

# **Zbirka algoritamskih zadataka osnovni nivo**

<https://petlja.org/biblioteka/r/kursevi/Zbirka>  
2018.11.29

# 1. Aritmetika

## 1.1. Formule

### 1.1.1. Geometrijske formule

#### 1.1.1.1. Trening

Sportista se na početku treninga zagreva tako što trči po ivicama pravougaonog terena dužine  $d$  i širine  $s$ . Napisati program kojim se određuje koliko metara pretrči sportista dok jednom obiđe teren.

##### *Ulaz*

U prvoj liniji standardnog ulaza se nalazi celobrojna vrednost  $d$ , a u sledećoj liniji celobrojna vrednost  $s$  ( $0 < d \leq 100$ ,  $0 < s \leq 100$ ) koje redom predstavljaju dužinu i širinu terena izraženu u metrima.

##### *Izlaz*

Jedan ceo broj koji predstavlja broj metara koje pretrči sportista dok jednom obiđe teren.

#### **Primer**

##### *Ulaz*

50  
25

##### *Izlaz*

150

#### 1.1.1.2. List papira

Na stolu se nalaze listovi papira različitih dimenzija, ali su svi pravougaonog oblika. Program treba na osnovu zadate širine i visine lista u milimetrima da odredi površinu lista u kvadratnim milimetrima.

##### *Ulaz*

U jednoj liniji standardnog ulaza nalaze se dve celobrojne vrednosti  $V$  i  $S$  ( $0 < V \leq 300$ ,  $0 < S \leq 300$ ) koje predstavljaju širinu i visinu lista papira izraženu u milimetrima.

##### *Izlaz*

Jedan ceo broj koji predstavlja površinu lista u kvadratnim milimetrima

**Primer****Ulaz**

279 216

**Izlaz**

60264

**1.1.1.3. Stolnjak**

Napisati program kojim se izračunava potrebna dužina trake za porub stolnjaka kružnog oblika čija je površina P.

**Ulaz**

U liniji standardnog ulaza se nalazi pozitivna realna vrednost P.

**Izlaz**

Jedan realan broj koji predstavlja potrebnu dužinu trake za porub zaokružen na pet decimala.

**Primer****Ulaz**

7853.982

**Izlaz**

314.15927

**1.1.1.4. Rastojanje tačaka**

Napiši program koji izračunava i ispisuje rastojanje između tačaka zadatih svojim koordinatama.

**Ulaz**

Sa standardnog ulaza unose se četiri realna broja, svaki u posebnom redu. Prva dva broja Ax i Ay predstavljaju koordinate tačke A=(Ax,Ay), dok druga dva broja Bx i By predstavljaju koordinate tačke B=(Bx,By).

**Izlaz**

Na standardni izlaz ispisati jedan realan broj koji predstavlja rastojanje između tačaka A i B, zaokružen na pet decimala.

**Primer****Ulaz**

0

0

1

1

**Izlaz**

1.41421

### 1.1.1.5. Pravougaonik dat naspramnim temenima

Date su koordinate dva naspramna temena pravougaonika čije su stranice paralelne koordinatnim osama. Napisati program kojim se prikazuju dužina dijagonale, obim i površina tog pravougaonika.

#### **Ulaz**

Sa standardnog ulaza učitavaju se 4 cela broja iz intervala  $[-1000, 1000]$ , svaki u posebnoj liniji. Učitani brojevi predstavljaju redom x i y koordinatu dva naspramna temena pravougaonika.

#### **Izlaz**

Na standardnom izlazu prikazati dužinu dijagonale pravougaonika, zaokruženu na dve decimale, obim i površinu datog pravougaonika. Svaki podatak prikazati u posebnoj liniji.

#### **Primer**

##### **Ulaz**

5  
-1  
-3  
6

##### **Izlaz**

10.63  
30  
56

### 1.1.1.6. Površina trougla datih temena

Napiši program koji izračunava površinu ako su poznate njegove koordinate temena.

#### **Ulaz**

Sa standardnog ulaza unosi se 6 realnih brojeva (svaki u zasebnom redu). Prva dva predstavljaju x i y koordinatu temena A, druga dva x i y koordinatu temena B i poslednja dva predstavljaju x i y koordinatu temena C.

#### **Izlaz**

Na standardni izlaz ispisati jedan realan broj koji predstavlja površinu trougla (dopuštena greška izračunavanja je  $10^{-5}$ ).

#### **Primer**

##### **Ulaz**

0  
0  
0  
1  
1

0

**Izlaz**

0.5

**1.1.1.7. Fudbalski teren**

Fudbalski teren dimenzija  $d \times s$  treba ograditi pravougaonom ogradom tako da je rastojanje stranica ograde od linije terena  $r$ . Odrediti dužinu ograde.

**Ulaz**

$d$  - dužina terena u metrima ( $90 \leq d \leq 120$ )

$s$  - širina terena u metrima ( $45 \leq s \leq 90$ )

$r$  - rastojanje ograde od terena u metrima ( $2 \leq r \leq 10$ )

**Izlaz**

Dužina ograde u metrima.

**Primer****Ulaz**

100

60

3

**Izlaz**

344

**1.1.1.8. Nivo bazena**

Tokom oluje palo je  $n$  litara vode po kvadratnom metru. Napisati program kojim se određuje za koliko centimetara se podigao nivo vode u otvorenom bazenu datih dimenzija, ako znamo da se voda nije prelila preko bazena.

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se koliko litara vode je palo po kvadratnom metru, realan broj  $n$ . U sledeće tri linije nalaze se dimenzije bazena izražene u metrima i to redom dužina, širina i dubina realni brojevi  $a, b, c$ .

**Izlaz**

Na standardnom izlazu prikazati jedan realan broj na dve decimale koji predstavlja za koliko centimetara se podigao nivo vode u bazenu.

**Primer****Ulaz**

75.5

4

7

1.7

**Izlaz**

7.55

**1.1.1.9. Trougao određen pravom i koordinatnim početkom**

Odredi površinu i obim trougla kojeg prava  $y=a \cdot x+b$  ( $a,b \neq 0$ ), zadata koeficijentima  $a$  i  $b$ , obrazuje sa koordinatnim osama.

**Ulaz**

Dva realna broja različita od 0. U prvoj liniji  $a$ , a u drugoj  $b$ .

**Izlaz**

Prikazati u prvoj liniji površinu trougla, a u drugoj obim, zaokružene na dve decimale.

**Primer****Ulaz**

2.5

3.0

**Izlaz**

1.80

7.43

**1.1.2. Kretanje****1.1.2.1. Putovanje**

Porodica je krenula autom na letovanje. Automobilom treba da pređu  $s$  kilometara krećući se ravnomernom brzinom od  $v$  km/h. Napisati program kojim se određuje koliko će im sati trebati da pređu put.

**Ulaz**

U prvoj liniji standardnog ulaza se nalazi realna vrednost  $v$ , a u sledećoj liniji realna vrednost  $s$  koje redom predstavljaju brzinu izraženu u km/h i planirani pređeni put u kilometrima.

**Izlaz**

Jedan realan broj zaokružen na dve decimale koji predstavlja potrebno vreme u satima.

**Primer****Ulaz**

60

1050

**Izlaz**

17.50

### 1.1.2.2. Biciklista

Biciklista se kretao ravnomernom brzinom od  $v_0$  m/s. Na početku dugačke nizbrdice počeo je da se kreće ubrzano sa ubrzanjem od  $a$  m/s<sup>2</sup>. Napisati program kojim se određuje koliki će deo nizbrdice, izraženo u metrima, da pređe nakon jednog minuta, kao i kolika će mu biti brzina u tom trenutku. NAPOMENA: Kod ravnomerno ubrzanog kretanja pređeni put se izražava formulom  $s = v_0 \cdot t + a \cdot t^2$ , dok se brzina nakon vremena  $t$  izražava formulom  $v = v_0 + a \cdot t$ .

#### Ulaz

U prvoj liniji standardnog ulaza se nalazi realna vrednost  $v_0$  koja predstavlja brzinu izraženu u m/s, a u sledećoj liniji realna vrednost  $a$  koja predstavlja ubrzanje izraženo u m/s<sup>2</sup>.

#### Izlaz

Dva realna broja, svaki u posebnom redu, koja predstavljaju pređeno rastojanje u metrima i traženu brzinu u metrima u sekundi.

#### Primer

##### Ulaz

10  
0.2

##### Izlaz

960.00  
22.00

### 1.1.2.3. Sustizanje automobila

Sa istog startnog mesta kreću dva automobila prvi se kreće brzinom  $v_1$  m/s, drugi  $v_2$  m/s pri čemu je  $v_2 > v_1$ . Drugi automobil polazi  $t$  minuta posle prvog. Napisati program kojim se određuje posle koliko minuta od polaska drugi automobil sustiže prvog.

#### Ulaz

Unose se tri realna broja koja redom predstavljaju brzine prvog i drugog automobila izražene u m/s i broj minuta posle koliko kreće drugi automobil. Svaki broj je u posebnoj liniji standardnog ulaza.

#### Izlaz

Na standardnom izlazu prikazati broj minuta zaokružen na dve decimale posle koliko će drugi automobil sustiže prvog.

#### Primer

##### Ulaz

20  
25  
10

**Izlaz**

40.00

#### 1.1.2.4. Rastojanje kuća

Pera i Mika žive u istoj ulici, Mikina kuća je udaljenija od škole. Oni idu u školu istim putem, polaze iz kuće u isto vreme i ravnomerno se kreću. Pera se kreće brzinom  $v_1$  m/s, a Mika brzinom  $v_2$  m/s ( $v_2 > v_1$ ). Napisati program kojim se određuje koliko je rastojanje između njihovih kuća, ako je posle  $t$  sekundi Mika bio  $d$  metara iza Pere.

**Ulaz**

Unose se četiri realna broja koji redom predstavljaju brzinu kretanja Pere ( $v_1$ ), brzinu kretanja Mike ( $v_2$ ), brzine su izražene u ms, broj sekundi ( $t$ ) i rastojanje između Pere i Mike u metrima ( $d$ ). Svaki podatak je u posebnoj liniji standardnog ulaza.

**Izlaz**

Na standardnom izlazu prikazati realan broj, na dve decimale, koji predstavlja koliko je rastojanje između njihovih kuća.

#### Primer

**Ulaz**

1.6  
2.1  
10  
30.0

**Izlaz**

35.00

#### 1.1.2.5. Kolona

Vojnici se kreću u koloni dužine  $d$  metara, brzinom  $v$  km/h. Komandir koji se nalazi na kraju kolone, šalje kurira na biciklu sa porukom na čelo kolone. Predavši poruku, kurir se odmah vraća nazad. Brzina kurira bicikliste je  $v_k$  km/h ( $v_k > v$ ). Napisati program kojim se određuje posle koliko vremena se kurir vraća na kraj kolone.

**Ulaz**

Na standardnom ulazu nalaze se tri pozitivna realna broja, svaki u posebnom redu, koji redom predstavljaju dužinu kolone u metrima ( $m$ ), i brzine vojnika i kurira izražene u kilometrima na čas (km/h).

**Izlaz**

Na standardnom izlazu ispisate vreme posle koliko se kurir vraća na kraj kolone. Vreme prikazati u minutama na dve decimale.



**Primer****Ulaz**

850

3.5

20

**Izlaz**

5.26

**1.1.2.6. Braća i pas**

Dva brata nalaze se na udaljenosti  $d$  metara, kreću se istim pravcem a suprotnim smerovima. Prvi kreće se kreće ravnomerno brzinom od  $v_1$  km/h a drugi  $v_2$  km/h. Pas koji je brži od oba brata ravnomerno trči brzinom od  $v_{pms}$ . Pas trči od prvog ka drugom bratu, kada stigne do drugog okreće se i trči nazad ka prvom, kada stigne okreće se i trči ka drugom i to ponavlja sve dok se braća ne sretnu. Napiši program koji određuje koliko je pas metara pretrčao.

**Ulaz**

Na standardnom ulaza nalaze se četiri realna broja, svaki u posebnoj liniji, koji redom predstavljaju rastojanje između braće izraženo u m, brzine dva brata izražene u km/h, redom brzina prvog brata pa brzina drugog brata, i brzinu psa izraženu u ms.

**Izlaz**

Ispisuje se jedan realan broj zaokružen na dve decimale koji predstavlja broj metara koje je pas pretrčao.

**Primer****Ulaz**

500.00

3.00

2.00

4.00

**Izlaz**

1440.00

**1.1.3. Linearne jednacine i sistemi****1.1.3.1. Zbirovi po tri stranice pravougaonika**

Ako je zbir neke tri stranice pravougaonika  $m$ , a zbir neke druge tri stranice istog pravougaonika  $n$ , napisati program kojim se određuje obim i površina tog pravougaonika.

**Ulaz**

Prva linija standardnog ulaza sadrži realan broj  $m$  koji predstavlja zbir neke tri stranice pravougaonika, druga linija sadrži realan broj  $n$  koji predstavlja zbir neke druge tri stranice pravougaonika.

**Izlaz**

U prvoj liniji standardnog izlaza prikazati na dve decimale obim pravougaonika, a u drugoj liniji površinu pravougaonika takođe na dve decimale.

**Primer****Ulaz**

8  
10

**Izlaz**

12.00  
8.00

**1.1.3.2. Prosek na takmičenju**

Na takmičenju iz matematike učestvovalo je  $N$  učenika. Izveštaj o broju poena odštampan je na dve strane. Nastavnik greškom nije poneo prvu stranu izveštaja, ali se seća da je na dnu strane pisalo da je prosečan broj poena za tu stranu bio  $p_1$ . Na drugoj strani (koju ima kod sebe) su podaci o  $K$  učenika i prosečan broj poena za tu stranu je  $p_2$ . Napisati program kojim se određuje koliki je prosečan broj poena svih učenika.

**Ulaz**

Na standardnom ulazu nalaze se  
u prvoj liniji prirodan broj  $N$  ukupan broj učenika ( $N \leq 200$ )  
u drugoj liniji prirodan broj  $K$  broj učenika na drugoj strani ( $K \leq 100$ )  
u trećoj liniji realan broj  $p_1$  prosečan broj poena učenika na prvoj strani ( $0 \leq p_1 \leq 100$ )  
u četvrtoj liniji realan broj  $p_2$  prosečan broj poena učenika na drugoj strani ( $0 \leq p_2 \leq 100$ )

**Izlaz**

Na standardnom izlazu prikazati, na dve decimale, prosečan broj svih učenika.

**Primer****Ulaz**

80  
30  
78.20  
89.30

**Izlaz**

82.36

### 1.1.3.3. Ograda terase

Na terasi dužine  $d$  metara treba rasporediti  $n$  stubića širine  $s$  centimetara tako da rastojanje između stubića, kao i između stubića i zida bude isto.

#### **Ulaz**

$d$  - dužina terase u metrima ( $1 \leq d \leq 20$ )

$n$  - broj stubića ( $1 \leq n \leq 100$ )

$s$  - širina stubića u centimetrima ( $5 \leq s \leq 15$ )

#### **Izlaz**

Rastojanje između stubića u centimetrima, zaokružen na dve decimale.

#### **Primer**

##### **Ulaz**

10

30

10

##### **Izlaz**

22.58

### 1.1.4. Linearna zavisnost

#### 1.1.4.1. Kurs

Napisati program kojim se na osnovu datog kupovnog i prodajnog kursa evra prema dinaru i kupovnog i prodajnog kursa dolara prema dinaru, izračunava kupovni i prodajni kurs dolara prema evru i evra prema dolaru. Kupovni kurs valute 1 prema valuti 2 predstavlja iznos valute 1 koji se može kupiti jedinicom valute 2 (valuta 1 se tretira kao roba koja se kupuje), a prodajni kurs valute 1 prema valuti 2 predstavlja iznos valute 2 koji se može dobiti prodajom jedinice valute 1 (valuta 1 se tretira kao roba koja se prodaje). Na primer, kupovni kurs evra prema dinaru je broj dinara potrebnih da bi se kupio jedan evro, dok je prodajni kurs evra prema dinaru broj dinara koji se mogu dobiti prodajom jednog evra. Sve transakcije evra prema dolaru i obrnuto se obavljaju, po propisima NBS, preko konverzije u dinare.

#### **Ulaz**

U svakoj od četiri linije standardnog ulaza nalaze se redom kupovni kurs evra prema dinaru, prodajni kurs evra prema dinaru, kupovni kurs dolara prema dinaru, prodajni kurs dolara prema dinaru (realni brojevi).

#### **Izlaz**

U svakoj od četiri linije standardnog izlaza ispisuju se redom realni brojevi, kupovni kurs dolara prema evru, prodajni kurs dolara prema evru, kupovni kurs evra prema dolaru i prodajni kurs evra prema dolaru, zaokruženi na 4 decimale.

**Primer****Ulaz**

123.59  
122.85  
109.72  
109.06

**Izlaz**

0.8931  
0.8824  
1.1332  
1.1197

**1.1.4.2. Generator slučajnih brojeva**

U računarstvu se često javlja potreba za generisanjem slučajnih (nasumičnih) brojeva. Većina programskih jezika pružaju funkciju koja vraća nasumično odabran realan broj iz intervala  $[0,1)$  (ovo znači da je taj broj veći ili jednak 0, a strogo manji od 1), podrazumevajući da korisnik može naknadnim aritmetičkim transformacijama dobiti broj iz bilo kog intervala  $[a,b)$  (za koji važi da je  $a < b$ ). Napiši program kojim prikazuješ kako se to radi.

