

**LATVIJAS 15. INFORMĀTIKAS OLIMPIĀDES III POSMA UZDEVUMU APSKATS**  
**Pirmā diena (2002.gada 25.marts)**



Uzdevuma nosaukums:	<b>Svari</b>	<b>Nams</b>	<b>Operācija @</b>
Programmas nosaukums:	SVARI.EXE	NAMS.EXE	OPERA.EXE
Ievaddatu faila nosaukums:	SVARI.DAT	NAMS.DAT	OPERA.DAT
Izvaddatu faila nosaukums:	SVARI.REZ	NAMS.REZ	OPERA.REZ
Izpildes laika ierobežojums vienam testam* :	1 sekunde	1 sekunde	2 sekundes
Testu skaits:	12	12	10
Maksimāli iespējamais punktu skaits par uzdevumu:	<b>100</b>	<b>100</b>	<b>100</b>
Palīgfaili uz dalībnieka disketes	Ievaddatu faila piemērs Izvaddatu faila piemērs Rezultāta pārbaudes programma <b>svartst.exe</b>	Ievaddatu faila piemērs Izvaddatu faila piemērs	Ievaddatu faila piemērs Izvaddatu faila piemērs

Datu un rezultātu failus norādiet **bez** pilnā ceļa (uzskatiet, ka datu un rezultātu faili atrodas tekošajā katalogā)!

Neaizmirstiet saglabāt savas programmas (izejas tekstus un .exe moduļus) ar norādītajiem nosaukumiem uz disketēm saknes katalogā!

\* Izpildes laika ierobežojums norādīts datoram, uz kura programmas tiks testētas.

Sacensību un testa datora ātrdarbības salīdzināšanai izmantojiet programmu « **atrums.exe** », kas atrodas uz disketes.

Jautājumus par uzdevumu formulējumiem centieties uzdot sacensību pirmās stundas laikā!

**LATVIJAS 15. INFORMĀTIKAS OLIMPIĀDES**  
**III POSMA UZDEVUMI**  
**Pirmā diena (2002.gada 25.marts)**



### 1. "Svari"

Ir doti  $k$  atsvari, kuru svāri ir zināmi. Šie atsvari ir apzīmēti ar latīņu alfabēta pirmajiem  $k$  burtiem.

Bez tam ir divi atsvari, kuru svārs nav zināms. Šie atsvari ir apzīmēti ar burtiem  $x$  un  $y$ .

Katra atsvara (gan zināmā, gan nezināmā) svārs ir izsakāms veselos kilogramos un nepārsniedz  $10^8$  kg.

Ir svāru svāri, kur katrā kausā atļāuts likt patvaļīgu doto atsvaru skaitu.

Ir izdevies iegūt visu svēršanu rezultātus, kas veikti saskaņā ar sekojošiem noteikumiem:

- katrā svēršanā piedalās vismaz viens no atsvariem  $x$  vai  $y$ .
- katrā svēršanā uz katra no kausiem ir uzlikts vismaz viens atsvars
- katrā svēršanā vismaz viens no atsvariem  $x$  vai  $y$  atrodas kreisajā kausā.
- ja svēršanā piedalās gan  $x$ , gan  $y$ , tad  $x$  noteikti atrodas kreisajā kausā.

Uzrakstiet programmu, kas ievadītiem svēršanu rezultātiem nosaka, cik kilogramus sver atsvari  $x$  un  $y$ !

#### **Ievaddati**

Teksta faila SVARI.DAT pirmajā rindā dota naturāla skaitļa  $k$  (zināmo atsvaru skaits,  $1 \leq k \leq 10$ ) vērtība. Nākošajā faila rindā doti  $k$  naturāli skaitļi, kas atdalīti ar tukšumsimboliem - zināmo atsvaru svārs kilogramos. Pirmais skaitlis ir atsvara  $a$  svārs, otrais - atsvara  $b$  svārs, utt. alfabētiskā secībā.

