

Uzdevuma nosaukums:	Apgabals zem trepītēm	Kārtas numurs	Punkti
Ievaddatu faila nosaukums:	<b>apgabals.dat</b>	<b>numurs.dat</b>	<b>punkti.dat</b>
Izvaddatu faila nosaukums:	<b>apgabals.rez</b>	<b>numurs.rez</b>	<b>punkti.rez</b>
Klases vārds risinājumam valodā <i>Java</i>	<b>Apgabals</b>	<b>Numurs</b>	<b>Punkti</b>
Izpildes laika ierobežojums vienam testpiemēram sekundēs (laiks tiek mērīts uz testēšanas servera):	<b>Pascal / C / C++</b>		
	<b>0,4</b>	<b>0,3</b>	<b>0,3</b>
	<b>Java</b>		
	<b>0,9</b>	<b>0,7</b>	<b>0,7</b>

Ievaddatu un izvaddatu failu nosaukumi jānorāda **bez** pilnā ceļa (uzskatiet, ka tie atrodas tekošajā katalogā) un tieši tā, kā norādīts uzdevuma formulējumā (**ar mazajiem burtiem**).

Izpildes laika atmiņas ierobežojums: **256MB**. Maksimāli iespējamais punktu skaits par uzdevumu ir **100**. Lai risinājums tiktu atzīts par derīgu pamattestēšanai, tam jāizdod pareiza atbilde **visiem** uzdevuma formulējumā dotajiem **piemēriem**.

Uzdevumu tekstos lietotais pieraksts  $A \leq x, y, z \leq B$  (kur A un B – skaitļi, bet x, y un z – kādi aprakstā lietoti mainīgie), nozīmē, ka vieni un tie paši skaitliskie ierobežojumi attiecas uz katru mainīgo atsevišķi, t.i., vienlaikus ir spēkā sakarības:  $A \leq x \leq B$ ,  $A \leq y \leq B$  un  $A \leq z \leq B$ . Līdzīgi,  $x, y < 100$  nozīmē, ka vienlaikus  $x < 100$  un  $y < 100$ .

Kompilējot programmas uz servera, tiks lietoti šādi kompilatori:

Valodai PASCAL:

- FreePascal (versija 2.6.4) ar parametriem  
`-O2 -XS -Sg -Cs64000000`

Valodai C:

- GNU C (versija 4.9.2) ar parametriem  
`-std=gnu99 -O2 -s -static -lm -xc -Wformat -Werror=format`

Valodai C++:

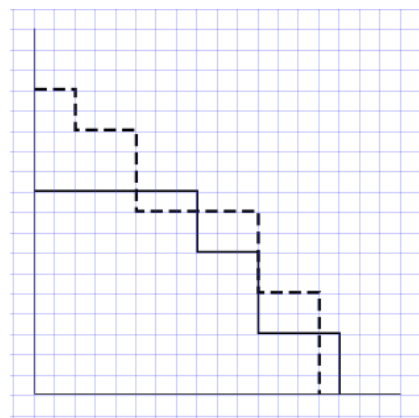
- GNU C++ (versija 4.9.2) ar parametriem  
`-std=gnu++11 -O2 -s -static -xc++ -Wformat -Werror=format`

Valodai Java:

- Java7 (versija OpenJDK 1.7.0\_65 jeb 7u65)

## Apgabals zem trepītēm

Par *trepītēm* saucim secīgu savstarpēji ortogonālu nogriežņu virkni, kas novilkta rūtiņu lapā pa rūtiņas sadalošajām līnijām un sastāv no pāra skaita vesela garuma nogriežņu. Aplūkosim tikai tādas trepītes, kas ierobežo pirmā kvadranta stūri: trepīšu pirmais nogrieznis garumā  $x_1$  ir novietots horizontāli un viens tā galapunkts atrodas uz ordinātu ass, tā galā atrodas uz leju vērsts vertikāls nogrieznis garumā  $y_1$ . Tam savukārt galā atrodas pa labi horizontāls nogrieznis garumā  $x_2$ , kam galā uz leju vērsts vertikāls nogrieznis garumā  $y_2$ . Tā viens aiz otra seko horizontālie un vertikālie nogriežņi līdz pēdējais vertikālais nogrieznis, kas ir garumā  $y_n$ , beidzas uz abscisu ass.



Attēlā redzamas divas trepītes: ar nepārtraukto līniju attēlotas trepītes, kuras nogriežņu garumi ir 8,3,3,4,4,3, bet ar raustīto – kuras nogriežņu garumi ir 2,2,3,4,6,4,3,5.