**Ulaz**

Sa standardnog ulaza učitava se jedan realan broj  $x$  iz intervala  $[0,1)$  za koji pretpostavljaš da ga je vratio generator slučajnih brojeva, zatim, u narednom redu jedan realan broj  $a$  i onda u narednom redu broj  $b$  (pri tom važi  $a < b$ ).

**Izlaz**

Na standardni izlaz ispiši jedan realan broj iz intervala  $[a,b)$  koji se dobija kada se uneti broj  $x$  preslika linearnom funkcijom koja preslikava interval  $[0,1)$  na interval  $[a,b)$ . Rezultat prikazati zaokružen na 5 decimala.

**Primer 1****Ulaz**

0.5  
-2  
3

**Izlaz**

0.50000

**Primer 2****Ulaz**

0  
-2  
3

**Izlaz**

-2.00000

**Primer 3****Ulaz**

0.8

-2

3

**Izlaz**

2.00000

**1.1.4.3. Grupa radnika**

$n$  radnika uradi posao za  $s$  sati. Napisati program kojim se određuje za koliko sati će posao biti završen ako se priključi još  $m$  radnika?

**Ulaz**

U tri linije standardnog ulaza se redom nalaze prirodan broj  $n$ , realan broj  $s$  i prirodan broj  $m$ .

**Izlaz**

Jedan realan broj, zaokružen na dve decimale, koji predstavlja broj sati potrebnih da se posao završi ukoliko se priključi još  $m$  radnika početnom broju od  $n$  radnika.

**Primer****Ulaz**

1

3

1

**Izlaz**

1.5

**1.1.4.4. Podela intervala**

Dat je interval  $[a,b]$  realne prave. Napisati program kojim se određuje tačka realne prave koja interval deli u razmeri  $p:q$ .

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se početak intervala realan broj  $a$ , u drugoj liniji kraj intervala realan broj  $b$  ( $a < b$ ). Treća i četvrta linija sadrži prirodne brojeve redom  $p$  i  $q$ .

**Izlaz**

Prikazuje se jedan realan broj na dve decimale koji predstavlja traženu tačku.

**Primer****Ulaz**

0  
10  
1  
4

**Izlaz**

2.00

**1.1.4.5. Taksi**

U taksi su istovremeno ušla tri putnika, ali su izlazili jedan po jedan. Dogovorili su se da u plaćanju svakog dela vožnje podjednako učestvuju putnici koji su u tom delu vožnje bili u taksiju. Ako se zna stanje na taksimetru u dinarima kada je izašao svako od njih, koliko treba da plati putnik koji je izašao prvi, koliko drugi, a koliko treći?

**Ulaz**

Realni brojevi, u svakom redu po jedan, u sledećem redosledu:

c1 - stanje na taksimetru u momentu izlaska prvog putnika  
c2 - stanje na taksimetru u momentu izlaska drugog putnika  
c3 - stanje na taksimetru u momentu izlaska trećeg putnika

Pri tome važi  $c1 < c2 < c3$ .

**Izlaz**

Tri realna broja zaokružena na dve decimale (u svakom redu po jedan), koji predstavljaju troškove putovanja prvog, drugog i trećeg putnika.

**Primer****Ulaz**

1200.0  
1500.0  
2000.0

**Izlaz**

400.00  
550.00  
1050.00

**1.1.5. Stepenovanje i korenovanje****1.1.5.1. Stepen i koren**

Đoka je vežbao množenje decimalnih brojeva tako što je za neki broj  $a$  računao proizvod  $a \cdot a \cdot \dots \cdot a$  ( $n$  puta), tj. proizvod u kojem se broj  $a$  javlja  $n$  puta kao činilac. Nakon toga se sa svojom drugaricom Danicom igrao tako što

joj je zadavao da ona pogodi broj  $a$  kada joj on kaže broj  $n$  i rezultat koji je dobio. Napiši program koji Đoki pomaže da proveriti da li je dobro računao, a Danici pomaže da proveriti da li je dobro pogodila nepoznat broj.

#### **Ulaz**

Sa standardnog ulaza unosi se decimalni broj  $x$  (sa najviše dve decimale) i ceo broj  $n$ .

#### **Izlaz**

Na standardni izlaz se ispisuju dva realna broja (zaokružena na pet decimala) i to svaki u posebnom redu:

broj  $y$  takav da je  $y = x \cdot x \cdot \dots \cdot x$  ( $n$  puta)

broj  $y$  takav da je  $x = y \cdot y \cdot \dots \cdot y$  ( $n$  puta).

#### **Primer**

##### **Ulaz**

2.5

5

##### **Izlaz**

97.65625

1.20112

### **1.1.6. Aritmetički i geometrijski niz**

#### **1.1.6.1. N-ti dan treninga**

Vaterpolista Marko učestvuje na pripremama za takmičenje u trajanju od  $n$  dana. Prvog dana priprema Marko prepliva  $a$  metara, a svakog sledećeg dana za  $d$  metara više nego prethodnog dana. Napisati program kojim se određuje koliko metara Marko prepliva poslednjeg dana priprema.

##### **Ulaz**

Unose se tri prirodna broja, svaki u posebnoj liniji standardnog ulaza. Brojevi redom predstavljaju broj dana koliko pripreme traju  $n$  ( $n \leq 20$ ), koliko metara Marko prepliva prvog dana priprema  $a$  ( $a \leq 3000$ ), koliko metara svaki dan Marko više pliva u odnosu na prethodni dan  $d$  ( $d \leq 1000$ ).

##### **Izlaz**

Na standardnom izlazu prikazati koliko metara Marko prepliva poslednjeg dana priprema.

#### **Primer**

##### **Ulaz**

10

500

100

**Izlaz**

1400

**1.1.6.2. Sportske pripreme**

Vaterpolistima kluba "Delfini" organizovane su pripreme u trajanju od  $n$  dana. Prvog dana priprema svaki vaterpolisti prepliva po  $a$  metara a svakog sledećeg dana za  $d$  metara više nego prethodnog dana. Napisati program kojim se određuje koliko ukupno kilometara, tokom tih priprema, prepliva svaki vaterpolista kluba "Delfini".

**Ulaz**

Unose se tri prirodna broja, svaki u posebnoj liniji standardnog ulaza. Brojevi redom predstavljaju broj dana koliko pripreme traju  $n$  ( $n \leq 20$ ), koliko metara svaki vaterpolista kluba "Delfin" prepliva prvog dana priprema  $a$  ( $a \leq 3000$ ), koliko metara svaki dan vaterpolista kluba "Delfin" više pliva u odnosu na prethodni dan  $d$  ( $d \leq 1000$ ).

**Izlaz**

Na standardnom izlazu prikazati koliko kilometara ukupno prepliva svaki vaterpolista tokom pripreme, u obliku realnog broja zaokruženog na dve decimale.

**Primer****Ulaz**

10

500

100

**Izlaz**

9.50

**1.1.6.3. Proizvodnja malina**

Vlasnik imanja, Marinko, odlučio je da gaji maline. Napravio je plan za  $n$  godina. Prve godine planira da proizvede  $t$  tona malina, a svake sledeće da poveća proizvodnju za  $p\%$ . Napisati program kojim se određuje koliko tona malina Marinko planira da proizvede  $n$ -te godine uzgajanja malina.

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $n$  ( $n \leq 10$ ) broj planiranih godina. Druga i treća linija standardnog ulaza sadrži po jedan realan broj, koji redom predstavljaju koliko tona malina Marinko planira da proizvede prve godine  $t$  ( $0 < t < 5$ ) i za koliki procenata Marinko planira da poveća proizvodnju svake godine  $p$  ( $10 \leq p \leq 50$ ).

**Izlaz**

Na standardnom izlazu prikazati, na dve decimale, koliko tona malina Marinko planira da proizvede  $n$ -te godine uzgajanja malina.



**Primer****Ulaz**

5  
2  
25

**Izlaz**

4.88

**1.1.6.4. Suma niza brojeva**

Profesor matematike je postavio sledeći zadatak:

Odrediti sumu  $n$  brojeva, ako je prvi broj dati broj  $a$ , a svaki sledeći broj dobija se tako što prethodni broj pomnožimo sa datim brojem  $q$ .

Učenik koji prvi reši zadatak dobija peticu. Pomozite učenicima, napišite program kojim se za dati prirodan broj  $n$ , i date realne brojeve  $a$  i  $q$  određuje tražena suma.

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $n$  ( $n \leq 50$ ).

Druga linija standardnog ulaza sadrži realan broj  $a$  ( $0 < a \leq 10$ ).

Treća linija standardnog ulaza sadrži realan broj  $q$  ( $0 < q < 1$ ).

**Izlaz**

Na standardnom izlazu prikazati, na pet decimala, traženu sumu brojeva.

**Primer****Ulaz**

4  
20.00  
0.50

**Izlaz**

37.50000

**1.1.6.5. Aritmetički trougao**

Koliki je zbir brojeva u datom redu sledećeg trougla?

```

      1
     2 3 4
    5 6 7 8 9
   10 11 12 13 14 15 16
  ...

```

**Ulaz**

Redni broj reda trougla čiju sumu treba izračunati.

**Izlaz**

Suma vrednosti u zadatom redu trougla, pri čemu brojanje redova počinje od 1.

**Primer****Ulaz**

3

**Izlaz**

35

**1.1.6.6. Aritmetički kvadrat**

Brojevi od 0 do  $n^2-1$  ređaju se u kvadrat. Na primer, za  $n=5$  dobija se sledeći kvadrat:

```
0  1  2  3  4
5  6  7  8  9
10 11 12 13 14
15 16 17 18 19
20 21 22 23 24
```

Napiši program koji izračunava zbir brojeva u datoj vrsti i datoj koloni kvadrata (i njih ćemo brojati od nule).

**Ulaz**

Sa standardnog ulaza učitavaju se sledeći celi brojevi (svaki u posebnom redu).

$n$  - dimenzija kvadrata ( $1 < n \leq 1000$ )

$i$  - broj vrste i kolone čiji se zbir traži ( $0 \leq i < n$ )

**Izlaz**

Na standardni izlaz ispisati dva cela broja, svaki u posebnom redu:

zbir brojeva u  $i$ -toj vrsti kvadrata i

zbir brojeva u  $i$ -toj koloni kvadrata

**Primer****Ulaz**

5

1

**Izlaz**

35

55

**1.1.6.7. Rastvor**

Masena koncentracija soli u rastvoru definiše se kao količnik mase soli i zapremine rastvora. Eksperiment započinje sa 1 litara rastvora masene koncentracije  $\rho$  g/dm<sup>3</sup> tj.  $\rho$  grama po litri soli. Napisati program kojim se

određuje masena koncentracija soli u rastvoru ako se  $n$  puta odlije  $a$  litara rastvora i dolije  $a$  litara vode?

**Ulaz**

$l$  - zapremina rastvora u litrima ( $1 \leq l \leq 1000$ )

$\rho$  - maseni udeo soli u početnom rastvoru, u gramima po litru ( $0 \leq \rho \leq 400$ )

$a$  - zapremina (u litrima) koja se odliva-doliva ( $0 \leq a \leq l$ )

$n$  - broj odlivanja-dolivanja ( $0 \leq n \leq 10$ )

**Izlaz**

Masena koncentracija soli u rastvoru nakon  $n$  odlivanja-dolivanja (realan broj na dve decimale).

**Primer**

**Ulaz**

10

100

1

10

**Izlaz**

34.87

**1.1.6.8. Kredit**

Jovana je uzela kredit iz banke u iznosu od  $K$  evra. Godišnja kamatna stopa na ovaj kredit je  $g_k$ , ali se kamata zaračunava nakon svakog meseca na iznos preostalog duga, kada se i vrši uplata rate u visini  $r$  evra. Ako se kredit izmiruje u  $N$  jednakih mesečnih rata, napiši program koji određuje koliki treba da bude iznos mesečne rate.

**Ulaz**

Sa standardnog ulaza unose se sledeći brojevi, svaki u posebnom redu:

realni broj  $K$  - uzeti iznos kredita tj. glavnica ( $100 \leq K \leq 50000$ )

realni broj  $g_k$  - godišnja kamatna stopa u procentima ( $0 \leq g_k \leq 30$ )

ceo broj  $N$  - broj mesečnih rata ( $1 \leq N \leq 360$ )

**Izlaz**

Na standardni izlaz ispisati jedan realan broj - iznos mesečne rate  $r$ . Rezultat ispisati sa tačnošću od jednog centa.

**Primer**

**Ulaz**

1000

12

2

**Izlaz**

507.51

Pošto je godišnja kamata 12% mesečna kamata je 1%. Nakon mesec dana Jovana banci duuguje glavniciu od 1000 evra i kamatu koja iznosi jedan posto od toga tj. 10 evra. Ujedno uplaćuje ratu koja je jednaka 507.51 evro i njen dug je u tom trenutku jednak 502.49 evra. Nakon drugog meseca se na taj iznos zaračunava kamata od 1% tako da dug u tom trenutku iznosi 507.51 evro. Uplatom druge rate ceo dug se izmiruje.

### **1.1.7. Zaokruživanje realnih brojeva**

#### **1.1.7.1. Pertla**

Od kanapa date dužine isecaju se pertle date dužine. Odrediti koliko pertli je moguće izrezati i koliko kanapa na kraju preostane.

##### **Izlaz**

Sa standardnog ulaza učitavaju se dva pozitivna realna broja. U prvom redu data je dužina kanapa, a u drugom data je dužina pertle.

##### **Primer**

Na standardni izlaz ispisati ceo broj koji predstavlja broj pertli koje je moguće iseći i u sledećem redu realni broj koji predstavlja dužinu preostalog kanapa.

##### **Ulaz**

50.5

10

##### **Izlaz**

5

0.5

#### **1.1.7.2. Točak**

Napiši program koji na osnovu prečnika točka bicikla, brzine kretanja bicikla i vremena koje se bicikl kreće određuje koliko puta se točak okrenuo (računajući i okret koji još nije potpuno završen).

##### **Ulaz**

Sa standardnog ulaza se učitavaju realni brojevi koji predstavljaju prečnik točka bicikla u cm, brzinu kretanja bicikla u ms i vreme kretanja bicikla u s.

##### **Izlaz**

Na standardni izlaz se zapisuje jedan ceo broj koji predstavlja broj započetih okreta bicikla.

##### **Primer**

##### **Ulaz**

60

5

5

**Izlaz**

14

**1.1.7.3. Kružna meta**

Data je kružna meta od  $n$  koncentričnih krugova poluprečnika od  $a$  cm do  $n \cdot a$  cm. Za pogodak u unutrašnjost najmanjeg kruga dobija se  $n$  poena, a za pogodak u unutrašnjost svakog sledećeg prstena dobija se po poen manje. Napisati program kojim se određuje koliko se dobija poena za pogodak u tačku  $A(x,y)$  unutar mete.

**Ulaz**

U prvom redu standardnog ulaza nalazi se broj krugova, ceo broj  $n$  ( $1 \leq n \leq 20$ ), u drugom širina jednog kruga, realan broj  $a$ , u trećem redu  $x$ -koordinata tačke koja je pogođena, a u drugom njena  $y$ -koordinata. Koordinate su realni brojevi, a tačka se sigurno nalazi unutar mete ( $\sqrt{x^2+y^2} < n \cdot a$ ).

**Izlaz**

Jedini red standardnog izlaza treba da sadrži osvojeni broj poena.

**Primer****Ulaz**

10

1.0

1.2

5.3

**Izlaz**

5

**1.1.7.4. Skaliranje**

Profesor je obećao učenicima da će broj poena koji su osvojili na kontrolnom zadatku proporcionalno uvećati tako da Pera koji je najbolje uradio i osvojio 95 poena dobije maksimalni broj od 100 poena. Međutim, ovom operacijom celi brojevi poena se pretvaraju u realne, moguće sa puno ili čak beskonačno decimala. Zato je profesor odlučio da uvećani broj poena zaokruži sa preciznošću od pola poena. Svaki broj se zaokružuje najbližem celom broju polovina poena (ako postoje dve mogućnosti, zaokružuje se na veću). Na primer, broj 87.368 poena se zaokružuje na 87.5, 54.211 poena na 54, a 73.75 se zaokružuje na 74 poena. Napisati program koji za uneti broj poena određuje njegovu vrednost zaokruženu na ovaj način.

**Ulaz**

Sa standardnog ulaza učitava se jedan ceo broj između 0 i 95 koji predstavlja broj poena učenika pre skaliranja.

**Izlaz**

Na standardni izlaz ispisati jedan realan broj koji predstavlja broj poena učenika nakon skaliranja.

**Primer 1****Ulaz**

95

**Izlaz**

100

**Primer 2****Ulaz**

84

**Izlaz**

88.5

**Primer 3****Ulaz**

0

**Izlaz**

0

**1.1.7.5. Dinari i pare**

Napiši program koji za dati realni broj dinara određuje odgovarajući ceo broj dinara i ceo broj para.