Nākošajā rindā dots naturāls skaitlis  $s$  - svēršanu skaits. Dotajam zināmo atsvaru skaitam ir doti visi iespējamo svēršanu rezultāti, kas atbilst iepriekšaprakstītajiem ierobežojumiem.

Nākošajās  $s$  faila rindās dots pa vienai svēršanai katrā. Katra rinda ir formātā

*<kreisā\_kausa\_saturs><zīme><labā\_kausa\_saturs>*

Katra kausa saturu norāda kā šajā kausā esošajiem atvāriem atbilstošo burtu virkni.

Zīme "<" nozīmē, ka labais kauss ir bijis smagāks par kreiso, ">" nozīmē, ka kreisais kauss ir bijis smagāks par labo un "=" nozīmē, ka kausi ir bijuši līdzsvarā. Svēršanu aprakstos nav tukšumsimbolu.

#### **Izvaddati**

Teksta faila SVARI.REZ vienīgajā rindā jāizvada divu naturālu skaitļu  $x$  un  $y$  vērtības - atsvaru svāri kilogramos. Starp abiem skaitļiem jābūt vienam tukšumsimbolam.

Ja  $x$  un  $y$  vērtības viennozīmīgi noteikt nav iespējams, jāizvada jebkurš derīgs  $x$  un  $y$  vērtību pāris.

#### **Piemērs**

Ievaddati(fails SVARI.DAT)

Izvaddati(fails SVARI.REZ)

1

73 146

73

6

x=a

x<y

xa=y

x<ya

xy>a

y>a

## 2. "Nams"

Kāds uzņēmējs uzbūvējis jaunu namu ar  $N$  stāviem. Katrā stāvā ir  $K$  dzīvokļi, kurus jaunais īpašnieks vēlas izīrēt. Interesi par dzīvokļu īri ir izrādījuši jau  $M$  potenciālie īrnieki, turklāt par katru no viņiem īpašniekam ir zināms mazākais un lielākais stāva numurs, kurā īrnieks ir ar mieru apmesties. Īpašnieks ir ieinteresēts izīrēt pēc iespējas vairāk dzīvokļus.

Uzrakstiet programmu, kas nosaka, kā pēc iespējas vairāk īrniekiem izīrēt pa dzīvoklim tā, lai īrniekam piešķirtais dzīvoklis atrastos viņam tīkamā stāvā!

### *Ievaddati*

Faila `NAMS.DAT` pirmā rinda satur trīs veselus skaitļus  $N$ ,  $K$  un  $M$ , kas atdalīti ar vienu tukšumsimbolu ( $1 \leq N \leq 100$ ,  $1 \leq K \leq 100$ ,  $1 \leq M \leq 10000$ ). Nākamās  $M$  faila rindas satur pa diviem veseliem skaitļiem katrā –  $i+1$ -ā rinda satur skaitļus  $A$  un  $B$ , kas atdalīti ar tukšumsimbolu un apzīmē attiecīgi mazāko un lielāko stāva numuru, kurā  $i$ -tais īrnieks ir ar mieru dzīvot ( $1 \leq A \leq B \leq N$ ).

### *Izvaddati*

Failam `NAMS.REZ` jāsatur tieši  $M$  rindas. Katrā rindā jābūt veselam skaitlim. Ja  $i$ -tajam īrniekam ir izīrēts dzīvoklis  $j$ -tajā stāvā, tad  $i$ -tajai rindai jāsatur skaitlis  $j$ . Ja  $i$ -tajam īrniekam dzīvoklis nav izīrēts, tad  $i$ -tajā rindā jābūt skaitlim 0.

Ja uzdevumam iespējami vairāki atrisinājumi, jāizvada jebkurš no tiem.

