



LATVIJAS 38. INFORMĀTIKAS OLIMPIĀDE

IESILDĪŠANĀS SACENSĪBAS – 2024. GADA NOVEMBRIS

Daļu salīdzināšana

Grūtība: ★★☆☆☆

Uzrakstīt datorprogrammu, kas ievadītiem veseliem skaitļiem a, b, c un d nosaka, kura no daļām $\frac{a}{b}$ un $\frac{c}{d}$ ir lielāka, vai arī ka abas daļas ir vienādas.

Ievaddati

Ievaddatu pirmajā rindā dotas četrus veselu skaitļu a, b, c un d ($-10^{18} \leq a, b, c, d \leq 10^{18}$; $b, d \neq 0$) vērtības. Starp katriem diviem blakus skaitļiem ievaddatos ir tukšumzīme.

Izvaddati

Izvaddatu vienīgajā rindā jāizvada vesels skaitlis:

- -1 , ja $\frac{a}{b} < \frac{c}{d}$;
- 0 , ja $\frac{a}{b} = \frac{c}{d}$;
- 1 , ja $\frac{a}{b} > \frac{c}{d}$.

Ierobežojumi un prasības

Atmiņas apjoma un izpildes laika ierobežojumus skatīt kā paziņojumu testēšanas sistēmā.

Klases vārds valodā Java rakstītam risinājumam: **ABCD**

Piemēri

<i>Ievaddati</i>	<i>Izvaddati</i>
1 2 -3 -6	0
<i>Ievaddati</i>	<i>Izvaddati</i>
27 -11 -32 13	1
<i>Ievaddati</i>	<i>Izvaddati</i>
111 37 -82 -27	-1

1. apakšuzdevuma testu ievaddati

<i>Ievaddati</i>	
1489510958 -275178547 8159500073 -162901677	
<i>Ievaddati</i>	<i>Ievaddati</i>
-1276 1289 -3011 3036	123456 234567 345678 456789

Apakšuzdevumi un to vērtēšana

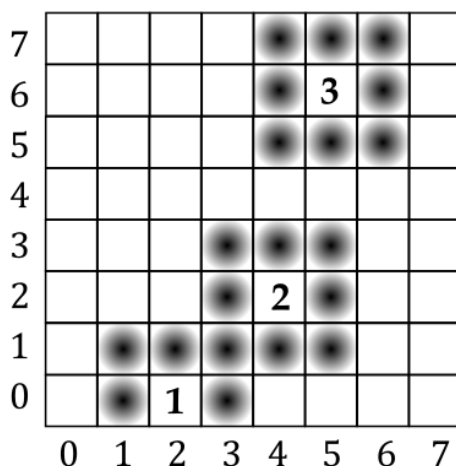
Nr.	Testu apraksts	Punkti
1.	Uzdevuma tekstā dotie trīs testi	2
2.	$-10^9 \leq a, b, c, d \leq 10^9$	23
3.	Bez papildu ierobežojumiem	75
Kopā:		100

Atklātās rūtiņas

Grūtība: ★★★★★

Pēteris ir izdomājis jaunu datorspēli, kas notiek uz $N \times N$ liela rūtiņu laukuma un kurā darbojas viens vai vairāki tēli. Katrs tēls atrodas kādā no laukuma rūtiņām. Vienā laukuma rūtiņā var atrasties vairāki tēli. Spēlētājam ir atklātas (redzamas) tikai tās rūtiņas, kuras atrodas netālu no kāda tēla, bet pārējās ir aizklātas (slēptas). Precīzāk – dotam skaitlim d spēlētājam ir redzamas tikai laukuma rūtiņas ar koordinātām $[t_{x_i}, t_{y_i}]$, kurās atrodas tēli, un visas rūtiņas, kuru koordinātām $[x, y]$ vienlaikus ir spēkā sakarības $|x - t_{x_i}| \leq d$ un $|y - t_{y_i}| \leq d$.