**Ulaz**

Sa standardnog ulaza se unosi jedan realan broj zaokružen na dve decimale koji predstavlja iznos novca.

**Izlaz**

Na standardni izlaz napisati dva cela broja, svaki u posebnom redu, koji predstavljaju broj dinara i broj para.

**Primer****Ulaz**

123.45

**Izlaz**

123

45

## 1.2. Celobrojno deljenje

### 1.2.1. Pojam div i mod

#### 1.2.1.1. Šahovska tabla broj crnih polja

Na šahovskoj tabli širine  $n$  i dužine  $m$ , gornje levo polje je bele boje. Napisati program kojim se određuje broj crnih polja na datoj šahovskoj tabli.

##### **Ulaz**

Na standardnom ulazu nalaze se dva prirodna broja  $n, m \leq 100$ .

##### **Izlaz**

Broj crnih polja na šahovskoj tabli.

##### **Primer**

##### **Ulaz**

5  
3

##### **Izlaz**

7

#### 1.2.1.2. Podela jabuka

Podela jabuka. Aca je dobio  $n$  jabuka i na jednake delove (po trećinu) podelio braći Branku, Canetu, i Dušanu. Preostale jabuke pojeo je sam. Branko je svoje jabuke podelio braći (Aci, Canetu i Dušanu), a što je preostalo pojeo je sam. Tako su postupili prvo Cane, pa onda i Dušan. Koliko jabuka posle ovakve podele ima svako od njih (zanemarujući one koje su pojedene).

##### **Ulaz**

Broj jabuka  $n$  koje je dobio Aca.

##### **Izlaz**

Broj jabuka koje je dobio Aca (prvi red), Branko (drugi red), Cane (treći red) i Dušan (četvrti red).

##### **Primer**

##### **Ulaz**

100

##### **Izlaz**

44  
33  
19  
0

### Objašnjenje

Aca je dobio 100 jabuka. Po 33 je dao Branku, Canetu i Dušanu, a jednu je pojeo sam, tako da su nakon prve podele Aca, Branko, Cane i Dušan imali redom 0, 33, 33, 33 jabuke. Nakon toga, Branko je podelio braći po 11 jabuka i ništa mu nije ostalo, tako da je broj jabuka nakon druge podele 11, 0, 44, 44. Nakon toga, Cane je delio jabuke tako što je braći dao po 14 jabuka, a dve je sam pojeo, tako da je nakon treće podele stanje 25, 14, 0, 58. Na kraju, Dušan je braći dao po 19 jabuka, a jednu preostalu je sam pojeo. Zato je na kraju stanje bilo 44, 33, 19, 0.

#### 1.2.1.3. Šahovsko polje

Na šahovskoj tabli  $8 \times 8$  šahovska polja se broje, počev od donjeg levog polja, kolona po kolona od A1 do H8. Napisati program kojim se određuje u kojoj koloni i kom redu je polje koje je n-to izbrojano i koje po redu će biti izabrano polje u datoj koloni a i redu b.

##### **Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $1 \leq n \leq 64$ , druga linija sadrži dva karaktera, prvo slovo od A do H koje predstavlja kolonu a, a zatim cifru od 1 do 8 koja predstavlja red b.

##### **Izlaz**

U prvoj liniji standardnog izlaza prikazati broj kolone (slovo od A do H) i broj reda (cifru od 1 do 8) polja šahovske table koje je n-to izbrojano. Druga linija standardnog izlaza sadrži redni broj polja koje se nalazi u koloni a i redu b.

### Primer

##### **Ulaz**

19  
F5

##### **Izlaz**

C3  
45

#### 1.2.1.4. Razlomak u mešoviti broj

Za date prirodne brojeve a i b napisati program kojim se dati nepravilni razlomak  $a/b$  prevodi u mešoviti broj  $n \frac{c}{b}$ , takav da važi da je  $c/b < 1$ .

##### **Ulaz**

U prvoj liniji standardnog ulaza nalazi se prirodan broj a koji predstavlja brojilac nepravog razlomka, a u drugoj liniji prirodan broj b različit od nule koji predstavlja imenilac razlomka ( $a \geq b$ ).



**Izlaz**

Prva i jedina linija standardnog izlaza sadrži mešoviti zapis razlomka, preciznije prirodan broj, brojilac i imenilac mešovitog broja međusobno odvojeni sa po jednom prazninom (blanko znakom).

**Primer****Ulaz**

23  
8

**Izlaz**

2 7 8

**1.2.2. Pozicioni zapis****1.2.2.1. Brojevi****1.2.2.1.1. Zbir cifara**

Napiši program koji učitava četvorocifreni broj i ispisuje zbir svih cifara tog broja.

**Ulaz**

U jednoj liniji standardnog ulaza nalazi se četvorocifreni broj.

**Izlaz**

Zbir cifara četvorocifrenog broja.

**Primer****Ulaz**

1234

**Izlaz**

10

**1.2.2.1.2. Brojanje ovaca**

Tri pastira Jovan, Dule i Stanoje broje stado od najviše 120 ovaca na kraju svakog dana, na prste desne ruke. Broje sva trojica na sledeći način:

Jovan broji jednu po jednu ovcu na prste desne ruke. Svaki put kad izbroji pet ovaca on kaže Duletu da izbroji još jedan prst na svojoj desnoj ruci, a Jovan krene brojanje od početka.

Dule broji jedan po jedan prst samo kad mu Jovan to kaže. Kada Dule izbroji do pet on kaže Stanoju da izbroji još jedan prst na svojoj desnoj ruci, a Dule krene brojanje od početka.

Stanoje broji jedan po jedan prst samo kad mu Dule to kaže.

Kada završe brojanje svaki od njih prstima desne ruke pokazuje broj između 0 i 4. Napisati program koji pastirima pomaže da na osnovu broja otvorenih prstiju sva tri pastira redom izračunaju koliko stado ima ovaca.

**Ulaz**

Na standardnom ulazi nalaze se 3 cela broja iz intervala  $[0,4]$  koja redom predstavljaju broj otvorenih prstiju desne ruke naših pastira Jovana, Duleta i Stanoja.

**Izlaz**

Na standardnom izlazu u jednoj liniji prikazati broj ovaca u stadu.

**Primer**

**Ulaz**

3  
1  
3

**Izlaz**

83

**1.2.2.1.3. Razmeni cifre**

Napiši program kojim se u datom prirodnom broju razmenjuje cifra jedinica i cifra stotina. Za broj sa manje od tri cifre smatramo da su nedostajuće cifre jednake 0.

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se prirodan broj.

**Izlaz**

Broj dobijen posle razmene cifre jedinica i cifre stotina.

**Primer**

**Ulaz**

2349

**Izlaz**

2943

**1.2.2.1.4. Izbaci cifru stotina**

Napiši program koji iz dekadnog zapisa broja izbacuje cifru stotina.

**Ulaz**

Sa standardnog ulaza unosi se jedan prirodan broj manji od milijardu.

**Izlaz**

Na standardni izlaz ispisuje se broj dobijen izbacivanjem cifre stotina iz dekadnog zapisa unetog broja.

**Primer****Ulaz**

123456

**Izlaz**

12356

**1.2.2.1.5. Oktalni brojevi**

U računarstvu se često koriste tzv. oktalni brojevi - brojevi zapisani u osnovi 8, korišćenjem samo cifara od 0 do 7. Napiši program koji vrši konverziju četvorocifrenih oktalnih brojeva u dekadne vrednosti i obratno.

**Ulaz**

Sa standardnog ulaza se učitavaju 4 oktalne cifre (svaka u posebnom redu) i nakon toga dekadno zapisani prirodni broj  $n$  ( $0 \leq n < 84$ ).

**Izlaz**

Na standardnom izlazu u prvoj liniji ispisati dekadnu vrednost broja formiranog od učitanih oktalnih cifara (sdesna nalevo), a u drugoj liniji četvorocifrenu oktalnu reprezentaciju broja  $n$ .

**Primer****Ulaz**

1

2

3

4

1234

**Izlaz**

668

2322

**1.2.2.2. Vreme****1.2.2.2.1. Trkači**

Dva trkača trče po kružnoj stazi dužine  $s$  km, jedan brzinom od  $v_1$  km/h, drugi brzinom od  $v_2$  km/h. Koliko će vremena trebati bržem trkaču da za ceo krug prestigne sporijeg?

**Ulaz**

Podaci na dve decimale:

$s$  - dužina kružne staze ( $1 \leq s \leq 10$ )

$v_1$  - brzina prvog trkača ( $1 \leq v_1 \leq 45$ )

$v_2$  - brzina drugog trkača ( $1 \leq v_2 \leq 45$ )

**Izlaz**

Celobrojne vrednosti broja sati, minuta i sekundi, koje predstavljaju vreme potrebno da sporiji trkač za ceo krug sustigne sporijeg. Sekunde prikazati zaokružene na najbližu celobrojnu vrednost.

**Primer****Ulaz**

10.00

5.00

6.00

**Izlaz**

10

0

0

**1.2.2.2.2. Ponoć**

Napiši program koji za dato vreme u toku dana određuje koliko je sekundi proteklo od prethodne ponoći i koliko je sati, minuta i sekundi ostalo do sledeće ponoći.

**Ulaz**

Sa standardnog ulaza unose se tri cela broja koji određuju jedan vremenski trenutak:

h - sat ( $0 \leq h < 24$ )

m - minut ( $0 \leq m < 60$ )

s - sekund ( $0 \leq s < 60$ )

**Izlaz**

Na standardni izlaz ispisati dve linije. U prvoj ispisati ceo broj koji predstavlja broj sekundi proteklih od prethodne ponoći. U drugoj ispisati tri cela broja razdvojena dvotačkama - broj sati  $h_p$  ( $0 \leq h_p < 24$ ), broj minuta  $m_p$  ( $0 \leq m_p < 60$ ) i broj sekundi  $s_p$  ( $0 \leq s_p < 60$ ) do naredne ponoći.

**Primer****Ulaz**

10

23

52

**Izlaz**

37432

13:36:8

**1.2.2.2.3. Trajanje vožnje**

Poznati su sat, minut i sekund početka i kraja vožnje autobusom (vožnja počinje i završava se u jednom danu). Napisati program koji određuje koliko sati, minuta i sekundi je trajala ta vožnja.

**Ulaz**

Sa standardnog ulaza učitava se 6 brojeva (svaki u zasebnom redu). Prvo sat, minut i sekund početka vožnje, a zatim sat, minut i sekund kraja vožnje.

**Izlaz**

Na standardni izlaz se ispisuje jedan red u kome su tri broja (sat, minut i sekund) razdvojena dvotačkom.

**Primer****Ulaz**

2  
59  
8  
3  
1  
14

**Izlaz**

0:2:6

**1.2.2.2.4. Trajanje vožnje kroz dva dana**

Poznati su sat, minut i sekund početka i kraja vožnje autobusom (moguće je i da je vožnja počela u jednom, a završila se u narednom danu, ali se zna da je trajala manje od 24 sata). Napisati program koji određuje koliko sati, minuta i sekundi je trajala ta vožnja.

**Ulaz**

Sa standardnog ulaza učitava se 6 brojeva (svaki u zasebnom redu). Prvo sat, minut i sekund početka vožnje, a zatim sat, minut i sekund kraja vožnje.

**Izlaz**

Na standardni izlaz se ispisuje jedan red u kome su tri broja (sat, minut i sekund) razdvojena zapetom.

**Primer****Ulaz**

23 59 59  
0 0 1

**Izlaz**

0:0:2

**Ulaz**0 0 1  
23 59 59**Izlaz**

23:59:58

**1.2.2.2.5. Autobus**

Ako je poznato trajanje putovanja i vreme kada autobus treba da stigne na odredište, napiši program koji izračunava kada autobus treba da pođe sa polazišta (polazak može biti u prethodnom danu u odnosu na dolazak).

**Ulaz**

Sa standardnog ulaza učitava se četiri cela broja:  
trajankeh ( $0 \leq \text{trajankeh} < 12$ ) i trajanjem ( $0 \leq \text{trajanjem} < 60$ ) - sati i minuti  
trajanja vožnje  
dolazakh ( $0 \leq \text{dolazakh} < 24$ ) i dolazakm ( $0 \leq \text{dolazakm} < 60$ ) - sat i minut dolaska

**Izlaz**

Na standardni izlaz ispisati dva cela broja razdvojena dvotačkom:  
polazakh ( $0 \leq \text{polazakh} < 24$ ) i polazakm ( $0 \leq \text{polazakm} < 60$ ) - sat i minut polaska

**Primer 1****Ulaz**2  
30  
13  
0**Izlaz**

10:30

**Primer 2****Ulaz**12  
0  
8  
0**Izlaz**

20:0

**1.2.2.2.6. Vreme završetka filma**

Poznati su sat i minut početka filma i dužina trajanja filma u minutima.  
Napisati program koji određuje sat i minut završetka filma.

**Ulaz**

Sa standardnog ulaza učitava se 3 prirodna broja (svaki u zasebnom redu). Prvo sat i minut početka filma, a zatim dužina trajanja filma u minutama.

**Izlaz**

Na standardni izlaz se ispisuje jedan red u kome su dva broja (sat, minut) razdvojena dvotačkom koji predstavljaju vreme završetka filma.

**Primer****Ulaz**

23  
15  
115

**Izlaz**

1:10

**1.2.2.2.7. Čekanje**

Dva stara drugara su dogovorila da se nađu u centru grada. Poznato je vreme (u obliku sata, minuta i sekunde) kada je svako od njih došao na sastanak (oba vremena su u jednom danu). Napiši program koji određuje koliko dugo je onaj koji je prvi stigao čekao onog drugog.

**Ulaz**

Sa standardnog ulaza se unosi šest celih brojeva (svaki je zadat u posebnom redu). Sat (između 0 i 23), minut (između 0 i 59) i sekund (između 0 i 59) dolaska prvog čoveka i sat, minut i sekund dolaska drugog čoveka. Podaci predstavljaju ispravno zadate vremenske trenutke u okviru jednog dana.

**Izlaz**

Na standardni izlaz ispisati broj sati, minuta i sekundi koliko je onaj drugar koji je prvi došao čekao drugog. Brojeve ispisati u jednom redu, razdvojene dvotačkom.

**Primer****Ulaz**

3  
45  
20  
7  
23  
50

**Izlaz**

3:38:30

**1.2.2.2.8. UNIX vreme**

Od operativnog sistema UNIX vreme u računarima se izražava kao broj sekundi proteklih od početka epohe tj. od 1. januara 1970. godine. Po uzoru na to, osmislili smo sistem merenja vremena u kome se vreme izražava u broju milisekundi proteklih od uključivanja računara. U nekom trenutku, na računaru je puštena pesma. Ako je poznato vreme od trenutka uključivanja računara kada je pesma puštena i dužina trajanja pesme u milisekundama, napiši program koji određuje kada je pesma završena.

**Ulaz**

Sa standardnog ulaza učitavaju se sledeći celi brojevi (svaki u posebnom redu):

dan ( $0 \leq \text{dan} \leq 10$ ), sat ( $0 \leq \text{sat} < 24$ ), min ( $0 \leq \text{min} < 60$ ), sek ( $0 \leq \text{sek} < 60$ ), mili ( $0 \leq \text{mili} < 1000$ ) - broj dana, sati, minuta, sekundi i milisekundi proteklih od trenutka uključivanja računara do puštanja pesme.

trajanje ( $0 \leq \text{trajanje} \leq 1000000$ ) - broj milisekundi koliko traje pesma.

**Izlaz**

Na standardni izlaz ispisati sledeće cele brojeve, razdvojene dvotačkama: dan ( $0 \leq \text{dan} \leq 100$ ), sat ( $0 \leq \text{sat} < 24$ ), min ( $0 \leq \text{min} < 60$ ), sek ( $0 \leq \text{sek} < 60$ ), mili ( $0 \leq \text{mili} < 1000$ ) - broj dana, sati, minuta, sekundi i milisekundi proteklih od trenutka uključivanja računara do završetka pesme.

**Primer 1****Ulaz**

3  
10  
15  
23  
843  
100000

**Izlaz**

3:10:17:3:843

**Primer 2****Ulaz**

4  
23  
59  
59  
517  
12345

**Izlaz**

5:0:0:11:862



**Primer 3****Ulaz**

10  
23  
59  
59  
999  
1000000

**Izlaz**

11:0:16:39:999

**1.2.2.2.9. Tajmer**

Tajmer se u trenutku h:m:s (h od 0 do 23, m i s od 0 do 59) naštima da zvoni nakon perioda koji se iskazuje sa hh:mm:ss ( $0 \leq hh, mm, ss \leq 1000$ ). Može se zadati 100:100:100 što je isto kao da je zadato 101:41:40. Odrediti kada će tajmer da zvoni u formatu h:m:s+dan (dan uzima vrednosti počev od 0).

**Ulaz**

Prvo startno vreme - u svakom redu po jedan podatak: sat, minut i sekund.  
Zatim trajanje perioda - u svakom redu po jedan podatak: broj sati, minuta i sekundi.

**Izlaz**

Sat, minut, sekund i dan u formatu h:m:s+dan.