### *Piemērs*

Ievaddati(fails <code>NAMS.DAT</code> )	Izvaddati(fails <code>NAMS.REZ</code> )
3 2 6	1
1 2	1
1 1	3
3 3	0
1 1	2
2 2	2
1 2	

### 3. "Operācija @"

Cipariem no 0 līdz  $n$  ( $1 \leq n \leq 9$ ) ir definēta operācija @, kuras rezultāts arī ir cipars robežās no 0 līdz  $n$ . Piemēram, ja  $n=2$ , tad @ var aprakstīt ar sekojošas tabulas palīdzību, attēlojot visas iespējamās operācijas  $a@b$  vērtības (kur  $a$  un  $b$  - cipari robežās no 0 līdz 2):

		<b>b</b>		
		0	1	2
<b>a</b>	0	<b>1</b>	<b>1</b>	<b>0</b>
	1	<b>2</b>	<b>1</b>	<b>0</b>
	2	<b>1</b>	<b>2</b>	<b>2</b>

Kā redzams, šajā gadījumā operācija nav nedz komutatīva, jo  $0@1 = 1$ , bet  $1@0 = 2$ , nedz asociatīva ( $0@1$ )@2 = 0, bet  $0@(1@2)=1$ .

Ja dota izteiksme  $x_1@x_2@x_3@x_4@...@x_{k-1}@x_k$  (kur katrs  $x_i$  ir robežās no 0 līdz  $n$ ), tad tās vērtību aprēķina no kreisās puses uz labo tāpat kā izteiksmes (...((( $x_1@x_2$ )@ $x_3$ )@ $x_4$ )@... $x_{k-1}$ )@ $x_k$ ) vērtību, sākot ar visdziļāk iekļautajām iekavām - vispirms aprēķina  $x_1@x_2$  rezultātu  $r_1$ , pēc tam aprēķina  $r_1@x_3$  rezultātu, utt.

Mainot iekavu izvietojumu sākotnējā izteiksmē, tās rezultāts var mainīties.

Piemēram, ja dota izteiksme  $1@1@1@1@0$ , tad piemērā redzamajai operācijai @ izteiksmes  $(1@(1@1))@(1@0)$  vērtība ir 0, bet  $((1@1)@1)@0$  vērtība ir 2.

Uzrakstiet programmu, kas dotam  $n$ , ar tabulu uzdotai operācijai @ un ciparu virknei  $x_1x_2x_3x_4...x_{k-1}x_k$  katram veselam skaitlim  $t$  ( $0 \leq t \leq n$ ) nosaka, vai izteiksmē  $x_1@x_2@x_3@x_4@...@x_{k-1}@x_k$  iespējams salikt iekavas tā, lai šīs izteiksmes vērtība būtu  $t$ .

#### Ievaddati

Teksta faila OPERA.DAT pirmajā rindā dota naturāla skaitļa  $n$  vērtība ( $1 \leq n \leq 9$ ). Nākošajā  $n+1$  faila rindā ir uzdota operācija @. Katrā no šīm rindām ir  $n+1$  cipars: faila  $i+2$ -ās rindas  $j+1$ -ais cipars ir operācijas  $i@j$  rezultāts ( $0 \leq i \leq n, 0 \leq j \leq n$ ). Starp katriem diviem blakus cipariem failā ir viens tukšumsimbols.

Faila  $n+3$ -ajā rindā ir dota ciparu virkne  $x_1x_2x_3x_4...x_{k-1}x_k$  bez atdalošiem tukšumiem. Virknes garums ir vismaz divi un ne vairāk kā 250 cipari.

#### Izvaddati

Teksta failam OPERA.REZ jāsatur tieši  $n+1$  rinda. Katram  $t$  ( $0 \leq t \leq n$ ) faila  $t+1$ -ajā rindā jābūt vārdam "VAR", ja iekavas dotajā izteiksmē iespējams salikt tā, lai izteiksmes vērtība būtu  $t$ , vai "NEVAR", ja šādi iekavas izvietot nav iespējams.