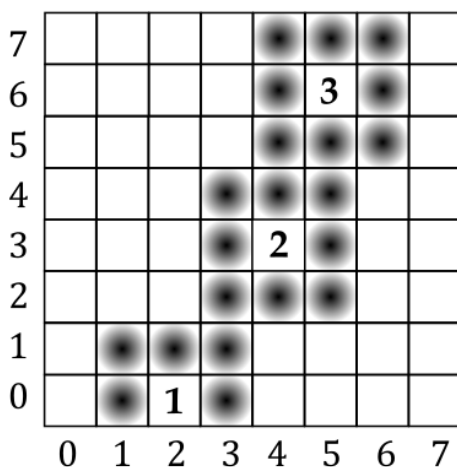
1. attēlā parādīts spēles laukuma piemērs, kur $N = 8, d = 1$ un tēli atrodas rūtiņās ar koordinātām $[2; 0]$, $[4; 2]$ un $[5; 6]$. Attēlā šīs rūtiņas atzīmētas ar cipariem.



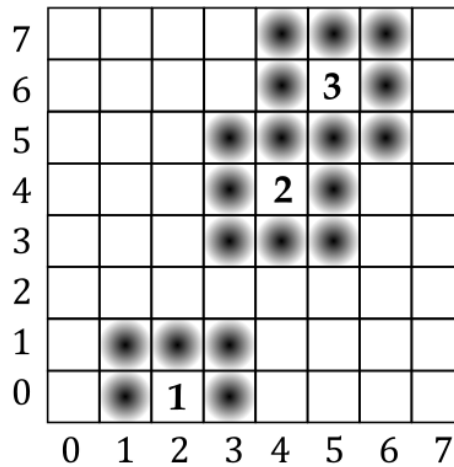
1. attēls: Laukuma piemērs

Šajā situācijā atklātas ir 23 no 64 laukuma rūtiņām.

Spēles gaitā tēli var pārvietoties uz blakus rūtiņām (kopīga mala) un tad mainās tas, kuras rūtiņas ir vai nav atklātas. Piemēram, ja 2. tēls pāriet vienu soli rindas indeksa pieaugšanas virzienā, tad atklāto rūtiņu skaits ir 24 (2. att.), bet, ja divus, tad – 22 (3. att.).



2. attēls: 2. tēls pagājis vienu soli



3. attēls: 2. tēls pagājis divus soļus

Uzrakstiet programmu, kas dotam tēlu izvietojumam un veiktajiem gājieniem nosaka katrā brīdī atklāto rūtiņu skaitu!

Ievaddati

Ievaddatu pirmajā rindā dotas četru veselu nenegatīvu skaitļu N (laukuma malas garums, $1 \leq N \leq 2000$), d (ap tēliem atklāto rūtiņu attālums, $0 \leq d \leq 30$), T (tēlu skaits, $1 \leq T \leq 10^5$) un G (tēlu izdarīto gājienu skaits, $0 \leq G \leq 2 \times 10^5$) vērtības.

Nākamajās T ievaddatu rindās katrā dotas viena tēla atrašanās vietas koordinātas – kolonnas numurs k ($0 \leq k < N$) un rindas numurs r ($0 \leq r < N$). Katram i ($1 \leq i \leq T$) i -tā tēla atrašanās vietas koordinātas ir dotas ievaddatu $i + 1$ -ajā rindā.

Nākamajās G ievaddatu rindās katrā dots viena tēla viena gājiena apraksts – tēla numurs t ($1 \leq t \leq T$) un v (pārvietošanās virziens, $1 \leq v \leq 4$). Virziena v vērtība 1 nozīmē, ka tēla atrašanās vietas rindas koordinātas vērtība palielinās par 1, vērtība 2 – ka kolonnas koordinātas vērtība palielinās par 1, vērtība 3 – ka rindas koordinātas vērtība samazinās par 1, bet vērtība 4 – ka kolonnas koordinātas vērtība samazinās par 1. Starp katriem diviem blakus skaitļiem ievaddatos ir tukšumzīme.

Izvaddati

Izvaddatos jābūt $G + 1$ rindai, kur katrā rindā ir naturāla skaitļa vērtība. Pirmajā rindā jābūt atklāto rūtiņu skaitam laukumā pirms tēli ir sākuši izdarīt gājienu. Katram i ($1 \leq i \leq G$) izvaddatu $i + 1$ -ajā rindā jābūt atklāto rūtiņu skaitam pēc pirmo i ievaddatos doto gājienu izdarīšanas.

Ierobežojumi un prasības

Atmiņas apjoma un izpildes laika ierobežojumus skatīt kā paziņojumu testēšanas sistēmā.

