

**LATVIJAS 23.INFORMĀTIKAS OLIMPIĀDES IV POSMS**  
**UZDEVUMU APSKATS**  
**Pirmā diena (2010.gada 16.aprīlis)**



Uzdevuma nosaukums:	<b>TAISNSTŪRU SKAITS</b>	<b>APKOPĒJA ROMBA</b>	<b>TRAUSLĀ KLAVIATŪRA</b>
Ievaddatu faila nosaukums:	ts.dat	romba.dat	trakla.dat
Izvaddatu faila nosaukums:	ts.rez	romba.rez	trakla.rez
Izpildes laika ierobežojums vienam testpiemēram (laiks tiek mērīts uz testēšanas servera):	2 sekundes	2 sekundes	0,2 sekundes
Atmiņas ierobežojums:	64MB	128MB	64MB
Maksimāli iespējamais punktu skaits par uzdevumu:	100	100	100

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

Kompilējot programmas uz servera, tiks lietoti šādi kompilatori:

Valodai PASCAL:

- FreePascal (versija 2.2.0) ar parametriem `-O2 -Sg -Cs16777216`

Valodai C:

- GNU C (versija 3.4.2 un 4.4.1) ar parametriem  
`-std=c99 -O2 -s -static -lm -Wl,--stack,16777216`
- Microsoft Visual C 2008 ar parametriem  
`/TC /O2 /link /STACK:16777216`

Valodai C++:

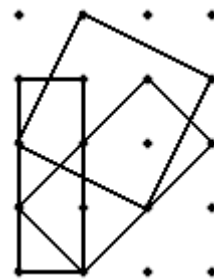
- GNU C++ (versija 3.4.2 un 4.4.1) ar parametriem  
`-O2 -s -static -Wl,--stack,16777216`
- Microsoft Visual C++ 2008 ar parametriem  
`/TP /O2 /link /STACK:16777216`

## TAISNSTŪRU SKAITS

$N \times M$  punkti izvietoti regulāra rūtiņu režģa virsotnēs.

Zīmējumā parādīts  $4 \times 5$  punktu liels režģis, kurā iezīmēti trīs taisnstūri, kuru visas virsotnes atrodas režģa punktos.

Uzrakstiet programmu, kas aprēķina, cik ir tādu taisnstūru, kuru visas virsotnes atrodas dotā režģa punktos!



### Ievaddati

Teksta faila **ts.dat** pirmajā rindā dotas divu naturālu skaitļu  $N$  ( $2 \leq N \leq 5\,000$ ) un  $M$  ( $2 \leq M \leq 5\,000$ ) vērtības, kas atdalītas ar tukšumzīmi.

### Izvaddati

Teksta faila **ts.rez** vienīgajā rindā jāizvada naturāls skaitlis – dažādo taisnstūru, kuru virsotnes ir dotā režģa punktos, skaits.

### Piemēri

Ievaddati (ts.dat)	Izvaddati (ts.rez)
4 2	6

Ievaddati (ts.dat)	Izvaddati (ts.rez)
3 5	33

## APKOPĒJA ROMBA

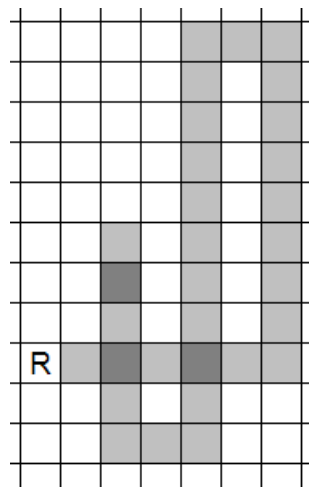
Apkopēja Romba uzkopj ļoti lielas pils ļoti lielas centrālās zāles ļoti lielo grīdu, kuru klāj vienādas kvadrātveida flīzes.

Katrā brīdī Romba atrodas uz kādas no flīzēm un tīra priekšā esošo flīzi. Sākumā Romba atrodas uz kādas no neiztīrītajām flīzēm (zīmējumā šī flīze apzīmēta ar „R”) un sāk tīrīt flīzes taisnā virzienā pēc kārtas (zīmējumā pa labi). Pēc kārtējās flīzes notīrīšanas Romba pārvietojas uz pēdējo notīrīto flīzi un, nemainot tīrīšanas virzienu, sāk tīrīt nākamo flīzi.