#### Piemērs

Ievaddati(fails OPERA.DAT)

2  
1 1 0  
2 1 0  
1 2 2  
11110

Izvaddati(fails OPERA.REZ)

VAR  
NEVAR  
VAR

**LATVIJAS 15. INFORMĀTIKAS OLIMPIĀDES III POSMA UZDEVUMU APSKATS**  
**Otrā diena (2002.gada 26.marts)**

Uzdevuma nosaukums:	<b>Joslas</b>	<b>Tornis</b>
Programmas nosaukums:	JOSLAS.EXE	TORNIS.EXE
Ievaddatu faila nosaukums:	JOSLAS.DAT	TORNIS.DAT
Izvaddatu faila nosaukums:	JOSLAS.REZ	TORNIS.REZ
Izpildes laika ierobežojums vienam testam*:	3 sekundes	2 sekundes
Testu skaits:	15	15
Piezīmes par vērtēšanu:	Iespējams daļējs punktu skaits, ja laukuma noklājums ir korekts, bet izmantoto joslu skaits vai krāsojums nav korekts	Ja visiem testa datiem programma dos atbildi "NEVAR", par šo uzdevumu punkti netiks piešķirti
Maksimāli iespējamais punktu skaits par uzdevumu:	<b>150</b>	<b>150</b>
Palīgfaili uz dalībnieka disketes	Ievaddatu faila piemērs Izvaddatu faila piemērs Rezultāta pārbaudes programma <b>jostst.exe</b>	Ievaddatu faila piemērs Izvaddatu faila piemērs

Datu un rezultātu failus norādiet **bez** pilnā ceļa (uzskatiet, ka datu un rezultātu faili atrodas tekošajā katalogā)!

Neaizmirstiet saglabāt savas programmas (izejas tekstus un .exe moduļus) ar norādītajiem nosaukumiem uz disketēm saknes katalogā!

\* Izpildes laika ierobežojums norādīts datoram, uz kura programmas tiks testētas.

Sacensību un testa datora ātrdarbības salīdzināšanai izmantojiet programmu « **atrums.exe** », kas atrodas uz disketes.

Jautājumus par uzdevumu formulējumiem centieties uzdot sacensību pirmās stundas laikā!

**LATVIJAS 15. INFORMĀTIKAS OLIMPIĀDES**  
**III POSMA UZDEVUMI**  
**Otrā diena (2002.gada 26.marts)**



### 1. "Joslas"

Ir dots rūtiņu laukums, kas sastāv no  $n \times n$  kvadrātveida rūtiņām. Laukuma rindas un kolonnas ir numurētas pēc kārtas ar skaitļiem no 1 līdz  $n$ .

Bez tam ir dotas  $2n$  joslas, kur katras joslas izmērs ir  $n \times 1$  vienība. Tieši  $n$  joslas ir melnas (visas to rūtiņas ir nokrāsotas melnā krāsā), un  $n$  joslas ir baltas (visas to rūtiņas ir nokrāsotas baltā krāsā).

Šīs joslas pēc kārtas tiek liktas uz laukuma tā, ka katra josla noklāj tieši vienu (katra savu) laukuma rindu vai kolonnu. Beigās katrai laukuma rūtiņai atbilst (ir redzama, skatoties no augšas) vai nu melna, vai balta rūtiņa.

Viena iespējama  $3 \times 3$  laukuma noklāšana ir parādīta zīmējumā (ar krustiņiem apzīmētas nenoklātās rūtiņas):



Uzrakstiet programmu, kas ievadītam laukuma beigu noklājumam atrod vienu iespējamu joslu likšanas secību!

#### **Ievaddati**

Teksta faila JOSLAS.DAT pirmajā rindā dota naturāla skaitļa  $n$  (laukuma malas garums,  $1 < n \leq 2000$ ) vērtība.