Interesēsīsimies par tā apgabala, kas vienlaikus atrodas zem visām dotajām trepītēm, lielumu – rūtiņu kopskaitu. Zem attēlā redzamajām divām trepītēm vienlaikus atrodas 107 rūtiņas.

Uzrakstiet programmu, kas dotam vienu vai vairāku trepīšu aprakstam nosaka tā apgabala lielumu, kas vienlaikus atrodas zem visām dotajām trepītēm!

### Ievaddati

Teksta datnes **apgabals.dat** pirmajā rindā dots trepīšu skaits - naturāls skaitlis  $N(N \leq 10^5)$ .

Nākamajās  $N$  datnes rindās seko trepīšu apraksts – katru trepīšu aprakstam veltot savu rindu. Katram  $i(1 \leq i \leq N)$   $i$ -to trepīšu apraksts dots datnes  $i+1$ -ajā rindā. Pirmais skaitlis ir trepīšu posmu skaits - naturāls pāru skaitlis  $p_i$ . Aiz tā šajā pat rindā doti  $p_i$  naturāli skaitļi – trepīšu posmu garumi, kuru kopsumma nepārsniedz  $2 \cdot 10^9$ .

Zināms, ka visu trepīšu posmu kopskaits nepārsniedz  $2 \cdot 10^5$  un starp katriem diviem blakus skaitļiem ievaddatos ir viena tukšumzīme.

### Izvaddati

Teksta faila **apgabals.rez** vienīgajā rindā jāizvada naturāls skaitlis – tā apgabala, kas vienlaikus atrodas zem visām dotajām trepītēm, lielums – rūtiņu kopskaits.

### Piemēri

Ievaddati (apgabals.dat)	Izvaddati (apgabals.rez)	Piezīme
2 6 8 3 3 4 4 3 8 2 2 3 4 6 4 3 5	107	Atbilst uzdevuma tekstā dotajam piemēram.

Ievaddati (apgabals.dat)	Izvaddati (apgabals.rez)
1 4 2 2 3 3	19

**1.apakšuzdevuma testu ievaddati**

ievaddati (apgabals.dat)
2
14 300 99 99 99 99 99 99 99 99 99 99 99 99 300
8 198 198 198 198 198 198 198 198 198

ievaddati (apgabals.dat)
2
10 80 160 160 275 59 45 501 80 80 80
8 160 160 351 53 49 187 240 160

ievaddati (apgabals.dat)
3
4 2942 4413 4413 2942
4 1471 4413 4413 4413
4 4413 4413 4413 1471

**Apakšuzdevumi un to vērtēšana**

Nr.	Testu apraksts	Punkti
1.	Uzdevuma tekstā dotie trīs testi	2
2.	Visu trepīšu visu posmu garumu kopsumma nepārsniedz $2 \cdot 10^6$	15
3.	$N=2$	18
4.	$N \leq 10$	20
5.	Bez papildus ierobežojumiem	45
Kopā:		100

## Kārtas numurs

Kādā valstī visu automašīnu numuri sastāv no viena un tā paša ciparu komplekta. Numuri tiek izdoti augošā secībā un nesākas ar ciparu 0. Kad vienā gadā ir izdoti 1000000009 numuri, tad numuru izsniegšana tiek pārtraukta un turpināta nākamajā gadā.

Piemēram, numurs 20231 ir 16. šajā gadā izdots numurs - pirms šī numura tika izsniegti numuri 10223, 10232, 10322, 12023, 12032, 12203, 12230, 12302, 12320, 13022, 13202, 13220, 20123, 20132, 20213.

Uzrakstiet programmu, kas dotam numuram nosaka, kurš pēc kārtas gadā tas ir izsniegts!

### Ievaddati

Teksta datnes **numurs.dat** pirmajā rindā dots naturāls skaitlis, kura pieraksts satur ne vairāk kā  $10^5$  ciparus un nesākas ar 0.

### Izvaddati

Teksta datnes **numurs.rez** vienīgajā rindā jāizvada naturāls skaitlis – dotā numura izsniegšanas kārtas numurs gadā.

### Piemēri

Ievaddati (numurs.dat)	Izvaddati (numurs.rez)	Piezīme
20231	16	Atbilst uzdevuma tekstā dotajam piemēram.

Ievaddati (numurs.dat)	Izvaddati (numurs.rez)	Piezīme
51111222233334444	9007992	Šis numurs ir izsniegts otrajā gadā. Visu izsniegto numuru virknē šis pēc kārtas ir 1009008001.numurs.