Klases vārds valodā Java rakstītam risinājumam: **Atklat**

Piemērs

<i>Ievaddati</i>	<i>Izvaddati</i>	<i>Piezīme</i>
8 1 3 2	23	Atbilst piemēram uzdevuma tekstā.
2 0	24	
4 2	22	
5 6		
2 1		
2 1		
2 1		

Apakšuzdevumi un to vērtēšana

Nr.	Testu apraksts	Punkti
1.	$G = 0$	2
2.	$T, G \leq 500$	18
3.	$d = 1$	32
4.	Bez papildu ierobežojumiem	48
Kopā:		100

Monētas

Grūtība: ★★★★★☆

Kasieris Naudiņš katru naudas summu S vēlas izsniegt, izmantojot pieejamo N nominālu monētas tā, ka kopējais monētu skaits ir mazākais iespējamais. Var uzskatīt, ka katra nomināla monētas Naudiņam ir pieejamas neierobežotā skaitā.

Piemēram, ja $S = 10$ un pieejamas trīs veidu monētas, kuru nomināli ir 3, 5 un 7 vienības, tad nepieciešamo naudas summu var izveidot ar divu monētu palīdzību: $3 + 7$ vai $5 + 5$. Savukārt, izmantojot tikai šī veida monētas, nav iespējams izveidot naudas summas, kuru vērtība ir 1, 2 vai 4.

Uzrakstiet programmu, kas dotai S vērtībai un monētu nomināliem atrod mazāko nepieciešamo monētu skaitu vai arī nosaka, ka šo vērtību izveidot nav iespējams!

Ievaddati

Ievaddatu pirmajā rindā dotas naturālu skaitļu N (monētu nominālu skaits, $N \leq 3000$) un S (naudas summa, $S \leq 3000$) vērtības. Nākamajā rindā doti N **atšķirīgi** naturāli skaitļi a_i ($a_i \leq 3000$) – pieejamo monētu nomināli. Starp katriem diviem blakus skaitļiem ievaddatos ir tukšumzīme.

Izvaddati

Izvaddatu vienīgajā rindā jāizvada vesels skaitlis – mazākais monētu skaits, kāds nepieciešams, lai izveidotu naudas summu S , vai -1 , ja summu S izveidot nav iespējams.

Ierobežojumi un prasības

Atmiņas apjoma un izpildes laika ierobežojumus skatīt kā paziņojumu testēšanas sistēmā.

Klases vārds valodā Java rakstītam risinājumam: **Monetas**

Piemēri

Ievaddati	Izvaddati	Ievaddati	Izvaddati
3 10 3 7 5	2	3 4 3 5 7	-1

1. apakšuzdevuma testu ievaddati

Ievaddati	Ievaddati	Ievaddati
5 100 3 7 11 13 19	7 1000 13 79 73 31 43 59 97	7 999 1 2 5 10 20 50 100

Apakšuzdevumi un to vērtēšana

Nr.	Testu apraksts	Punkti
1.	Uzdevuma tekstā dotie trīs testi	2
2.	$n \leq 2$	13
3.	$n, s \leq 50$	20
4.	Katram i eksistē vesels x , ka $a_i = 2^x$	31
5.	Bez papildu ierobežojumiem	34
Kopā:		100

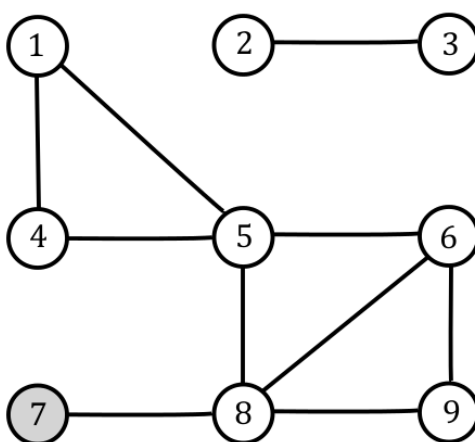
Netālās pilsētas

Grūtība: ★★★☆☆

Kādā valstī atrodas N pilsētas, kuras savā starpā saista ātrgaitas ceļi. Katrs ceļš ir divvirzienu un savieno divas pilsētas. Ārpus pilsētām ceļi nekrustojas.