**Primer****Ulaz**

23  
59  
59  
0  
0  
1

**Izlaz**

0:0:0+1

**Ulaz**

1  
1  
1  
48  
0  
0

**Izlaz**

1:1:1+2

### 1.2.2.3. Uglovi

#### 1.2.2.3.1. Radijani

U matematici se uglovi često mere brojem radijana. Radijan je onaj ugao čija je dužina luka jednaka poluprečniku. Napiši program koji za dati broj radijana izračuna najbliži ugao u stepenima, minutima i sekundama.

##### **Ulaz**

Sa standardnog ulaza unosi se pozitivan realan broj radijana.

##### **Izlaz**

Na standardni izlaz ispisati tri cela broja stepeni ( $0 \leq \text{stepeni} < 360$ ), minuti ( $0 \leq \text{minuti} < 60$ ) i sekundi ( $0 \leq \text{sekundi} < 360$ ), razdvojena dvotačkama, koji predstavljaju ugao u stepenima, minutima i sekundama najbliži datom uglu u radijanima.

#### **Primer 1**

##### **Ulaz**

1.57

##### **Izlaz**

89:57:16

#### **Primer 2**

##### **Ulaz**

1000

##### **Izlaz**

55:46:46

#### 1.2.2.3.2. Gps

Marija je našla geografske koordinate na kojima se nalazi zakopano blago, međutim njen GPS uređaj podržava geografske koordinate unete kao decimalni broj stepeni, dok su koordinate koje ona ima date u broju stepeni, minuta i sekundi. Napiši program koji će Mariji pomoći da prevede koordinate.

##### **Ulaz**

Sa standardnog ulaza unose se tri cela broja:

st ( $0 \leq \text{st} < 360$ ) - broj stepeni

min ( $0 \leq \text{min} < 60$ ) - broj minuta

sek ( $0 \leq \text{sek} < 60$ ) - broj sekundi

##### **Izlaz**

Na standardni izlaz ispisati jedan realan broj koji predstavlja ugao u stepenima (tolerancija greške je  $10^{-5}$ ).

**Primer*****Ulaz***

18

19

20

***Izlaz***

18.32222

**1.2.2.3.3. Točak sreće**

Točak sreće okrenemo za određeni ugao, pa ga ponovo okrenemo u istom smeru za neki drugi ugao. Točak je takav da se u jednom okretu ne može okrenuti više od 20 punih krugova. Napisati program kojim se određuje najmanji ugao za koji treba točak još jednom da okrenemo u istom smeru da bi se došao u početni položaj.

***Ulaz***

Sa standardnog ulaza učitava se 6 brojeva (svaki u zasebnom redu). Prvo sat minut i sekund prvog ugla, a zatim sat, minut i sekund drugog ugla za koji se točak okrene.

***Izlaz***

Na standardni izlaz se ispisuje jedan red u kome su tri broja (sat, minut i sekund) razdvojena prazninom.

**Primer*****Ulaz***

120

50

20

130

50

40

***Izlaz***

108 19 0

**1.2.2.3.4. Ugao satne kazaljke**

Ako je dat ugao koji satna kazaljka zaklapa sa vertikalnom polupravom koja spaja centar i vrh sata (broj 12), odrediti vreme u minutama i sekundama.

***Ulaz***

Sa standardnog ulaza učitava se jedan ceo broj koji predstavlja ugao u stepenima.

**Izlaz**

Na standardni izlaz ispisati dva cela broja  $h$  ( $0 \leq h < 12$ ) i  $m$  ( $0 \leq m < 60$ ) koji predstavljaju broj sati i minuta koji su najbliži položaju satne kazaljke.

**Primer****Ulaz**

90

**Izlaz**

3:0

**Primer 2****Ulaz**

200

**Izlaz**

6:40

**1.2.2.3.5. Ugao između kazaljki**

Odrediti ugao u stepenima i minutima između satne i minutne kazaljke ako je zadato vreme u satima i minutima. Ugao je uvek pozitivan i meri se u smeru kretanja kazaljki na satu (zato može da bude i veći od 180 stepeni).

**Ulaz**

Učitavaju se dva broja, svaki u posebnoj liniji. U prvoj liniji je broj sati (između 0 i 12), a u drugoj liniji je broj minuta (između 0 i 59).

**Izlaz**

Stepen ugla 0 do 359, i minut ugla od 0 do 59, razdvojeni dvotačkom tj. karakterom :.

**Primer 1****Ulaz**

1  
30

**Izlaz**

135:0

**Primer 2****Ulaz**

12  
30

**Izlaz**

165:0

**1.2.2.3.6. Poklapanje kazaljki**

Ako sat pokazuje tačno  $x$  sati, napisati program koji određuje posle koliko minuta će se prvi put poklopiti velika i mala kazaljka.

**Ulaz**

Broj sati od 0 do 11.

**Izlaz**

U minutima zaokruženim na najbližu celobrojnu vrednost od 0 do 60.

**Primer 1**

**Ulaz**

2

**Izlaz**

11

**Primer 2**

**Ulaz**

11

**Izlaz**

60

**1.2.3. Modularna aritmetika****1.2.3.1. Operacije po modulu**

Napiši program koji određuje poslednje tri cifre zbira i poslednje tri cifre proizvoda četiri uneta cela broja.

**Ulaz**

U svakom redu standardnog ulaza unosi se po jedan ceo broj iz intervala  $[1..999]$ .

**Izlaz**

Na standardni izlaz se ispisuju broj određen sa poslednje tri cifre zbira i broj određen sa poslednje tri cifre proizvoda unetih brojeva (eventualne vodeće nule se ne moraju ispisati).

**Primer**

**Ulaz**

999

999

999

999

*Izlaz*

996

1

**1.2.3.2. Monopol**

Monopol je igra u kojoj se igrači kreću po poljima koja su postavljena u krug. Igrači se uvek kreću u smeru kazaljke na satu. Pretpostavimo da su polja označena rednim brojevima koji kreću od 0, da su na početku igre oba igrača na tom polju i da se tokom igre igrači nisu kretali unatrag. Ako se zna broj polja koje je prvi igrač prešao od početka igre i broj polja koje je drugi igrač prešao od početka igre napiši program koji izračunava koliko koraka prvi igrač treba da napravi da bi došao na polje na kom se nalazi drugi igrač.

*Ulaz*

Sa standardnog ulaza se unose tri prirodna broja. U prvoj liniji dat je broj polja na tabli, u drugoj broj polja koje je od početka igre prešao prvi, a u trećoj broj polja koje je od početka igre prešao drugi igrač.

*Izlaz*

Na standardni izlaz treba ispisati koliko koraka unapred igrač treba da napravi da bi stigao na željeno polje.

**Primer 1***Ulaz*

10

3

7

*Izlaz*

4

**Primer 2***Ulaz*

10

7

3

*Izlaz*

6

**1.2.3.3. Razbrajalica**

Deca stoje u krugu i određuju ko će od njih da žmuri tako što izgovaraju neku razbrajalicu (na primer, eci-peci-pec). Ako decu obeležimo brojevima od 0 do  $n-1$ , redom kako stoje u krugu, ako je poznat broj reči (slogova) razbrajalice i ako je poznat redni broj deteta od kojeg počinje razbrajanje, napiši program

koji prikazuje redni broj deteta koje je odabrano da žmuri (tj. na kome se razbrajanje završava).

**Ulaz**

Unose se tri prirodna broja. Broj slogova razbrajalice, broj dece i redni broj deteta od kojeg počinje razbrajanje. Pretpostavićemo da je broj dece manji od 100, a da je broj slogova manji od 1000.

**Izlaz**

Ispisuje se jedan prirodni broj -- redni broj deteta odabranog da žmuri.

**Primer**

Pretpostavimo da sedmoro dece u krugu koristi razbrajalicu eci-peci-pec, a da brojanje kreće od deteta broj 3, razbrajanje će teći ovako:

eci-peci-pec-jasam-mali-zec-tisi-mala-prepe-lica-eci-peci-pec

3 4 5 6 0 1 2 3 4 5 6 0 1

**Ulaz**

13

7

3

**Izlaz**

1

### 1.2.3.4. Žmurke

Dete broji u igri žmurke 5,10,15,...,100, ali pošto ne zna brojeve veće od 100, posle 100 ono kreće da broji iz početka. Koji je broj rekao u n-tom koraku?

**Ulaz**

Sa standardnog ulaza učitava se jedan pozitivan ceo broj n.

**Izlaz**

Na standardni izlaz ispisuje se broj koji dete izgovara u n-tom koraku.

**Primer**

**Ulaz**

5

**Izlaz**

25

**Ulaz**

1234

**Izlaz**

70

### 1.2.3.5. Sat

Pretpostavimo da prilikom iskazivanja vremena za sate koristimo samo brojeve od 1 do 12. Tako za 17:45 kažemo 5:45, a za 0:32 kažemo da je 12:32. Ako je poznat broj sati i minuta takav da je broj sati u opsegu od 0 do 23, a minuta od 0 do 59, iskazati to vreme tako da je broj sati u opsegu 1 do 12.

#### **Ulaz**

Učitavaju se dva broja, svaki u posebnoj liniji. U prvoj liniji je broj sati (između 0 i 23), a u drugoj liniji je broj minuta (između 0 i 59).

#### **Izlaz**

Ispisuje se jedna linija u kojoj se prikazuje vremenski trenutak u formatu h:m, gde je broj sati h između 1 i 12, a broj minuta između 1 i 59.

### **Primer**

#### **Ulaz**

0  
32

#### **Izlaz**

12:32

#### **Ulaz**

5  
35

#### **Izlaz**

5:35

### 1.2.3.6. Zbir brojeva po modulu

Napiši program koji izračunava zbir prirodnih brojeva od 1 do n po datom modulu m.

#### **Ulaz**

Sa standardnog ulaza se unose dva cela broja, svaki u posebnom redu:  
n ( $10^{**}2 \leq n \leq 10^{**}9$ )  
m ( $2 \leq m \leq 100$ )

#### **Izlaz**

Na standardni izlaz ispisati jedan ceo broj koji predstavlja traženi zbir po modulu.

### **Primer**

#### **Ulaz**

100  
100



**Izlaz**

50

Zbir svih brojeva od 1 do 100 je 5050 i ostatak pri deljenju sa 100 je 50.

**1.2.3.7. Navijanje sata**

Dočepo se mali brata oćevoga sata. Pošto je još mali, zna da gleda samo satnu kazaljku. Sat se može navijati tako što se jednim dugmetom satna kazaljka pomera za jedan sat unapred (u smeru kretanja kazaljke na satu) a drugim dugmetom se pomera za jedan sat unazad (u suprotnom smeru kretanja kazaljke na satu). Ako se zna pozicija  $a$  na kojoj se trenutno nalazi kazaljka, pozicija  $b$  na kojoj brata želi da se nađe, napiši program koji određuje da li se sat može naviti tako što brata pritisne jedno od dva dugmeta tačno  $k$  puta.

**Ulaz**

Sa standardnog ulaza učitavaju se brojevi  $0 \leq a, b \leq 11$  i broj  $0 \leq k \leq 1000$ .

**Izlaz**

Na standardni izlaz napisati tekst napred ako se sat može naviti tako što se kazaljka pomera unapred, nazad ako se sat može naviti tako što se kazaljka pomera unazad (ako je sat moguće naviti na oba načina, ispisati napred, a zatim nazad u sledećem redu) tj. ne može ako se sat ne može naviti.

**Primer 1****Ulaz**

1

7

18

**Izlaz**

napred

nazad

**Primer 2****Ulaz**

2

4

10

**Izlaz**

nazad

**Primer 3****Ulaz**

7

9

3

**Izlaz**

ne moze

### **1.2.4. Celobrojno zaokruživanje**

#### **1.2.4.1. Pokloni**

Svaki poklon sadrži tačno  $k$  čokoladica. Ako na raspolaganju imamo ukupno  $n$  čokoladica, koliko poklona je moguće zapakovati?

**Ulaz**

Sa standardnog ulaza se učitavaju dva cela broja (svaki u posebnom redu):

$k$  ( $1 \leq k \leq 20$ ) - broj čokoladica u jednom poklonu

$n$  ( $0 \leq n \leq 1000$ ) - ukupan broj čokoladica

**Izlaz**

Na standardni izlaz ispisati jedan ceo broj - najveći broj poklona koje je moguće napraviti.

#### **Primer**

**Ulaz**

4

19

**Izlaz**

4

#### **1.2.4.2. Lift**

U jednom hotelu  $n$  ljudi čeka ispred lifta. U lift može da stane njih  $k$ . Koliko je najmanje vožnji liftom potrebno da se svi popnu u svoje sobe?

**Ulaz**

Sa standardnog ulaza učitavaju se dva cela broja, svaki u posebnom redu:

ukupan broj ljudi ispred lifta  $n$  ( $0 \leq n \leq 200$ )

broj ljudi koji mogu da odjednom stanu u lift  $k$  ( $1 \leq k \leq 10$ )

**Izlaz**

Na standardni izlaz ispisati jedan ceo broj - najmanji potreban broj vožnji liftom.

#### **Primer 1**

**Ulaz**

18

3

**Izlaz**

6

**Primer 2****Ulaz**

18

4

**Izlaz**

5

**1.2.4.3. Papir**

Napisati program koji izračunava masu određenog broja odštampanih knjiga, da bi izdavačka kuća znala koju nosivost kamiona treba da naruči.

**Ulaz**

Prirodan broj brojKnjiga( $1 \leq \text{brojKnjiga} \leq 1000$ ), broj odštampanih primeraka knjige.

Prirodan broj brojStranica( $30 \leq \text{brojStranica} \leq 500$ ), broj strana jedne knjige;

Prirodan broj duzinaStranice( $50 \leq \text{duzinaStranice} \leq 297$ ), dužina lista tj. strane zadata u milimetrima;

Prirodan broj sirinaStranice( $50 \leq \text{sirinaStranice} \leq 210$ ), širina lista tj. strane zadata u milimetrima;

Realan broj masaPapira( $50 \leq \text{masaPapira} \leq 100$ ), masa papira u gramima po kvadratnom metru.

Realan broj masaKorica( $10 \leq \text{masaKorica} \leq 100$ ), masa korica jedne knjige u gramima;

**Izlaz**

Realan broj koji predstavlja ukupnu masu knjiga u kilogramima, zaokružen na 3 decimale.

**Primer****Ulaz**

850

193

260

200

80.0

25.0

**Izlaz**

364.242

Potrebno je odštampati 850 primeraka knjige koja ima 193 strane. Jedan list papira ima dve strane, tako da se knjiga sastoji od 97 listova (na poslednjem listu odštampana je samo jedna strana) i korica. Masa jednog lista papira je 4.16 grama. Težina jedne knjige je zato jednaka zbiru mase listova koja je jednaka 403.52 grama i mase korica koja je 25 grama, tj. jedna knjiga je teška 428.52 grama. Zato je ukupna masa jednaka 364.242 kilograma.

#### 1.2.4.4. Planinari

Planinari se penju na vrh planine čija je visina zadata u metrima. Danju se mogu popeti prema gore izvestan broj metara, ali tokom noći spavaju, pa se moraju spustiti za određen broj metara. Napišite program koji će odrediti koliko dana treba planinarima da se popnu do vrha planine.

##### **Ulaz**

U prva tri reda standardnog ulaza nalaze se prirodni brojevi  $G, D, H$  ( $1 \leq D < G \leq H \leq 10^9$ ) koji redom označavaju:

$G$  - visinu metara koju planinari pređu danju penjući se naviše,

$D$  - visinu metara koju se planinari spuste noću,

$H$  - visina planine.

##### **Izlaz**

U prvi i jedini red standardnog izlaza ispišite broj dana koji će proći dok planinari ne stignu do vrha.

#### **Primer**

##### **Ulaz**

5  
1  
6

##### **Izlaz**

2

#### 1.2.4.5. Kruške

Učiteljica je kupila  $k$  krušaka, koje želi da podeli učenicima iz svog odeljenja. Ako u odeljenju ima  $u$  učenika, koliko još najmanje krušaka učiteljica treba da kupi da bi bila sigurna da će svakom detetu moći da da isti broj krušaka.

##### **Ulaz**

Sa standardnog ulaza unose se dva prirodna broja (svaki u posebnom redu):  
 $k$  ( $0 \leq k \leq 300$ ) broj krušaka koje učiteljica trenutno ima.  
 $u$  broj učenika u odeljenju ( $15 \leq u \leq 30$ ).

##### **Izlaz**

Na standardni izlaz ispisati jedan ceo broj koji predstavlja najmanji broj krušaka koje učiteljica treba da kupi.

#### **Primer 1**

##### **Ulaz**

73  
23

##### **Izlaz**

19

**Primer 2****Ulaz**

100

20

**Izlaz**

0

**1.2.4.6. Kofa**

U dečijem bazenu postoji gusarski brod na čijem se vrhu nalazi kofa koja se prazni na svakih  $k$  minuta. Ako se zna broj minuta proteklih od nekog ranijeg pražnjenja kofe napiši program koji određuje broj minuta preostalih do prvog sledećeg minuta u kome će se kofa sigurno isprazniti (0 ako će se kofa isprazniti u trenutnom minutu).

