4 2 **6** 4 4	4 2 **4** 6 4	6>4, divas monētas ir pārceltas uz ceturto stabiņu
6	4 2 4 **6** 4	4 2 4 **4** 6	6>4, divas monētas ir pārceltas uz piekto stabiņu
7	4 2 4 4 **6**	6 2 4 4 **4**	6>4, divas monētas ir pārceltas uz pirmo stabiņu
8	**6** 2 4 4 4	**4** 4 4 4 4	6>4, divas monētas ir pārceltas uz otro stabiņu

Šajā gadījumā bija nepieciešami astoņi gājieni, bet, ja līdzināšana būtu sākta ar trešo stabiņu, būtu pieticis ar četriem gājieniem.

Uzrakstiet programmu, kas dotām N un K vērtībām un monētu skaitam katrā stabiņā, atrod to stabiņu, no kura jāsāk līdzināšana, lai visu stabiņu nolīdzināšanai nepieciešamais gājienu skaits būtu mazākais iespējamais!

Ievaddati

Teksta datnes `stabini.dat` pirmajā rindā doti divi naturāli skaitļi N ($1 < N \le 100\ 000$) un K ($1 < K < 20\ 000$), kas atdalīti ar tukšumzīmi. Datnes otrajā rindā doti N naturāli skaitļi, kur katri divi blakusesoši skaitļi atdalīti ar tukšumzīmi, – monētu skaits stabiņos. Katram i ($1 \le i \le N$) i-tais skaitlis rindā norāda monētu skaitu i-tajā stabiņā. Zināms, ka katrā stabiņā sākumā ir vismaz viena monēta un ir iespējams atrast divus stabiņus, kuros monētu skaits ir atšķirīgs.

Izvaddati

Teksta datnes `stabini.rez` vienīgajā rindā jāizvada naturāls skaitlis – tā stabiņa numurs, no kura sākot līdzināšanu, būs nepieciešams mazākais gājienu skaits. Ja mazāko gājienu skaitu iespējams sasniegt, sākot līdzināšanu no dažādiem stabiņiem, jāizvada mazākais no šo stabiņu numuriem.

Piemēri

Ievaddati	**Izvaddati**	**Piezīmes**
`10 8` `7 9 7 9 7 9 7 9 7 9`	`2`	
`5 4` `3 1 6 4 6`	`3`	Atbilst uzdevuma formulējumā dotajam piemēram.