Viena no pilsētām ir valsts **galvaspilsēta** un šajā uzdevumā interesēsīmies par patvaļīgi izvēlētas pilsētas attālumu līdz galvaspilsētai. Uzskatīsim, ka kāda pilsēta atrodas **netālu** no galvaspilsētas, ja no tās iespējams aizbraukt līdz galvaspilsētai, izbraucot cauri ne vairāk kā d citām pilsētām.

Viena pilsētu un ceļu izvietojuma piemērs attēlots 1. attēlā. Pilsētas sanumurētas ar skaitļiem no 1 līdz N pēc kārtas un attēlotas ar aplīšiem, bet ceļi – kā līniju nogriežņi, kas šos aplīšus savieno. Šajā piemērā galvaspilsēta ir 7. pilsēta.



1. attēls: Pilsētu un ceļu piemērs

Šajā piemērā, ja $d = 1$, tad netālās pilsētas ir 5., 6., 7., 8. un 9., bet visas pārējās tādas nav.

Uzrakstiet programmu, kas norādītajam pilsētas numuram nosaka, vai šī pilsēta atrodas netālu no galvaspilsētas!

Ievaddati

Ievaddatu pirmajā rindā dotas piecu veselu nenegatīvu skaitļu N (pilsētu skaits, $1 \leq N \leq 10^5$), M (ceļu skaits, $0 \leq M \leq 10^5$), G (galvaspilsētas numurs, $1 \leq G \leq N$), P (pārbaudāmo pilsētu skaits, $1 \leq P \leq N$) un d (maksimālais pilsētu, kurām drīkst izbraukt cauri, skaits, $0 \leq d < N$) vērtības. Nākamajās M rindās dots ceļu apraksts: divi atšķirīgi naturāli skaitļi, kas norāda ceļa galos esošo pilsētu numurus. Ievaddatu pēdējā rindā doti P atšķirīgi naturāli skaitļi – pārbaudāmo pilsētu numuri. Starp katriem diviem blakus skaitļiem ievaddatos ir tukšumzīme.

Izvaddati

Izvaddatu vienīgajā rindā jāizvada P veseli skaitļi. Katram i ($1 \leq i \leq P$) i -tajam skaitlim jābūt 1, ja ievaddatos pēc kārtas i -tā pārbaudāmā pilsēta atrodas netālu no galvaspilsētas, vai 0 – pretējā gadījumā.

Ierobežojumi un prasības

Atmiņas apjoma un izpildes laika ierobežojumus skatīt kā paziņojumu testēšanas sistēmā.

Klases vārds valodā Java rakstītam risinājumam: **NP**

Piemērs

<i>Ievaddati</i>	<i>Izvaddati</i>	<i>Piezīme</i>
9 10 7 5 1 1 4 3 2 5 1 6 5 4 5 7 8 6 8 6 9 8 5 9 8 5 7 1 3 8	1 1 0 0 1	Atbilst aprakstam uzdevuma tekstā.

1. apakšuzdevuma testu ievaddati

<i>Ievaddati</i>
6 10 2 6 0 5 1 5 4 5 3 1 4 1 3 1 6 1 2 4 3 3 6 6 2 1 6 4 2 3 5

Apakšuzdevumi un to vērtēšana

Nr.	Testu apraksts	Punkti
1.	Uzdevuma formulējumā dotais tests	2
2.	$N \leq 10$	23
3.	$N \leq 1000$	24
4.	No katras pilsētas var nokļūt uz katru citu, turklāt tikai vienā vienīgā veidā	25
5.	Bez papildu ierobežojumiem	26
Kopā:		100

Receptes

Grūtība: ★☆☆☆☆

Slavenais konditors Mārtiņš Vakariņš ir savācis lielu toršu recepšu kolekciju. Pēdējā laikā Vakariņš ir pamanījis, ka ir grūti atcerēties vai atrast noteiktas tortes recepti, tādēļ ir izveidojis datorsistēmu, kurā ievadījis visas zināmās receptes, katrai no tortēm norādot nepieciešamo izejvielu daudzumu kā N veselu nenegatīvu skaitļu virkni $a_1 a_2 \dots a_N$. Katram $i (1 \leq i \leq N)$ a_i norāda i -tās sastāvdaļas daudzumu noteiktās vienībās (piemēram, cukura vai miltu mērvienība būs izteikta gramos, olu dzeltenumi gabalos, utt.). Ja konkrētai receptei attiecīgā sastāvdaļa nav nepieciešama, tad atbilstošā a_i vērtība ir 0.