**Ulaz**

U prvom redu standardnog ulaza nalazi se ceo pozitivan broj  $k$  ( $0 < k \leq 60$ ) koji predstavlja broj minuta između svaka dva uzastopna pražnjenja kofe, a u drugom broj  $n$  ( $0 \leq n \leq 240$ ) koji predstavlja broj minuta od nekog prethodnog pražnjenja kofe.

**Izlaz**

Na standardni izlaz ispisati traženi broj.

**Primer****Ulaz**

5

23

**Izlaz**

2

**Ulaz**

5

25

**Izlaz**

0

**1.2.4.7. Deljivi oko broja**

Dati su celi brojevi  $n$  i  $k$ . Napiši program koji određuje najveći ceo broj  $l \leq n$  deljiv brojem  $k$  i najmanji ceo broj  $d \geq n$  deljiv brojem  $k$ .

**Ulaz**

Sa standardnog ulaza unose se dva cela broja (svaki u posebnom redu).

$n$  ( $0 \leq n \leq 100000$ )

$k$  ( $1 \leq k \leq 10000$ )

**Izlaz**

Na standardni izlaz ispisati dva tražena cela broja, svaki u posebnom redu.

**Primer 1****Ulaz**

23

5

**Izlaz**

20

25

**Primer 2****Ulaz**

49

7

**Izlaz**

49

49

**1.2.4.8. Sečenje pločica**

Pravougaonu terasu dimenzija  $d \times s$  centimetara kvadratnih treba popločati korišćenjem pločica kvadratnog oblika stranice  $p$  centimetara, koje se postavljaju tako da su im stranice paralelne stranicama terase. Napisati program kojim se određuje koliko se pločica mora seći radi popločavanja, kao i površinu dela terase koji zauzimaju sečene pločice. Od svake sečene pločice koristi se jedan deo, a drugi odbacuje.

**Ulaz**

$p$  - stranica pločice u cm ( $10 \leq p \leq 50$ )

$d$  - dužina prostorije u cm ( $200 \leq d \leq 10000$ )

$s$  - širina prostorije u cm ( $200 \leq s \leq 10000$ )

**Izlaz**

Broj pločica koje se moraju seći i površina koju pokrivaju sečene pločice.

**Primer****Ulaz**

20

310

270

**Izlaz**

29

5700

#### 1.2.4.9. Broj deljivih u intervalu

Napiši program koji određuje koliko u intervalu  $[a,b]$  postoji brojeva deljivih brojem  $k$ .

##### **Ulaz**

Sa standardnog ulaza unose se tri cela broja, svaki u posebnom redu.

$a$  ( $0 \leq a \leq 10^{**9}$ )

$b$  ( $a \leq b \leq 10^{**9}$ )

$k$  ( $1 \leq k \leq 10^{**9}$ )

##### **Izlaz**

Na standardni izlaz ispisati traženi ceo broj.

##### **Primer**

##### **Ulaz**

30

53

5

##### **Izlaz**

5

Brojevi su 30, 35, 40, 45 i 50.

#### 1.2.4.10. Sudoku

Mirko je mali programer koji pokušava da isprogramira igricu sudoku. Blizu je da završi, ali mu je potrebna mala pomoć. Smislio je da korisnik mišem bira kvadrat u koji će upisati broj. Polje se sastoji od 81 kvadratića raspoređenog u 9 vrsta i 9 kolona i organizovanog u 9 većih kvadrata (kao na slici). Svaki kvadratić je dimenzije 30 puta 30 piksela (ukupno polje je dimenzije 270 puta 270 piksela). Poznat je položaj piksela na koji je kliknuto mišem. Položaj je određen rednim brojevima (koordinatama) tog piksela po horizontali i po vertikali, računajući od donjeg levog ugla polja (pikseli se i po horizontali i po vertikali broje od 1 do 270). Potrebno je ispisati redni broj vrste, kolone i većeg kvadrata u kojem se nalazi piksel na koji je kliknuto (vrste se broje od 1 do 9 odozdo naviše, kolone od 1 do 9 sleva nadesno, a kvadrati po vrstama od donjeg levog ugla, kako je obeleženo na slici).

9								
8		7			8			9
7								
6								
5		4			5			6
4								
3								
2		1			2			3
1								
	1	2	3	4	5	6	7	8

**Ulaz**

Sa ulaza se učitavaju dva cela broja  $x$  i  $y$  ( $1 \leq x, y \leq 270$ ) koje predstavljaju koordinate piksela na koji je kliknuto.

**Izlaz**

Na standardni izlaz ispisati tri broja, svaki posebnom redu: redni broj vrste (od 1 do 9), redni broj kolone (od 1 do 9) i redni broj kvadrata (od 1 do 9).

**Primer 1****Ulaz**

128

230

**Izlaz**

8

5

8

**Primer 2****Ulaz**

180

180

**Izlaz**

6

6

5

**Primer 3****Ulaz**

181

181



**Izlaz**7  
7  
9

## 1.3. Tipovi

### 1.3.1. Celobrojni

#### 1.3.1.1. Sredina intervala

Na planeti M u ulici M žive mravi. Sve kućice u ulici su označene prirodnim brojevima. Mravica Ana živi u kućici broj  $a$ , a mrav Boris iza nje, u kućici broj  $b$ . Mrav Viktor živi u kućici tačno na sredini između Ane i Borisa. Napiši program koji određuje broj kućice u kojoj živi Viktor (ako je između Ane i Borisa paran broj kućica, Viktor živi u onoj koja je bliža Ani).

**Ulaz**

Sa standardnog ulaza unose se dva cela broja:

 $a$  ( $0 \leq a \leq 2 \cdot 10^9$ ) - broj kućice u kojoj živi Ana $b$  ( $a+2 \leq b \leq 2 \cdot 10^9$ ) - broj kućice u kojoj živi Boris**Izlaz**

Na standardni izlaz ispisati jedan ceo broj - broj kućice u kojoj živi Viktor.

#### Primer 1

**Ulaz**

432457

562321

**Izlaz**

497389

#### Primer 2

**Ulaz**

2

9

**Izlaz**

5

#### Primer 3

**Ulaz**

2000000000

1999999998

**Izlaz**

1999999999

**1.3.1.2. Faktorijel 64bit**

Napiši program koji izračunava vrednosti faktorijela datog celog broja.

**Ulaz**

Sa standardnog ulaza unosi se broj  $n$  ( $1 \leq n \leq 20$ ).

**Izlaz**

Na standardni izlaz ispisuje se jedan broj  $n!$ .

**Primer****Ulaz**

20

**Izlaz**

2432902008176640000

**1.3.1.3. Jabuke u trouglu**

Jabuke su poslagane u trougao sledećeg oblika:

\*

\*\*

\*\*\*

\*\*\*\*

Napiši program koji određuje ukupan broj jabuka u trouglu.

**Ulaz**

Sa standardnog ulaza unosi se ceo broj  $n$  ( $n \leq 6 \cdot 10^{**9}$ ) koji određuje broj redova trougla.

**Izlaz**

Na standardni izlaz ispisati jedan ceo broj koji predstavlja ukupan broj jabuka u trouglu.

**Primer 1****Ulaz**

3500000000

**Izlaz**

6125000001750000000

**Primer 2****Ulaz**

5000000000

**Izlaz**

12500000002500000000

**1.3.1.4. Množenje velikih brojeva**

Žuti mravi žele da odrede površinu livade po kojoj se kreću. Pošto su oni jako mali, dužina i širina livade su izraženi u mikrometrima.

**Ulaz**

Sa standardnog ulaza se unose dva cela broja  $a$  i  $b$  ( $0 \leq a, b \leq 4 \cdot 10^{**9}$ ) koji predstavljaju dimenzije livade u mikrometrima.

**Izlaz**

Na standardni izlaz ispiši površinu livade u mikrometrima kvadratnim.

**Primer 1****Ulaz**

1234567890  
3216549870

**Izlaz**

3971049186085674300

**Primer 2****Ulaz**

4000000000  
4000000000

**Izlaz**

16000000000000000000

**1.3.2. Karakteri****1.3.2.1. Klasifikacija karaktera**

Napisati program koji za učitani ASCII karakter ispituje da li je malo slovo, veliko slovo, cifra ili nešto četvrto

**Ulaz**

U jednoj liniji standardnog ulaza nalazi se jedan karakter.

**Izlaz**

U jednoj liniji standardnog izlaza prikazati jednu od sledećih informacija: MALO SLOVO, VELIKO SLOVO, CIFRA, OSTALO.

**Primer 1***Ulaz*

a

*Izlaz*

MALO SLOVO

**Primer 2***Ulaz*

B

*Izlaz*

VELIKO SLOVO

**Primer 3***Ulaz*

7

*Izlaz*

CIFRA

**Primer 4***Ulaz*

:

*Izlaz*

OSTALO

**1.3.2.2. Transformacija karaktera**

Napisati program kojim se transformiše učitani karakter tako što se malo slovo prebacuje u veliko, veliko slovo u malo, a ostali karakteri se ne menjaju.

*Ulaz*

U prvoj liniji standardnog ulaza nalazi se jedan karakter.

*Izlaz*

Na standardnom izlazu prikazati transformisan karakter.

**Primer***Ulaz*

t

*Izlaz*

T

### 1.3.2.3. Abecedno ogledalo

Katarina je odлучila da svojoj drugarici pošalje šifrovanu poruku, koja sadrži samo slova engleske abecede, cifre i interpunkcijske znake. Svako slovo će šifrovati posebno na osnovu narednih pravila. Mala slova se šifruju velikim slovima tako što se slovo a šifruje slovom Z, slovo b šifruje slovom Y, c slovom X itd., sve do slova y koje se šifruje slovom B i z koje se šifruje slovom A. Velika slova se šifruju potpuno analogno - od A koje se šifruje sa z do Z koje se šifruje sa a. Ostali karakteri se ne menjaju.

#### **Ulaz**

Sa standardnog ulaza unosi se jedna linija teksta, završena karakterom tačka (karakterom .).

#### **Izlaz**

Na standardni izlaz ispisati šifrovani tekst (bez karaktera tačka).

### **Primer**

#### **Ulaz**

Zdravo svima.

#### **Izlaz**

aWIZEL HERNZ

## 2. Grananje

### 2.1. Jednostavno grananje

#### 2.1.1. relacijski operatori

##### 2.1.1.1. Jabuke

Pera i Mika su brali jabuke. Pera je ubrao  $p$ , a Mika  $m$  jabuka. Napiši program koji proverava da li je Pera uspeo da nabere više jabuka nego Mika.

##### *Ulaz*

Sa standardnog ulaza unose se dva prirodna broja (svaki u posebnom redu). U prvom redu broj jabuka koje je ubrao Pera, a u drugom broj jabuka koje je ubrao Mika.

##### *Izlaz*

Ispisati na standardnom izlazu: DA - ako je Pera ubrao više jabuka od Mike, a NE - ako Pera nije ubrao više jabuka od Mike.

#### **Primer 1**

##### *Ulaz*

73

10

##### *Izlaz*

DA

#### **Primer 2**

##### *Ulaz*

48

48

##### *Izlaz*

NE

#### **Primer 3**

##### *Ulaz*

35

56

**Izlaz**

NE

**2.1.1.2. Zbir godina braće i sestre**

Pera, Mika i Laza su tri brata rođena u istom danu, a Ana je njihova 3 godine starija sestra. Napisati program kojim se proverava da li uneti broj može biti zbir njihovih godina.

**Ulaz**

Sa standardnog ulaza unosi se jedan pozitivan prirodan broj manji od 500.

**Izlaz**

Na standardnom izlazu prikazati reč da ako uneti broj može biti zbir godina Pere, Mike, Laze i Ane, a ako ne može prikazati reč ne.

**Primer****Ulaz**

27

**Izlaz**

da

**2.1.1.3. Jednakostranični trougao datog obima**

Date su celobrojne dužine triju stranica trougla. Napisati program kojim se proverava da li postoji jednakostranični trougao celobrojne dužine stranica istog obima kao dati trougao i koja mu je dužina stranice.

**Ulaz**

Na standardnom ulazu nalaze se 3 prirodna broja koja predstavljaju stranice trougla.

**Izlaz**

Jedna linija standardnog izlaza sadrži reč da ako postoji traženi trougao, za kojom odvojeno prazninom, sledi dužina stranice tog trougla, ako traženi trougao ne postoji sadrži reč ne.

**Primer****Ulaz**

3  
4  
5

**Izlaz**

da 4

#### 2.1.1.4. Teme pravougaonika

Date su koordinate tri temena pravougaonika sa celobrojnim koordinatama čije su stranice paralelne koordinatnim osama. Temena su data u proizvoljnom redosledu. Napisati program kojim se određuju koordinate četvrtog temena.

##### **Ulaz**

Sa standardnog ulaza učitavaju se celobrojne koordinate tri temena pravougaonika - u svakom redu po dva podatka (koordinata X i koordinata Y), odvojena razmakom.

##### **Izlaz**

X i Y koordinata traženog temena razdvojene razmakom.

##### **Primer**

##### **Ulaz**

3 5  
7 5  
7 9

##### **Izlaz**

3 9

#### 2.1.2. Logički operatori

##### 2.1.2.1. Dva broja istog znaka

Napisati program kojim se proverava da li su dva cela broja istog znaka.

##### **Ulaz**

Na standarnom ulazu nalaze se dva cela broja  $a, b$  različita od 0,  $-10^{**4} \leq a, b \leq 10^{**4}$ .

##### **Izlaz**

Jedna linija standarnog izlaza koja sadrži reč da ako su dati brojevi istog znaka, inače sadrži reč ne.

##### **Primer**

##### **Ulaz**

234  
-34

##### **Izlaz**

ne

##### 2.1.2.2. Isti kvadrant

Napisati program kojim se proverava da li dve tačke  $A(x_1, y_1)$ ,  $B(x_2, y_2)$  pripadaju istom kvadrantu. Smatraćemo da tačke na pozitivnom delu x ose



pripadaju prvom i četvrtom kvadrantu, tačke na negativnom delu x ose pripadaju drugom i trećem kvadrantu, slično tačke na pozitivnom delu y ose pripadaju prvom i drugom kvadrantu, a na negativnom delu y ose trećem i četvrtom kvadrantu, a da koordinatni početak pripada svim kvadrantima.

#### **Ulaz**

Standardni ulaz sadrži četiri cela broja, svaki u posebnoj liniji:

$x_1, y_1$  ( $-10^{**4} \leq x_1, y_1, \leq 10^{**4}$ ) - koordinate tačke A( $x_1, y_1$ )

$x_2, y_2$  ( $-10^{**4} \leq x_2, y_2, \leq 10^{**4}$ ) - koordinate tačke B( $x_2, y_2$ )

#### **Izlaz**

Na standardnom izlazu u jednoj liniji prikazati reč da ako tačke pripadaju istom kvadrantu u suprotnom prikazati reč ne.

### **Primer**

#### **Ulaz**

12

-45

15

23

#### **Izlaz**

ne

### **2.1.2.3. Tačka u pravougaoniku i krugu**

Napiši program koji za tačku u ravni zadatu svojim koordinatama ispituje da li pripada zadatom krugu i zadatom pravougaoniku čije su stranice paralelne sa koordinatnim osama.

#### **Ulaz**

Sa standardnog ulaza učitavaju se sledeći realni brojevi (brojevi u istom redu su razdvojeni jednim razmakom):

$x, y$  - koordinate tačke,

$x_0, y_0$  - koordinate zajedničkog centra kruga i pravougaonika,

$r$  - poluprečnik kruga,

$w, h$  - dužina i širina stranica pravougaonika.

#### **Izlaz**

Na standardni izlaz ispisati dva linije teksta. U prvoj liniji treba da piše jeste u krugu ako tačka ( $x, y$ ) pripada krugu sa centrom ( $x_0, y_0$ ) poluprečnika  $r$  odnosno nije u krugu ako tačka ne pripada krugu. U drugoj liniji treba da piše jeste u pravougaoniku ako tačka ( $x, y$ ) pripada pravougaoniku čiji je centar (težište) u tački ( $x_0, y_0$ ), čije su stranice paralelne koordinatnim osama i čija je dužina  $w$  tj.  $h$ , odnosno nije u pravougaoniku ako tačka ne pripada unutrašnjosti tog pravougaonika. Granica kruga (kružnica) i pravougaonika smatraju se njihovim delom.

**Primer****Ulaz**

1 1  
0 0  
1  
2 2

**Izlaz**

nije u krugu  
jeste u pravougaoniku

**2.1.2.4. Radno vreme**

Radno vreme jedne organizacije je između 9 i 17 časova. Odrediti da li je poslato mejl stigao u toku radnog vremena.

**Ulaz**

Sa standardnog ulaza se unose dva cela broja, svaki u posebnom redu, koji predstavljaju sat i minut poslatog mejla.

**Izlaz**

Na standardni izlaz ispisati da ako je mejl stigao u toku radnog vremena tj. ne ako nije (9:00 spada u radno vreme, dok 17:00 ne spada).