Katrā no nākošajām  $n$  faila rindām ir doti tieši  $n$  burti ("M" vai "B"), kas atdalīti ar tukšumsimboliem. Ja  $j$ -tais burts faila  $i+1$ -jā rindā ir "M", tas nozīmē, ka laukuma  $i$ -tās rindas  $j$ -tās kolonnas atbilstošā rūtiņa ir melna. Atbilstošs burts "B" nozīmē, ka laukuma rūtiņa ir balta.

Zināms, ka katram dotajam testam eksistē vismaz viens atrisinājums.

#### **Izvaddati**

Teksta failam JOSLAS.REZ jāsaturs tieši  $2n$  rindas.  $i$ -tajā faila rindā jābūt aprakstītai  $i$ -tās pēc kārtas joslas uzlikšanai formā

`<joslas_krāsa><tukšumsimbols><numurs><tukšumsimbols><rinda_vai_kolonna>`

Joslas krāsa jāapzīmē ar vienu burtu - B (balta) vai M (melna).

Numuram jābūt robežās no 1 līdz  $n$  un jāatbilst tās laukuma rindas vai kolonnas numuram, kura tiek noklāta ar kārtējo joslu.

Rinda vai kolonna jāapzīmē ar vienu burtu - R (rinda) vai K (kolonna).

Ja dotajiem datiem iespējami vairāki atrisinājumi, jāizvada viens no tiem.

#### **Piemērs**

Ievaddati(fails JOSLAS.DAT)

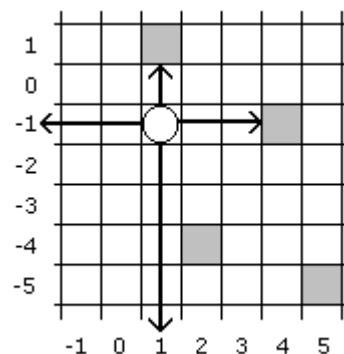
```
3
B B B
B B M
M M M
```

Izvaddati(fails JOSLAS.REZ)

```
M 2 R
B 1 K
M 3 K
B 1 R
B 2 K
M 3 R
```

## 2."Tornis"

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. Zīmējumā redzams, uz kurām rūtiņām tornis drīkst pārvietoties, ja četras lapas rūtiņas ir izgrieztas un tornis pirms gājiena atrodas rūtiņā ar koordinātām (1;-1). 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 norādītas rūtiņas var nokļūt līdz otrai norādītai rūtiņai (ja tas vispār ir iespējams), pārvietojoties saskaņā ar iepriekšaprakstītajiem noteikumiem.

### Ievaddati

Faila `TORNIS.DAT` pirmajā rindiņā ir doti četri veseli skaitļi  $x_1, y_1, x_2, y_2$ , kas atdalīti ar tukšumsimboliem.  $x_1$  (kolonnas numurs) un  $y_1$  (rindas numurs) norāda tās rūtiņas koordinātes, 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.

Faila otrajā rindā dots naturāls skaitlis  $n$  - izgrieztu rūtiņu skaits ( $n \leq 1000$ ).

Faila nākošajās  $n$  rindās katrā doti divi veseli skaitļi, kas atdalīti ar tukšumsimbolu - vienas izgrieztās rūtiņas koordinātas. Faila  $i+2$ -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

Faila `TORNIS.REZ` vienīgajā rindā jāizvada viens vesels skaitlis - mazākais gājienu skaits, kurā tornis no vienas norādītas rūtiņas var nokļūt līdz otrai norādītai rūtiņai. Ja tornis nevar nokļūt līdz otrai norādītajai rūtiņai, tad faila vienīgajā rindā jāizvada vārds "NEVAR".

### Piemēri

Ievaddati (`TORNIS.DAT`)

1 -1 5 -4

4

1 1

5 -5

2 -4

4 -1

Izvaddati (`TORNIS.REZ`)

3

Ievaddati (`TORNIS.DAT`)

10 11 5001 -4733

5

5001 -4732

5001 -4734

1 1

5000 -4733

5002 -4733

Izvaddati (`TORNIS.REZ`)

NEVAR