Vakariņš vēlas izgatavot pēc iespējas vairāk toršu pēc vienas receptes un zina, cik katra veida izejvielas šobrīd ir noliktavā. Viņš vairākām receptēm vēlas noskaidrot, kādu lielāko daudzumu toršu iespējams izgatavot pēc katras receptes.

Piemēram, ja $N = 4$ un izejvielu daudzumi ir tādi, kā norādīts tabulā:

	1. izejviela	2. izejviela	3. izejviela	4. izejviela
Noliktavā	8	7	10	6
1. recepte	1	0	3	2
2. recepte	2	1	2	1
3. recepte	3	4	5	0

, tad būs iespējams izgatavot vai nu trīs tortes pēc pirmās receptes, vai četras tortes pēc otrās receptes, vai arī vienu torti pēc trešās receptes.

Uzrakstiet programmu, kas dotam izejvielu daudzumam noliktavā un recepšu aprakstiem aprēķina toršu skaitu!

Ievaddati

Pirmajā rindā dotas divu naturālu skaitļu N (izejvielu veidu skaits, $N \leq 10^5$) un R (dažādo recepšu skaits, $1 < R \leq 10$) vērtības, kas atdalītas ar tukšumzīmi.

Nākamajā rindā doti N nenegatīvi veseli skaitļi, kur katri divi blakus skaitļi atdalīti ar tukšumzīmi. Katram $i (1 \leq i \leq N)$ i -tais skaitlis norāda i -tās izejvielas daudzumu noliktavā.

Nākamajās R rindās katrā dots vienas tortes receptes apraksts - tortes pagatavošanai nepieciešamais katra veida izejvielu daudzums.

Katrā rindā doti N nenegatīvi veseli skaitļi, kur katri divi blakus skaitļi atdalīti ar tukšumzīmi. Katram $i (1 \leq i \leq N)$ i -tais skaitlis norāda receptē nepieciešamo i -tās izejvielas daudzumu. Zināms, ka katras tortes receptei vismaz vienas izejvielas daudzums ir pozitīvs skaitlis.

Nevienai izejvielai tās daudzums noliktavā vai receptē nepārsniedz 10^9 vienības.

Izvaddati

Izvaddatiem jāsaturs tieši R rindas. Katram $r (1 \leq r \leq R)$ r -tajā rindā jāizvada vesels nenegatīvs skaitlis – lielākais toršu skaits, kādu iespējams izgatavot pēc šīs receptes.

Ierobežojumi un prasības

Atmiņas apjoma un izpildes laika ierobežojumus skatīt kā paziņojumu testēšanas sistēmā.

Klases vārds valodā Java rakstītam risinājumam: **Rec**

Piemērs

<i>Ievaddati</i>	<i>Izvaddati</i>
4 3	3
8 7 10 6	4
1 0 3 2	1
2 1 2 1	
3 4 5 0	

<i>Ievaddati</i>	<i>Izvaddati</i>
5 2	0
6 4 0 3 5	2
1 1 1 1 1	
1 1 0 1 2	

1. apakšuzdevuma testu ievaddati

<i>Ievaddati</i>
4 2
251 433 109 77
11 7 8 5
5 2 6 3

<i>Ievaddati</i>
4 2
999 999 999 999
8 7 6 5
11 12 13 14

<i>Ievaddati</i>
4 2
765 432 987 789
8 7 9 8
76 43 98 78

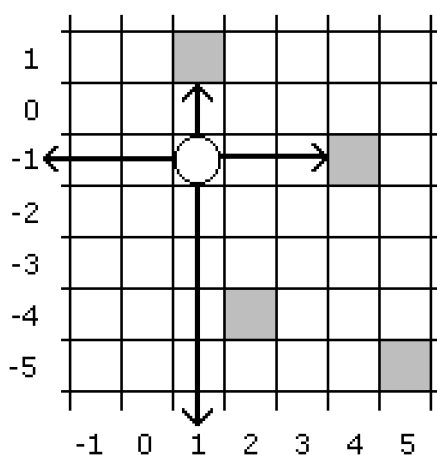
Apakšuzdevumi un to vērtēšana

Nr.	Testu apraksts	Punkti
1.	Uzdevuma tekstā dotie trīs testi	2
2.	$N \leq 3$	10
3.	Nevienai izejvielai tās daudzums noliktavā vai receptē nepārsniedz 1000	28
4.	Bez papildu ierobežojumiem	60
Kopā:		100