**Primer 1****Ulaz**

14  
53

**Izlaz**

da

**Primer 2****Ulaz**

17  
01

**Izlaz**

ne

**2.1.2.5. Kućni red**

Kućni red zabranjuje pravljenje buke pre 6 časova i između 13 i 17 časova, i nakon 22 časa. Napiši program koji radnicima govori da li u nekom datom trenutku mogu da izvode bučnije radove.

**Ulaz**

Sa standardnog ulaza se unosi ceo broj između 0 i 23 koji predstavlja sat.

**Izlaz**

Na standardni izlaz ispisati poruku moze ako je dozvoljeno izvoditi radove tj. ne moze ako nije.

**Primer 1****Ulaz**

5

**Izlaz**

ne moze

**Primer 2****Ulaz**

6

**Izlaz**

moze

**Primer 3****Ulaz**

13

**Izlaz**

ne moze

**2.1.2.6. Postoji li trougao datih dužina stranica**

Napisati program kojim se proverava da li postoji trougao sa datim dužinama stranica.

**Ulaz**

Na standarnom ulazu nalaze se tri realna pozitivna broja, svaki u posebnoj liniji. Brojevi predstavljaju dužine stranica a, b, c.

**Izlaz**

Jedna linija standarnog izlaza koja sadrži reč da ako postoji trougao, inače sadrži reč ne.

**Primer****Ulaz**

4.3

5.4

6.7

**Izlaz**

da

### 2.1.2.7. Prestupna godina

Napiši program koji proverava da li je uneta godina prestupna.

**Ulaz**

Sa standardnog ulaza se unosi broj godine između 1900 i 2200.

**Izlaz**

Na standardni izlaz ispisati da ako je godina prestupna tj. ne ako nije.

#### Primer 1

**Ulaz**

2004

**Izlaz**

da

#### Primer 2

**Ulaz**

2017

**Izlaz**

ne

### 2.1.2.8. Da li se dame napadaju

Napiši program koji proverava da li se dve dame (kraljice) na šahovskoj tabli međusobno napadaju (kraljice se napadaju ako se nalaze u istoj vrsti, istoj koloni ili na istoj dijagonali šahovske table).

**Ulaz**

Sa standardnog ulaza se učitavaju koordinate polja na kojem se nalazi jedna kraljica (dva broja između 0 i 7 razdvojena razmakom) i u narednom redu koordinate polja na kojem se nalazi druga kraljica (ponovo dva broja između 0 i 7 razdvojena razmakom). Pretpostavljamo da se kraljice nalaze na različitim poljima.

**Izlaz**

Na standardni izlaz ispisati tekst da ako se kraljice napadaju tj. ne u suprotnom.

#### Primer

**Ulaz**

5 3

1 7

**Izlaz**

da

### 2.1.3. Minimum i maksimum

#### 2.1.3.1. Kuglice

U svakoj od dve kutije nalaze se samo crvene i plave kuglice. Napiši program koji za dati broj crvenih i plavih kuglica u svakoj od kutija određuje najmanji broj kuglica koje je potrebno premestiti iz kutije u kutiju da bi u jednoj kutiji bile samo crvene, a u drugoj samo plave kuglice.

##### **Ulaz**

Sa standardnog ulaza učitavaju se četiri cela broja (svaki u posebnom redu): broj crvenih kuglica u prvoj kutiji, broj plavih kuglica u prvoj kutiji, broj crvenih kuglica u drugoj kutiji i broj plavih kuglica u drugoj kutiji.

##### **Izlaz**

Na standardni izlaz ispisati jedan ceo broj -- ukupan broj kuglica koje treba premestiti iz kutije u kutiju.

##### **Primer**

##### **Ulaz**

10  
5  
7  
8

##### **Izlaz**

12

Iz prve u drugu kutiju premešta se 5 plavih kuglica, a iz druge u prvu 7 crvenih kuglica.

#### 2.1.3.2. Skupovi sportista

U odeljenju ima  $n$  učenika, od čega  $k$  učenika ume da igra košarku, a  $f$  ume da igra fudbal. Na sportsko takmičenje povešće se svi učenici koji umeju da igraju oba sporta. Koliko će najmanje, a koliko najviše učenika ići na takmičenje?

##### **Ulaz**

Sa standardnog ulaza učitavaju se tri cela broja:  
 $n$  ( $20 \leq n \leq 30$ ) - ukupan broj učenika u odeljenju  
 $k$  ( $0 \leq k \leq n$ ) - broj učenika koji igraju košarku  
 $f$  ( $0 \leq f \leq n$ ) - broj učenika koji igraju fudbal

##### **Izlaz**

Na standardni izlaz ispisati dva cela broja (svaki u posebnom redu).  
 najmanji broj učenika koji će ići na takmičenje  
 najveći broj učenika koji će ići na takmičenje

**Primer****Ulaz**

27

15

19

**Izlaz**

7

15

**2.1.3.3. Intervali**

Data su dva zatvorena intervala realne prave  $[a_1, b_1]$  i  $[a_2, b_2]$ . Napisati program kojim se određuje: njihov pokrivač (najmanji interval realne prave koji sadrži date intervale), presek (najveći interval realne prave koji je sadržan u oba intervala ako postoji) i dužinu njihove unije (dela realne prave koji ti intervali pokrivaju).

**Ulaz**

Sa standardnog ulaza učitavaju se četiri realna broja:  $a_1$ ,  $b_1$ ,  $a_2$  i  $b_2$ , pri čemu važi  $a_1 < b_1$  i  $a_2 < b_2$ .

**Izlaz**

Na standardni izlaz ispisuje se 5 realnih brojeva (u podrazumevanom formatu ispisa): u prvom redu ispisuju se levi i desni kraj pokrivača razdvojeni razmakom, u narednom levi i desni kraj preseka razdvojeni razmakom ili tekst presek ne postoji ako se intervali ne seku i u trećem redu se ispisuje dužina unije.

**Primer 1****Ulaz**

1

3

2

4

**Izlaz**

1 4

2 3

3

**Primer 2****Ulaz**

0.5

4.5

-2.5

-1.5

**Izlaz**

-2.5 4.5

presek ne postoji

5

**2.1.3.4. Sijalice**

U sobi postoje plava i žuta sijalica. Tokom jednog dana i jedna i druga su neko vreme bile uključene. Za svaku sijalicu je poznato vreme (u obliku sat, minut, sekund) kada je uključena i kada je isključena. Napiši program koji određuje koliko vremena je soba bila osvetljena plavo, koliko vremena je bila osvetljena zeleno i koliko vremena je bila osvetljena žuto (soba je osvetljena zeleno dok su istovremeno uključene plava i žuta sijalica).

**Ulaz**

Sa standardnog ulaza se unosi 12 brojeva (četiri reda sa po tri broja razdvojena razmacima). Prva tri broja predstavljaju sat, minut i sekund uključivanja plave sijalice, naredna tri sat, minut i sekund njenog isključivanja, naredna tri sat, minut i sekund uključivanja žute sijalice i poslednja tri sat, minut i sekund njenog isključivanja.

**Izlaz**

Na standardni izlaz ispisati tri linije teksta. U prvoj ispisati tri broja razdvojena dvotačkama koja predstavljaju vreme (u satima, minutima i sekundama) koje je soba bila obojena plavo. U narednoj liniji u istom obliku ispisati vreme koje je soba bila obojena zeleno i u poslednjoj liniji u istom obliku ispisati vreme koje je soba bila obojena žuto.

**Primer 1****Ulaz**

13 15 0

14 20 0

13 45 0

15 30 0

**Izlaz**

0:30:0

0:35:0

1:10:0

**Primer 2****Ulaz**

10 50 20

12 40 11

8 45 30

10 30 20

**Izlaz**

1:49:51

0:0:0

1:44:50

**2.1.3.5. Pozitivan deo intervala**

Na celobrojnoj brojevnoj pravoj dat je interval  $[a,b]$ . Napiši program koji određuje dužinu dela tog intervala koji pripada pozitivnom delu prave.

**Ulaz**

Sa standardnog ulaza učitavaju se dva cela broja  $a$  i  $b$  ( $-10^{**5} \leq a \leq b \leq 10^{**5}$ ).

**Izlaz**

Na standardni izlaz ispisati jedan ceo broj koji predstavlja traženu dužinu pozitivnog dela intervala.

**Primer 1****Ulaz**

-3

5

**Izlaz**

5

**Primer 2****Ulaz**

2

5

**Izlaz**

3

**Primer 3****Ulaz**

-3

-1

**Izlaz**

0

**2.1.3.6. Deo pravougaonika u prvom kvadrantu**

Dat je pravougaonik čije su stranice paralelne koordinatnim osama. Odrediti površinu dela tog pravougaonika koji pripada prvom kvadrantu.



**Ulaz**

Sa standardnog ulaza učitavaju se 4 cela broja koji predstavljaju koordinate temena jedne dijagonale pravougaonika.

**Izlaz**

Na standardni izlaz ispisuje se jedan ceo broj - površina dela pravougaonika u prvom kvadrantu.

**Primer****Ulaz**

-3  
5  
2  
7

**Izlaz**

4

**2.1.3.7. Presek pravougaonika**

Odrediti površinu preseka dva pravougaonika čije su stranice paralelne koordinatnim osama. Pravougaonici su zadati sa po dva naspramna temena.

**Ulaz**

Sa standardnog ulaza učitava se 8 celih brojeva iz intervala  $[-100,100]$ .

$x_{11}, y_{11}$  - jedno teme prvog pravougaonika

$x_{12}, y_{12}$  - njemu naspramno teme

$x_{21}, y_{21}$  - jedno teme drugog pravougaonika

$x_{22}, y_{22}$  - njemu naspramno teme

Koordinate svake tačke su navedene u posebnom redu, razdvojene sa tačno jednim razmakom.

**Izlaz**

Na standardni izlaz ispisati jedan ceo broj - površinu preseka ta dva pravougaonika.

**Primer****Ulaz**

-5 -5  
2 2  
0 0  
3 3

**Izlaz**

4

### 2.1.3.8. Proja

Kukuruzni hleb, proja, se pravi mešanjem kukuruznog i pšeničnog brašna i vode (naravno, po želji možete dodati jaja, so, sir, kajmak i slično, ali mi ćemo razmatrati samo neophodne sastojke). Neophodno je da upotrebljena količina kukuruznog brašna bude dvostruko veća od upotrebene količine pšeničnog brašna. Količina vode koja se dodaje u mešavinu brašna je dvostruko manja od ukupne količine brašna. Ako je data količina kukuruznog i količina pšeničnog brašna, kojom raspolažemo odrediti koju količinu vode treba dodati ako želimo da umesimo najveću moguću proju od namirnica kojima raspolažemo.

#### **Ulaz**

U prvoj liniji standardnog ulaza nalazi se količina kukuruznog brašna kojom raspolažemo, a u drugoj količina pšeničnog brašna kojom raspolažemo (realni brojevi).

#### **Izlaz**

Količina vode koju treba dodati u mešavinu brašna (realni broj prikazan na dve decimale) navodi se u jednoj liniji standardnog izlaza.

### **Primer**

#### **Ulaz**

1524  
800

#### **Izlaz**

1143.00

### 2.1.3.9. Kraljevo rastojanje

Napiši program koji određuje koliko je poteza kralju na šahovskoj tabli potrebno da dođe na dato polje. Kralj se u svakom potezu može pomeriti najviše jedno polje u bilo kom od 8 smerova (gore, dole, levo, desno i dijagonalno).

#### **Ulaz**

Sa standardnog ulaza se učitavaju četiri broja između 1 i 8 (u svakoj liniji po dva, razdvojena razmakom). Prva dva broja predstavljaju koordinate polaznog, a druga dva broja koordinate dolaznog polja.

#### **Izlaz**

Na standardni izlaz ispisati jedan ceo broj koji predstavlja najmanji broj poteza potreban kralju da sa polaznog stigne do dolaznog polja.

### **Primer**

#### **Ulaz**

1 2  
3 4

**Izlaz**

2

**Primer****Ulaz**

8 3

2 5

**Izlaz**

6

**2.2. Ugnežđeno grananje****2.2.1. Intervali****2.2.1.1. Agregatno stanje**

Napisati program kojim se na osnovu temperature vode određuje njeno agregatno stanje. Ako je temperatura:

veća od 0°C i manja od 100°C - agregatno stanje je tečno

ne veća od 0°C - agregatno stanje je čvrsto,

ne manja od 100°C - agregatno stanje je gasovito.

Za temperaturu od tačno 0° smatra se da je agregatno stanje čvrsto, a za tačno 100° da je gasovito.

**Ulaz**

Temperatura - ceo broj od –100 do 200.

**Izlaz**

Na standardni izlaz ispisati jednu od sledećih reči: cvrsto, tecno, gasovito.

**Primer****Ulaz**

-10

**Izlaz**

cvrsto

**2.2.1.2. Školarina**

U jednoj privatnoj školi uvedeno je pravilo kojim se određuje iznos popusta koji ostvaruju učenici prilikom upisa u narednu školsku godinu. Učenici sa odličnim uspehom ostvaruju popust od 40% ukupnog iznosa školarine, sa vrlodobrim 20% a sa dobrim 10%. Takođe, učenici koji su osvojili nagradu na nekom od državnih takmičenja ostvaruju popust od 30% ukupnog iznosa školarine.

Ukoliko neki učenik ispunjava dva kriterijuma za popust primenjuje se

kriterijum po kome je popust veći. Na osnovu punog iznosa školarine, prosečne ocene učenika i informacije o nagradama sa takmičenja odrediti iznos koji učenik treba da plati pri upisu u narednu školsku godinu.

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se pun iznos školarine (realan broj), u drugoj prosečna ocena učenika (realan broj od 2.0 do 5.0) a u trećoj 0 ukoliko učenik nema nagradu ili 1 ukoliko je ima.

**Izlaz**

Iznos školarine koju učenik treba da plati (zaokružen na najbliži ceo broj) navodi se u jednoj linije standardnog izlaza.

**Primer**

**Ulaz**

4000

4.65

1

**Izlaz**

2400

**2.2.1.3. Uspeh učenika**

Napisati program kojim se na osnovu datog proseka učenika prikazuje uspeh učenika. Odličan uspeh ima učenik čiji je prosek veći ili jedan 4.5. Vrlodobar uspeh postiže učenik čiji je prosek veći ili jednak 3.5, a manji od 4.5, dobar uspeh se postiže za prosek koji je veći ili jedan 2.5 a manji od 3.5, dovoljan uspeh za prosek veći ili jednak 2, a manji od 2.5. Ako učenik ima neku jedinicu njegov prosek je 1, a uspeh mu je nedovoljan.

**Ulaz**

Sa standardnog ulaza učitava se jedan realan broj iz skupa  $[2,5] \cup \{1\}$  koji predstavlja prosek učenika.

**Izlaz**

Na standardni izlaz prikazati uspeh učenika reč odlican, vrlodobar, dobar, dovoljan ili nedovoljan.

**Primer**

**Ulaz**

3.75

**Izlaz**

vrlodobar

#### **2.2.1.4. Ocena na ispitu**

Ocena na ispitu određuje se na osnovu broja osvojenih poena (može se osvojiti između 0 i 100 poena). Svi studenti koji su dobili manje od 51 poen automatski padaju ispit i dobijaju ocenu 5. Ocena 6 se dobija za broj poena veći ili jednak od 51, a manji od 61, ocena 7 za broj poena veći ili jednak od 61, a manji od 71, ocena 8 za broj poena veći ili jednak od 71, a manji od 81, ocena 9 za broj poena veći ili jednak od 81 a manji od 91, a ocena 10 za broj poena veći ili jednak od 91.

##### ***Ulaz***

Sa standardnog ulaza učitava se jedan ceo broj između 0 i 100 koji predstavlja broj poena osvojenih na ispitu.

##### ***Izlaz***

Na standardni ispit ispisati jedan ceo broj - ocenu na ispitu.

#### **Primer 1**

##### ***Ulaz***

73

##### ***Izlaz***

8

#### **Primer 2**

##### ***Ulaz***

50

##### ***Izlaz***

5

#### **Primer 3**

##### ***Ulaz***

51

##### ***Izlaz***

6

#### **Primer 4**

##### ***Ulaz***

100

##### ***Izlaz***

10

### 2.2.1.5. Solidarni porez

Porez je određen na sledeći način. Prvih 60000 dinara bruto plate se ne oporezuje. Na deo bruto plate između 60000 i 100000 dinara, plaća se 20% poreza dok se na deo bruto plate preko 100000 dinara plaća porez od 25%. Napiši program koji za unetu bruto platu (platu pre odbijanja poreza) izračunava neto platu (platu posle odbijanja poreza).

#### **Ulaz**

Sa standardnog ulaza se učitava jedan realan broj -- iznos bruto plate.

#### **Izlaz**

Na standardni izlaz ispisuje se jedan realan broj -- iznos neto plate, zaokružen na dve decimale.

#### **Primer**

##### **Ulaz**

100000

##### **Izlaz**

92000.00

### 2.2.1.6. Odmor na pola puta

Putnik se kretao T1 sati brzinom V1 km/h, a zatim T2 sati brzinom V2 km/h i T3 sati brzinom V3 km/h. Planirao je da se odmori na pola puta, izračunati posle koliko vremena od početka putovanja je imao odmor.