### 1.apakšuzdevuma testu ievaddati

Ievaddati (numurs.dat)	Ievaddati (numurs.dat)	Ievaddati (numurs.dat)
2016	31337	332211

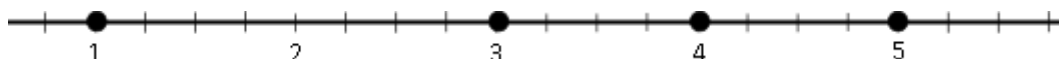
### Apakšuzdevumi un to vērtēšana

Nr.	Testu apraksts	Punkti
1.	Uzdevuma tekstā dotie trīs testi	2
2.	Numurā ir ne vairāk kā seši cipari	8
3.	Numurā katrs cipars parādās ne vairāk kā vienreiz	15
4.	Numurā ir ne vairāk kā 20 cipari	25
5.	Numurā ir ne vairāk kā 1000 cipari	25
6.	Bez papildus ierobežojumiem	25
Kopā:		100

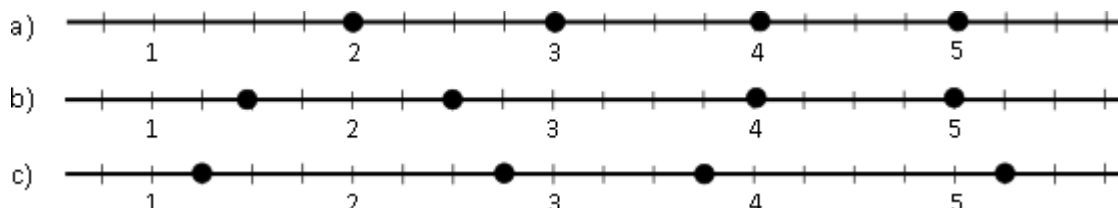
## Punkti

Uz bezgalīgi garas ass atzīmēti  $N$  punkti, katram ir zināms attālums no fiksēta ass punkta – vesels skaits centimetru. Nepieciešams punktus izvietot simetriski pret kādu ass punktu, ja nepieciešams, pārvietojot vienu vai vairākus punktus. Punktu pārvietojumam jābūt tādām, lai vistālāk pārvietotā punkta pārvietojums būtu mazākais iespējamais.

Piemēram, ja sākumā punktu atrašanās koordinātas ir 1, 3, 4 un 5 :



, tad simetrisku to izvietojumu var panākt vairākos veidos:



Variantā a) viens punkts ir pārvietots par vienu centimetru (no 1 uz 2), variantā b) divi punkti pārvietoti katrs par 0.5 cm (no 1 uz 1.5 un no 3 uz 2.5), bet variantā c) – visi četri punkti pārvietoti par 0.25 cm (no 1 uz 1.25, no 3 uz 2.75, no 4 uz 3.75 un no 5 uz 5.25). Līdz ar to labākais ir c) variants – neviens punkts nav pārvietots vairāk kā par 0.25 cm. Variantā b) šis lielums ir 0.5 cm, bet variantā a) 1 cm.

Uzrakstiet programmu, kas nosaka, kāds ir mazākais maksimālais kāda punkta pārvietojums, lai punktu konfigurācija kļūtu simetriska!

### Ievaddati

Teksta datnes **punkti.dat** pirmajā rindā dots punktu skaits – naturāls  $N(N \leq 10^5)$ . Datnes otrajā rindā dotas punktu koordinātas uz ass –  $N$  atšķirīgi veseli skaitļi augošā secībā. Katras koordinātas vērtība ir robežās no  $-10^{18}$  līdz  $10^{18}$ . Starp katriem diviem blakus skaitļiem ievaddatos ir viena tukšumzīme.

### Izvaddati

Teksta datnes **punkti.rez** vienīgajā rindā jāizvada reāls skaitlis ar tieši trim zīmēm aiz decimālā punkta - maksimālais kāda punkta pārvietojums, lai iegūtu simetrisku punktu konfigurāciju.

### Piemēri

Ievaddati (punkti.dat)	Izvaddati (punkti.rez)	Piezīme
4 1 3 4 5	0.250	Atbilst uzdevuma tekstā dotajam piemēram.
5 1 2 3 4 5	0.000	Dotā punktu konfigurācija jau ir simetriska – neviens punkts nav jāpārvieto.

**1.apakšuzdevuma testu ievaddati**

ievaddati (punkti.dat)
2
48325 349519

ievaddati (punkti.dat)
4
-397334 -271162 -113450 75808

ievaddati (punkti.dat)
6
-140629 -97768 -44193 20099 95106 180828

**Apakšuzdevumi un to vērtēšana**

Nr.	Testu apraksts	Punkti
1.	Uzdevuma tekstā dotie trīs testi	2
2.	N=3	4
3.	N=4	6
4.	N=5	8
5.	Koordinātu vērtības atrodas robežās no $-10^9$ līdz $10^9$	48
6.	Bez papildus ierobežojumiem	32
Kopā:		100