Tornis

Grūtība: ★★★★★

Uz bezgalīgas rūtiņu lapas, kurai dažas rūtiņas var būt izgrieztas, novietota šaha figūra - tornis. Lapas rindas un kolonnas ir sanumurētas pēc kārtas ar veseliem skaitļiem. Kolonnas ir numurētas no kreisās puses uz labo, bet rindas – no lejas uz augšu. Tornis vienā gājienā var pārvietoties uz jebkuru citu neizgrieztu lapas rūtiņu, kas ar sākotnējo atrodas vienā rindā vai kolonnā. Tornis nedrīkst pārvietoties pāri izgrieztai rūtiņai. Laukuma piemērā 1. attēlā redzams, uz kurām rūtiņām tornis drīkst pārvietoties, ja četras lapas rūtiņas ir izgrieztas un tornis pirms gājiena atrodas rūtiņā ar koordinātām $(1; -1)$.



1. attēls: Laukuma piemērs

Kā redzams, šajā gadījumā tornis var pārvietoties bezgalīgi tālu horizontālā virzienā pa kreisi vai vertikālā virzienā uz leju. Noskaidrot, ar kādu mazāko gājienu skaitu tornis no vienas norādītās rūtiņas var nokļūt līdz otrai norādītajai rūtiņai (ja tas vispār ir iespējams), pārvietojoties saskaņā ar iepriekš aprakstītajiem noteikumiem.

Ievaddati

Ievaddatu pirmajā rindiņā ir doti četri veseli skaitļi x_1, y_1, x_2, y_2 . x_1 (kolonnas numurs) un y_1 (rindas numurs) norāda tās rūtiņas koordinātas, kurā tornis atrodas sākumā, bet x_2 un y_2 analogiskā veidā norāda tās rūtiņas numuru, kurā tornim jānokļūst beigās. Zināms, ka $-10^9 \leq x_1, y_1, x_2, y_2 \leq 10^9$ un ka neviena no šīm divām rūtiņām nav izgriezta. Ievaddatu otrajā rindā dots naturāls skaitlis n – izgrieztu rūtiņu skaits ($n \leq 1000$). Ievaddatu nākamajās n rindās katrā doti divi veseli skaitļi, kas atdalīti ar tukšumsimbolu - vienas izgrieztās rūtiņas koordinātas. Ievaddatu $i + 2$ -ajā rindā ($1 \leq i \leq n$) ir dots i -tās izgrieztās rūtiņas kolonnas numurs p_i ($-10^9 \leq p_i \leq 10^9$) un rindas numurs q_i ($-10^9 \leq q_i \leq 10^9$).

Izvaddati

Izvaddatu vienīgajā rindā jāizvada viens vesels skaitlis – mazākais gājienu skaits, kurā tornis no vienas norādītās rūtiņas var nokļūt līdz otrai norādītajai rūtiņai. Ja tornis nevar nokļūt līdz otrai norādītajai rūtiņai, tad faila vienīgajā rindā jāizvada skaitlis -1 .

Ierobežojumi un prasības

Atmiņas apjoma un izpildes laika ierobežojumus skatīt kā paziņojumu testēšanas sistēmā.

Klases vārds valodā Java rakstītam risinājumam: **Tornis**

Piemēri

<i>Ievaddati</i>	<i>Izvaddati</i>
1 -1 5 -4 4 1 1 5 -5 2 -4 4 -1	3

<i>Ievaddati</i>	<i>Izvaddati</i>
10 11 5001 -4733 5 5001 -4732 5001 -4734 1 1 5000 -4733 5002 -4733	-1

Apakšuzdevumi un to vērtēšana

Nr.	Testu apraksts	Punkti
1.	$N \leq 10$	16
2.	$N \leq 100$	36
3.	Bez papildu ierobežojumiem	48
Kopā:		100