#### **Ulaz**

Sa standardnog ulaza učitava se 6 realnih brojeva, svaki u zasebnom redu: T1, V1, T2, V2, T3, V3.

#### **Izlaz**

Na standardni izlaz ispisati jedan realan broj koji predstavlja vreme potrebno da putnik stigne do pola puta. Dopusštena greška u izračunavanju i prikazu rezultata je  $10^{-3}$ .

#### **Primer**

##### **Ulaz**

1 10

2 20

3 30

##### **Izlaz**

3.667

### 2.2.1.7. Rastojanje tačka duž

Odrediti rastojanje date tačke A od date duži A0A1 koja leži na x-osi (rastojanje tačke i duži definiše se kao najkraće među rastojanjima između te tačke i tačaka te duži).

#### **Ulaz**

Sa standardnog ulaza unose se 4 realna broja (svaki u posebnom redu):

$x_0$  ( $-10 \leq x_0 < 10$ ) - x-koordinata tačke A0( $x_0, 0$ )

$x_1$  ( $x_0 < x_1 \leq 10$ ) - x-koordinata tačke A1( $x_1, 0$ )

$x, y$  ( $-20 \leq x, y \leq 20$ ) - koordinate tačke A( $x, y$ )

#### **Izlaz**

Na standardni izlaz ispisati jedan realan broj - najkraće rastojanje tačke A i duži A0A1 zaokruženo na pet decimala.

#### **Primer 1**

##### **Ulaz**

0

1

-1

1

##### **Izlaz**

1.41421

#### **Primer 2**

##### **Ulaz**

0

1

0.5

3

##### **Izlaz**

3.00000

### 2.2.1.8. Pravougaoni prsten

U programu za rad sa grafikom potrebno je implementirati interakciju mišem sa pravougaonicima nacrtanim na ekranu, čije su ivice paralelne koordinatnim osama. Pravougaonik se mišem prevlači po ekranu tako što se uhvati za ivicu, a da bi se to lakše radilo dopuštena je tolerancija od po 5 piksela sa svake strane ivica pravougaonika. Kada je miš blizu ivice pravougaonika, ivica se iscrtava drugom bojom. Takođe, kada je miš unutar pravougaonika (a kada miš nije blizu ivice) ceo pravougaonik se boji nekom trećom bojom. Napiši program koji za datu poziciju miša i dati pravougaonik određuje da li je miš blizu ivice (računajući datu toleranciju), ili je unutar pravougaonika, dovoljno daleko od ivice ili je van pravougaonika, dovoljno daleko od ivice.

**Ulaz**

Sa standardnog ulaza unosi se 6 celih brojeva, svaki u posbnoj liniji.

x,y - koordinate miša

x0,y0 - koordinate gornjeg levog temena pravougaonika

w,h - širina i visina pravougaonika.

Napomena: koordinata y u računarskoj grafici raste odozgo naniže.

**Izlaz**

Na standardni izlaz ispisati UNUTRA ako se miš nalazi unutar (proširenih) ivica pravougaonika, SPOLJA ako je miš van (proširenih) ivica pravougaonika, tj. NA IVICI ako se nalazi na (proširenoj) ivici pravougaonika.

**Primer 1****Ulaz**

101

101

100

100

100

100

**Izlaz**

NA IVICI

**Primer 2****Ulaz**

106

106

100

100

50

100

**Izlaz**

UNUTRA

**Primer 3****Ulaz**

94

94

100

100

50

100

**Izlaz**

SPOLJA



## 2.2.2. Diskretni skup vrednosti

### 2.2.2.1. Broj dana u mesecu

Napiši program koji za dati redni broj meseca i godinu određuje broj dana u tom mesecu. Voditi računa o tome da li je godina prestupna (godina je prestupna ako je deljiva sa 4, a nije deljiva sa 100, osim ako je deljiva sa 400, kada jeste prestupna).

#### **Ulaz**

Sa standardnog ulaza učitavaju se dva broja:  
broj meseca  $m$  ( $1 \leq m \leq 12$ ) i  
broj godine  $g$  ( $1900 \leq g \leq 2100$ )

#### **Izlaz**

Na standardni izlaz ispisati jedan ceo broj koji predstavlja broj dana u zadatom mesecu.

#### **Primer 1**

##### **Ulaz**

11  
2014

##### **Izlaz**

30

#### **Primer 2**

##### **Ulaz**

2  
2014

##### **Izlaz**

28

#### **Primer 3**

##### **Ulaz**

2  
2016

##### **Izlaz**

29

#### **Primer 3**

##### **Ulaz**

2  
2000

**Izlaz**

29

#### **Primer 4**

**Ulaz**

2

2100

**Izlaz**

28

#### **2.2.2.2. Naziv meseca**

Napiši program koji na osnovu unetog rednog broja meseca ispisuje njegov naziv.

**Ulaz**

Sa standardnog ulaza se unosi redni broj meseca - prirodan broj između 1 i 12.

**Izlaz**

Na standardni izlaz ispisati naziv meseca (ispisan latiničkim ASCII karakterima).

#### **Primer**

**Ulaz**

1

**Izlaz**

januar

#### **2.2.2.3. Redni broj meseca**

Napiši program koji na osnovu naziva meseca određuje njegov redni broj.

**Ulaz**

Sa standardnog ulaza se unosi naziv meseca zapisan latiničkim malim slovima.

**Izlaz**

Na standardni izlaz ispisati jedan prirodan broj koji predstavlja redni broj meseca.

#### **Primer**

**Ulaz**

mart

**Izlaz**

3

### 2.2.3. Leksikografsko grananje

#### 2.2.3.1. Ko je pre stigao

Dva drugara Pera i Mika su došla u školu. Ako se za svakog zna sat, minut i sekund kada je stigao, napiši program koji određuje koji od njih je stigao pre.

##### *Ulaz*

Sa standardnog ulaza učitava se 6 celih brojeva. Prvo sat, minut i sekund kada je stigao Pera, a zatim sat, minut i sekund kada je stigao Mika (pretpostavlja se da brojevi predstavljaju ispravno zadata vremena).

##### *Izlaz*

Na standardni izlaz ispisati jednu liniju teksta. Pera ako je pre stigao Pera, Mika ako je pre stigao Mika ili istovremeno ako su stigli istovremeno.

#### **Primer**

##### *Ulaz*

14  
23  
17  
15  
23  
11

##### *Izlaz*

Pera

#### 2.2.3.2. Punoletstvo

Napisati program kojim se ispituje da li će osoba čiji je datum rođenja poznat biti punoletna (imati punih 18 godina) nekog zadatog datuma.

##### *Ulaz*

U prve tri linije se unosi datum rođenja u redosledu dan, mesec i godina rođenja. U sledeće tri zadati datum u redosledu dan, mesec i godina. Oba datuma su ispravna.

##### *Izlaz*

Ispisati na standardnom izlazu: DA - ako će osoba biti punoletna, i NE - ako osoba neće biti punoletna. Ako je jedan zadati datum tačno 18 godina nakon datuma rođenja, osoba se smatra punoletnom (bez obzira na tačno vreme rođenja).

#### **Primer**

##### *Ulaz*

1  
5

1986

1

5

2004

**Izlaz**

DA

**Ulaz**

1

5

1986

30

4

2004

**Izlaz**

NE

**2.2.3.3. Pobjednik u dvije discipline**

Takmičari su radili testove iz matematike i programiranja. Za svaki predmet dobili su određeni broj poena (ceo broj od 0 do 50). Takmičari se rangiraju po ukupnom broju poena iz oba predmeta. Ako dva takmičara imaju isti broj poena, pobjednik je onaj koji ima više poena iz programiranja. Potrebno je napisati program koji određuje pobjednika takmičenja.

**Ulaz**

Učitavaju se podaci za tri takmičara. Za svakog takmičara učitava se broj poena iz matematike, a zatim broj poena iz programiranja, svaki u posebnom redu.

**Izlaz**

Potrebno je ispisati redni broj pobjednika. Brojevi počinju od 1. Ako su dva ili više takmičara ostvarila potpuno identičan uspeh, pobjednik je onaj koji ima manji redni broj (jer je ostvario više poena na kvalifikacionom takmičenju).

**Primer****Ulaz**

38

42

43

40

40

43

**Izlaz**

3

Prvi takmičar ukupno ima  $38+42=80$  poena, drugi  $43+40=83$ , a treći  $40+43=83$ . Drugi i treći takmičar imaju isti ukupni broj poena, ali treći takmičar ima više poena iz programiranja tako da je on pobjednik.

## **2.2.4. Hijerarhijsko grananje**

### **2.2.4.1. Linearna jednačina**

Napiši program koji određuje broj rešenja linearne jednačine  $a \cdot x + b = 0$  za realne vrednosti koeficijenata  $a$  i  $b$  i rešava je ako postoji jedinstveno rešenje.

#### **Ulaz**

U prvom redu koeficijent  $a$ , u drugom koeficijent  $b$ .

#### **Izlaz**

Ako ima jedno rešenje ispis rešenja na dve decimale, ako nema rešenja poruka: NEMA RESENJA, ako ima beskonačno mnogo rešenja poruka: BESKONACNO RESENJA.

#### **Primer**

##### **Ulaz**

1  
1

##### **Izlaz**

-1.00

##### **Ulaz**

0  
0

##### **Izlaz**

BESKONACNO RESENJA

##### **Ulaz**

0  
1

##### **Izlaz**

NEMA RESENJA

### **2.2.4.2. Kvadranti i ose**

Date su koordinate tačke. Napiši program koji određuje u kom kvadrantu ili na kojoj osi se ona nalazi.

#### **Ulaz**

Sa standardnog ulaza unose se dva cela broja (svaki u posebnom redu):  $x$  - koordinata i  $y$ -koordinata tačke ( $-10 \leq x, y \leq 10$ ).

**Izlaz**

Na standardni izlaz ispisati da li se tačka nalazi u nekom od četiri kvadranta, da li je na nekom delu neke koordinatne ose ili je koordinatni početak.

**Primer 1****Ulaz**

-1  
2

**Izlaz**

2. kvadrant

**Primer 2****Ulaz**

0  
3

**Izlaz**

pozitivni deo y ose

**Primer 3****Ulaz**

0  
0

**Izlaz**

koordinatni pocetak

**2.2.4.3. Položaj dve prave**

Date su dve prave svojim implicitnim jednačinama (jednačinama oblika  $Ax+By+C=0$ ). Odrediti njihov međusobni položaj (poklapaju se, paralelne su ili se seku u tački...)

**Ulaz**

U 6 linija standardnog ulaza nalazi se u svakoj po jedan realan broj. Prva tri broja redom predstavljaju koeficijente A,B i C jedne prave, a druga 3 koeficijente druge prave.

**Izlaz**

U jednoj liniji standardnog izlaza ispisuje se tekst poklapaju se za prave koje se poklapaju, paralelne su za paralelne prave ili seku se za prave koje se seku. Ako se prave seku u drugoj liniji standardnog izlaza ispisuju se koordinate presečne tačke (2 realna broja sa dve decimale).

**Primer****Ulaz**

5  
0  
-10  
1  
1  
0

**Izlaz**

seku se  
2.00 -2.00

**2.2.4.4. Rastojanje tačka pravougaonik**

Deca se igraju jurke u školskom dvorištu pravougaonog oblika i oko njega. Pravila igre su takva da je učenik bezbedan kada se uhvati za ogradu. Napiši program koji određuje koliko je najmanje rastojanje koje Đoka treba da pretrči da bi bio bezbedan (on se u početku može nalaziti i u dvorištu i van njega).

**Ulaz**

Sa standardnog ulaza učitava se 6 realnih brojeva (svaki u posebnom redu):  $x_0, y_0$  ( $0 \leq x_0, y_0 < 1000$ ) - koordinate donjeg levog temena pravougaonika koji predstavlja školsko dvorište u koordinatnom sistemu postavljenom tako da su ivice pravougaonika paralelne koordinatnim osama.

$x_1, y_1$  ( $x_0 < x_1 \leq 1000, y_0 < y_1 \leq 1000$ ) koordinate gornjeg desnog temena pravougaonika.

$x, y$  ( $0 \leq x, y \leq 1000$ ) - koordinate tačke na kojoj se nalazi Đoka.

**Izlaz**

Na standardni izlaz ispisati jedan realan broj - najkraće rastojanje Đoke do ograde u metrima (dopuštena je tolerancija je jedan centimetar, tj. rezultat ispisati zaokružen na dve decimale).

**Primer 1****Ulaz**

100  
100  
200  
200  
50  
170

**Izlaz**

50.00

**Primer 2*****Ulaz***

100  
100  
200  
200  
50  
300

***Izlaz***

111.80

**2.2.5. Razno****2.2.5.1. Sutrašnji datum**

Napiši program koji za uneti datum određuje datum narednog dana.

***Ulaz***

Sa standardnog ulaza se unose tri pozitivna cela broja (svaki u zasebnom redu) koja predstavljaju dan, mesec i godinu jednog ispravnog datuma.

***Izlaz***

Na standardni izlaz ispisati tri cela broja koja predstavljaju dan, mesec i godinu sutrašnjeg datuma. Svi brojevi se ispisuju u jednom redu, a iza svakog broja navodi se tačka.

**Primer 1*****Ulaz***

1  
1  
2016

***Izlaz***

2.1.2016.

**Primer 2*****Ulaz***

28  
2  
2016

***Izlaz***

29.2.2016.



**Primer 3*****Ulaz***

28

2

1900

***Izlaz***

1.3.1900.

**2.2.5.2. Jučerašnji datum**

Napiši program koji za uneti datum određuje datum prethodnog dana.

***Ulaz***

Sa standardnog ulaza se unose tri pozitivna cela broja (svaki u zasebnom redu) koja predstavljaju dan, mesec i godinu jednog ispravnog datuma.

***Izlaz***

Na standardni izlaz ispisati tri cela broja koja predstavljaju dan, mesec i godinu jučerašnjeg datuma. Svi brojevi se ispisuju u jednom redu, a iza svakog broja navodi se tačka.

**Primer 1*****Ulaz***

1

3

2016

***Izlaz***

29

2

2016

**Primer 2*****Ulaz***

1

3

1900

***Izlaz***

28.2.1900.

**Primer 3*****Ulaz***

1.1.2013.

**Izlaz**

31.12.2013.

**2.2.5.3. Vrsta trougla na osnovu uglova**

Data su tri konveksna ugla izražena u stepenima i minutima. Napisati program kojim se proverava da li to mogu biti uglovi trougla, i ako mogu kakav je to trougao u odnosu na uglove (oštrougli, pravougli ili tupougli).

**Ulaz**

Sa standardnog ulaza unosi se šest celih brojeva, svaki u posebnoj liniji. Brojevi predstavljaju tri ugla izražena u stepenima. Podaci predstavljaju ispravno zadate uglove, stepeni i minuti određuju ispravno zapisane uglove u intervalu od 0 stepeni i 0 minuta, do 180 stepeni i 0 minuta (minuti su uvek broj između 0 i 59).

**Izlaz**

Jedna linija standardnog izlaza koja sadrži reč ostrougli, pravougli ili tupougli, ako trougao sa datim uglovima postoji, u zavisnosti od vrste trougla, ili reč ne ako ne postoji trougao sa datim uglovima.

**Primer****Ulaz**

35  
23  
92  
37  
52  
0

**Izlaz**

tupougli

**2.2.5.4. Vrsta trougla na osnovu stranica**

Napiši program koji na osnovu dužina stranica trougla proverava da li takav trougao postoji i da li je jednakostranični, jednakokraki ili raznostranični.

**Ulaz**

Sa standardnog ulaza se unose tri pozitivna cela broja a, b i c, svaki u posebnom redu.

**Izlaz**

Na standardni izlaz ispisati ne postoji ako ne postoji trougao čije su dužine stranica a, b i c, jednakostranični ako su ti brojevi dužine stranica nekog jednakostraničnog trougla, jednakokraki ako su dužine stranica nekog jednakokrakog trougla ili raznostranični ako su dužine stranica nekog raznostraničnog trougla.

**Primer 1***Ulaz*

3

4

5

*Izlaz*

raznostranicni

**Primer 2***Ulaz*

3

4

3

*Izlaz*

jednakokraki

**Primer 3***Ulaz*

1

5

1

*Izlaz*

ne postoji

**2.2.5.5. Tenis**

Napiši program koji na osnovu broja osvojenih gemova dva igrača određuje ishod seta u tenisu.

*Ulaz*

Sa standardnog ulaza se unose dva prirodna broja (razdvojena razmakom) koji predstavljaju broj osvojenih gemova svakog igrača redom.

*Izlaz*

Ako je prvi igrač osvojio set na standardni izlaz ispisati poruku pobedio prvi. Ako je drugi igrač osvojio set ispisati pobedio drugi. Ako uneti rezultat neispravan ispisati neispravno. Ako set još nije završen ispisati nije završeno.