Kad Rombai apnīk pārvietoties taisnā virzienā, viņa uz pēdējās notīrītās flīzes pagriežas pa kreisi vai pa labi (var pagriezties arī vairākas reizes pēc kārtas). Pēc tam Romba atkal kādu gabalu tīra flīzes taisnā virzienā.

Katru darbību (taisnajā virzienā notīrīto flīžu skaitu un pagriešanās virzienu) Romba pieraksta, lai darbadienas beigās aprēķinātu kopējo notīrīto flīžu skaitu.

Uzrakstiet programmu, kas pēc Rombas pierakstiem aprēķina kopējo notīrīto flīžu skaitu!



### **ievaddati**

Teksta faila **romba.dat** pirmajā rindā dots naturāls skaitlis N (darbību skaits,  $N \leq 300\,000$ ). Katrā no nākamajām N rindām dots vienas Rombas darbības apraksts. Taisnvirziena flīžu tīrīšanai atbilst burts T, kam pēc tukšumzīmes seko naturāls skaitlis S ( $S \leq 1\,000\,000$ ) – šajā gājienā notīrīto flīžu skaits. Pagriezienam pa kreisi atbilst burts K, bet pagriezienam pa labi – burts L. Visas darbības dotas tādā secībā, kādā tās tika izpildītas.

### **Izvaddati**

Teksta faila **romba.rez** vienīgajā rindā jāizvada naturāls skaitlis – kopējais Rombas notīrīto flīžu skaits.

### **Piemērs**

ievaddati (romba.dat)	Izvaddati (romba.rez)	Piezīme
15 T 6 K T 3 T 5 K T 2 K T 10 L T 2 L T 5 K K T 1	31	Atbilst dotajam zīmējumam

## TRAUSLĀ KLAVIATŪRA

Arnoldam ir trausla klaviatūra, kuras īpašība ir tāda, ka katrs nospiestais taustiņš nostrādā tikai pirmajā nospiešanas reizē. Piemēram, mēģinot ievadīt vārdu „ABRAKADABRA”, iegūsim „ABRKD”. Tādu pašu rezultātu iegūtu, mēģinot ievadīt vārdu „ABARAKADABA” vai „ABRRRBRRRAKAD”.

Dotam rezultātā iegūtajam vārdam iespējams sastādīt to vārdu sarakstu „Arnolda īpašajā” secībā, kurus ievadot varēja iegūt minēto vārdu. Piemēram, vārdam „ARK” šādu vārdu saraksta sākums ir šāds:

**ARK, AARK, ARAK, ARKA, ARKK, ARKR, ARRK, AAARK, AARAK, ...**

„Arnolda īpašajā” secībā īsāki vārdi atrodas pirms garākiem, un vienāda garuma vārdi sakārtoti alfabētiski.

Uzrakstiet programmu, kas dotam rezultātā iegūtajam vārdam un kārtas numuram, kāds ievadāmais vārds ir visu vārdu ar šo rezultātu „Arnolda īpašajā” secībā sakārtotajā sarakstā, atrod ievadāmo vārdu!

Sarakstā vārdu indeksācija sākas ar 1.

### ***levaddati***

Teksta faila **trakla.dat** pirmajā rindā dotas divu naturālu skaitļu  $N$  ( $2 \leq N \leq 26$ ) un  $K$  ( $K \leq 10^{15}$ ) vērtības, kas atdalītas ar tukšumzīmi.  $K$  norāda vārda kārtas numuru vārdu sarakstā, kas sakārtots „Arnolda īpašajā” secībā. Faila otrajā rindā dots rezultātā iegūtais vārds – latīņu alfabēta lielo burtu virkne garumā  $N$ , kurā visi burti ir atšķirīgi.

### ***Izvaddati***

Teksta faila **trakla.rez** vienīgajā rindā jāizvada vārds, kas „Arnolda īpašajā” secībā sakārtotajā sarakstā ir  $K$ -tais pēc kārtas.

### ***Piemēri***

levaddati (trakla.dat)	Izvaddati (trakla.rez)
3 9 ARK	AARAK
levaddati (trakla.dat)	Izvaddati (trakla.rez)
2 17 OH	OHOHO