**Primer 1***Ulaz*

7

5

*Izlaz*

pobedio prvi

## **Primer 2**

*Ulaz*

3

7

*Izlaz*

neispravno

### 3. Iteracija

#### 3.0. Male serije

##### 3.0.1. Statistike

###### 3.0.1.1. Najlošiji kontrolni

Učenci tokom godine rade 5 kontrolnih zadataka. Nastavnik je obećao da će prilikom zaključivanja ocena svakom učeniku zanemariti najslabiju ocenu (ako postoji više takvih, zanemariće samo jednu). Napiši program koji učitava 5 ocena jednog učenika i ispisuje prosečnu ocenu kada se zanemari najslabije urađeni kontrolni.

###### **Ulaz**

U svakoj od pet linija standardnog ulaza nalazi se po jedna ocena (ceo broj između 1 i 5) koju je učenik dobio.

###### **Izlaz**

Prosečna ocena bez najlošijeg kontrolnog, zaokružena na dve decimale.

###### **Primer**

###### **Ulaz**

5  
4  
4  
5  
3

###### **Izlaz**

4.50

###### 3.0.1.2. Najjeftiniji za dinar

U jednoj prodavnici je u toku akcija u kojoj se kupcima koji kupe tri proizvoda nudi da najjeftiniji od ta tri proizvoda dobiju za jedan dinar. Napiši program koji određuje sniženu cenu tri proizvoda.

###### **Ulaz**

Sa standardnog ulaza unose se tri cela broja iz intervala od 50 do 5000 koji predstavljaju cene u dinarima za tri kupljena proizvoda.

**Izlaz**

Na standardni izlaz ispisati jedan ceo broj koji predstavlja ukupnu sniženu cenu ta tri proizvoda.

**Primer****Ulaz**

2499

3599

899

**Izlaz**

6099

**3.0.1.3. Prosek skijaša**

Na takmičenju u skijaškim skokovima pored udaljenosti koju skakač preskoči ocenjuje se i stil skakača. Pet sudija ocenjuju stil skakača ocenama od 0 do 20. Zatim se od svih dobijenih ocena izbrišu jedna najmanja i jedna najveća. Konačna ocena stila skakača računa se kao prosek preostalih ocena. Napiši program kojim se na osnovu ocena 5 sudija određuje konačna ocena stila skakača.

**Ulaz**

U svakoj od pet linija standardnog ulaza nalazi se po jedna ocena stila skakača (ceo broj između 0 i 20) koju je skijaš dobio od pet sudija.

**Izlaz**

Konačna ocena stila skakača prikazana na dve decimale.

**Primer****Ulaz**

3

5

0

1

5

**Izlaz**

3.00

**3.0.1.4. Najtopliji dan**

Date su dnevne temperature redom za svaki dane jedne sedmice. Napisati program kojim se određuje redni broj najtoplijeg dana te sedmice (ako ih ima više ispisati prvi). Dani u sedmici su numerisani brojevima od 1 do 7, počev od ponedeljka.

**Ulaz**

Sa standardnog ulaza unosi se 7 celih brojeva iz intervala  $[-50,50]$ . Svaki broj je u posebnom redu. Brojevi redom predstavljaju temperature izmere u ponedjeljak, utorak, sredu, četvrtak, petak, subotu i nedelju.

**Izlaz**

Na standardnom izlazu ispisati traženi redni broj najtoplijeg dana.

**Primer****Ulaz**

27  
32  
28  
27  
32  
31  
29

**Izlaz**

2

**3.0.1.5. H<sub>2</sub>SO<sub>4</sub>**

Koliko se molekula sumporne kiseline H<sub>2</sub>SO<sub>4</sub> može napraviti od datog broja atoma vodonika H, sumpora S i kiseonika O?

**Ulaz**

U svakoj od dve linije standardnog ulaza nalazi se ceo broj koji predstavlja dati broj atoma, redom, vodonika, sumpora, pa kiseonika.

**Izlaz**

U jednoj liniji standardnog izlaza ispisati ceo broj koji predstavlja broj mogućih molekula sumporne kiseline.

**Primer****Ulaz**

75  
58  
106

**Izlaz**

26

**3.0.1.6. Voćni paketi**

Životinje u zoološkom vrtu vole da jedu voće. U vrt se dopremaju paketi voća u kojima se nalazi a kilograma ananasa, b kilograma banana i j kilograma jabuka. Volonteri treba da pripreme dnevni obrok za koji znaju da treba da sadrži bar A kilograma ananasa, B kilograma banana i J kilograma jabuka.

Napiši program koji izračunava koji je najmanji broj paketa voća koje volonteri treba da naruče da bi mogli da pripreme dnevni obrok.

**Ulaz**

Sa standardnog ulaza učitava se 6 celih brojeva, svaki u posebnom redu: a, b, j, iz intervala [1,5] i A, B, J iz intervala [5,100].

**Izlaz**

Na standardni izlaz ispisati jedan ceo broj koji predstavlja broj paketa koje treba naručiti.

**Primer**

**Ulaz**

3  
5  
4  
12  
14  
17

**Izlaz**

5

**3.0.1.7. Najbliži proseku**

Napisati program koji za četiri data broja određuje koji je od ta četiri broja najbliži aritmetičkoj sredini datih brojeva (ako su dva broja jednako blizu ispisati prvi).

**Ulaz**

U četiri linije standardnog ulaza nalaze se četiri realna broja.

**Izlaz**

U jednoj liniji standardnog izlaza prikazati jedan od unetih realnih brojeva, na dve decimale, najbliži njihovoj aritmetičkoj sredini.

**Primer**

**Ulaz**

24.3  
20.2  
23.5  
22.7

**Izlaz**

22.70



### 3.0.1.8. Prvi i drugi maksimum

Napisati program kojim se određuju najveća dva različita broja od pet datih celih brojeva.

#### **Ulaz**

Na standardnom ulazu nalaze se 5 celih brojeva, svaki u posebnoj liniji.

#### **Izlaz**

Prva linija standardnog izlaza sadrži najveći broj od datih 5 brojeva. Druga linija standardnog izlaza sadrži drugi po veličini ceo broj od datih 5 brojeva. Ako su svi uneti brojevi jednaki druga linija treba da sadrži samo znak '-'.

#### **Primer 1**

##### **Ulaz**

2  
7  
-4  
7  
3

##### **Izlaz**

7  
3

#### **Primer 2**

##### **Ulaz**

12  
12  
12  
12  
12

##### **Izlaz**

12  
-

### 3.0.1.9. Najbolji takmičari

Na takmičenju u plivanju slobodnim stilom na 100 metara učestvovalo je 5 takmičara sa starnim brojevima od 1 do 5. Poznati su rezultati svih takmičara izraženi u sekundama. Napisati program koji ispisuje startne brojeve svih onih takmičara koji su postigli najbolji rezultat (najmanje vreme) na tom takmičenju.

#### **Ulaz**

U pet linija standardnog ulaza nalaze se pet realnih brojeva iz intervala [40,100], svaki broj u posebnoj liniji. Brojevi predstavljaju rezultate takmičara redom od takmičara sa startnim brojem 1 do takmičara sa startnim brojem 5.

**Izlaz**

Na standardnom izlazu ispisuju se startni brojevi takmičara koji su postigli najbolji rezultat. Svaki startni broj je u posebnoj liniji, startni brojevi su prikazani od najmanjeg do najvećeg.

**Primer****Ulaz**

54.7  
50.3  
53.2  
50.3  
58.6

**Izlaz**

2  
4

**3.0.1.10. Automobil u vođstvu**

Tri automobila kreću sa startne pozicije u trenucima  $0 \leq t_1 \leq t_2 \leq t_3$  konstantnim brzinama  $v_1$ ,  $v_2$  i  $v_3$ . Prikazati startne brojeve automobila koji su na vodećoj poziciji u trenutku  $t$  ( $t \geq 0$ ). Ako ni jedan automobil još nije krenuo, sva tri su u vodećoj poziciji.

**Ulaz**

Sa standardnog ulaza učitava se sedam celih brojeva, svaki u zasebnoj liniji  $t_1$ ,  $v_1$ ,  $t_2$ ,  $v_2$ ,  $t_3$ ,  $v_3$ ,  $t$  gde  $t_1, t_2, t_3, t$  predstavljaju vremena u sekundama, a  $v_1, v_2, v_3$  brzine u metrima po sekundi.

**Izlaz**

Na standardni izlaz ispisati jedan ili više celih brojeva (svaki u posebnom redu) koji predstavljaju redne brojeve automobila (1, 2 ili 3) koji su na vodećoj poziciji. Ako je više automobila istovremeno u vođstvu, njihove redne brojeve ispisati u rastućem redosledu.

**Primer****Ulaz**

10  
5  
5  
4  
0  
2  
30

**Izlaz**

1  
2

### 3.0.1.11. Pozicija najvećeg ugla trougla

Data su dva ugla raznostraničnog trougla u stepenima, minutima i sekundama. Odrediti koji od uglova trougla je najveći, prvi po redosledu unosa, drugi po redosledu unosa ili treći, određen u programu.

#### **Ulaz**

U svakoj od šest linija standardnog ulaza nalazi se po jedan ceo broj. Brojevi redom predstavljaju stepene, minute (manje od 60) i sekunde (manje od 60), prvo jednog, pa drugog ugla trougla. Zbir data dva ugla je manji od 180 stepeni.

#### **Izlaz**

U jednoj liniji standardnog izlaza ispisati broj 1, 2 ili 3.

#### **Primer**

##### **Ulaz**

75  
30  
14  
23  
15  
45

##### **Izlaz**

3

## 3.0.2. Filtriranje

### 3.0.2.1. Jutarnje temperature

Date su jutarnje temperature za 4 dana T1, T2, T3 i T4. Odrediti broj dana od ta 4 dana kada je jutarnja temperatura bila ispod prosečne temperature za ta 4 dana.

#### **Ulaz**

Na stanadarnom ulaza nalaze se četiri realna broja, svaki u posebnoj liniji, koji predstavljaju jutarnje temperature za četiri dana.

#### **Izlaz**

U prvoj liniji standardnog ulaza ispisuje se prosečna temperatura za ta četiri dana, zaokružena na dve decimale.

U drugoj liniji standardnog izlaza ispisuje se broj dana kada je temperatura bila ispod prosečne.

#### **Primer**

##### **Ulaz**

27.3

20.2  
23.5  
21.8

**Izlaz**

23.20  
2

### 3.0.2.2. Berza

Trgovac na berzi je tokom jedne radne nedelje svaki dan ili ostvarivao zaradu ili je gubio novac. Od svih dana u kojima je nešto zaradio, odrediti dan u kome je najmanje zaradio i vrednost koju je tog dana zaradio (ako je više takvih dana, prijaviti prvi) ili prijaviti da je svih dana gubio novac.

**Ulaz**

Sa standardnog ulaza unosi se 5 celih brojeva iz intervala  $[-10000, 10000]$  (iznosi koje je trgovac ostvario u ponedjeljak, utorak, sredu, četvrtak i petak), svaki u posebnom redu.

**Izlaz**

Ako je trgovac u nekom danu ostvario zaradu (iznos je strogo veći od nule), na standardni izlaz u prvom redu ispisati najmanji iznos zarade koji je ostvaren tokom nedelje i u drugom redu oznaku dana (PON, UTO, SRE, CET ili PET) u kome je ostvaren taj najmanji profit. Ako nijedan dan nije ostvarena zarada, ispisati red koji sadrži samo karakter -.

#### Primer 1

**Ulaz**

3200  
-420  
-10  
1350  
5670

**Izlaz**

1350  
CET

#### Primer 2

**Ulaz**

-4700  
-360  
-1000  
-1550  
-3245

**Izlaz**

-

### 3.0.2.3. Kvalifikacioni prag

Ako se znaju poeni 4 takmičara i kvalifikacioni prag na takmičenju iz programiranja, odrediti koliko takmičara je osvojilo dovoljno poena i kvalifikovalo se u sledeći krug takmičenja, a koliki je prosek poena onih koji se nisu kvalifikovali.

**Ulaz**

U pet linija standardnog ulaza nalaze se pet celih brojeva iz intervala  $[0,100]$ , svaki broj u posebnoj liniji. Prva četiri broja predstavljaju osvojene poene takmičara, a peti broj je kvalifikacioni prag.

**Izlaz**

Na standardnom izlazu ispisuje se:

- u prvoj liniji broj takmičara koji su se kvalifikovali,
- u drugoj liniji prosek poena (na dve decimale) za grupu takmičara koja se nije kvalifikovala, a u slučaju da su se svi kvalifikovali u drugoj liniji ispisati znak '-'.

### Primer

**Ulaz**

0  
30  
2  
99  
30

**Izlaz**

2  
1.00

### 3.0.2.4. Preticanje

Tri automobila startnih brojeva 1, 2, 3 kreću sa iste startne pozicije redom u trenucima  $t_1, t_2, t_3$  ( $0 \leq t_1 < t_2 < t_3$ ), i kreću se konstantnim brzinama  $v_1, v_2, v_3$ . Napisati program kojim određujemo trenutak kad se desi poslednje preticanje, ili ako nema preticanja ispisati tekst nema.

**Ulaz**

Sa standardnog ulaza učitava se 6 realnih brojeva, svaki u zasebnom redu:  $t_1, v_1, t_2, v_2, t_3, v_3$ .

**Izlaz**

Na standardni izlaz ispisati jedan realan broj koji predstavlja vreme poslednjeg preticanja, zaokružen na dve decimale ili tekst nema ako preticanja nije bilo.

**Primer 1*****Ulaz***

1.6  
20  
2  
12  
7.4  
2.1

***Izlaz***

nema

**Primer 2*****Ulaz***

6.3  
9  
15  
18  
19  
13

***Izlaz***

47.57

**3.0.3. Sortiranje****3.0.3.1. Uredi tri broja**

Milica kupuje patike. Dopala su joj se tri para. Napiši program koji ispisuje cene ta tri para patika od najjeftinijih do najskupljih.

***Ulaz***

Sa standardnog ulaza unose se tri cela broja iz intervala [1000,10000], svaki u posebnom redu.

***Izlaz***

Na standardni izlaz ispisuju se brojevi uređeni od najmanjeg do najvećeg.

**Primer*****Ulaz***

2300  
1800  
2799

***Izlaz***

1800  
2300

2799

**3.0.3.2. Najmanji trocifreni broj**

Od cifara četvorocifrenog prirodnog broja treba formirati najmanji trocifren broj.

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se četvorocifren prirodan broj  $n$  ( $1000 \leq n < 10000$ ).

**Izlaz**

Najmanji trocifren broj dobijen od cifara broja  $n$ .

**Primer****Ulaz**

5302

**Izlaz**

203

**3.0.3.3. Jednakokraki od 4 duži**

Data su četiri pozitivna realna broja koji predstavljaju dužine 4 duži. Napiši program koji ispituje da li se od neke 3 od njih može napraviti jednakokraki trougao?

**Ulaz**

Četiri pozitivna realna broja (svaki u zasebnom redu).

**Izlaz**

Jedna linija teksta u kojoj piše da ako je moguće napraviti trougao tj. ne ako nije.

**Primer****Ulaz**

1.1

2.2

3.3

4.4

**Izlaz**

ne

**Ulaz**

3.2

2.3

3.2

4.5

**Izlaz**

da

**Ulaz**

3.2

2.3

1.1

1.1

**Izlaz**

ne

**3.0.3.4. Prozor i kutija**

Petar i Luka treba da iznesu kutiju oblika kvadra datih dimenzija  $a, b, c$ , kroz prozor oblika pravougaonika dimenzija  $p \times q$  tako da su odgovarajuće ivice kutije budu paralelne ivicama prozora. Napisati program kojim se proverava da li je to moguće.

**Ulaz**

Sa standardnog ulaza unose se 5 prirodnih brojeva koji predstavljaju redom dimenzije kutije  $a, b, c$  i prozora  $p, q$  izražene u centimetrima.

**Izlaz**

U jednoj liniji standardnog izlaza prikazati reč da ako je moguće kutiju izneti kroz prozor, a reč ne ako to nije moguće.

**Primer 1****Ulaz**

75

30

50

70

60

**Izlaz**

da

**Primer 2****Ulaz**

30

40

50

30

40

**Izlaz**

da



### 3.0.3.5. Podela brojevne prave na intervale

Brojeva prava je trima različitim tačkama  $a$ ,  $b$  i  $c$  podeljena na četiri dela. Napisati program koji određuje u kom delu, prvom, drugom, trećem ili četvrtom, se nalazi uneta (četvrta) tačka  $d$ , različita od tačaka  $a$ ,  $b$  i  $c$ .

#### **Ulaz**

Na standardnom ulazu nalaze se 4 različita realna broja, svaki u posebnoj liniji, koji redom predstavljaju tačke kojima je realne prave podeljena ( $a$ ,  $b$  i  $c$ ) i tačku ( $d$ ) za koju određujemo u kom delu se nalazi.

#### **Izlaz**

U jednoj liniji standardnog izlaza ispisuje reč prvom, drugom, trecem ili cetvrtom u zavisnosti kom delu tačka  $d$  pripada.

#### **Primer**

##### **Ulaz**

5  
-6  
3  
1

##### **Izlaz**

drugom

### 3.0.3.6. Sumo rvači

Četiri sumo rvača poznatih težina čekaju ispred lifta poznate nosivosti. Koliki je najmanji broj vožnji potreban da bi se oni prevezli?

#### **Ulaz**

Sa standardnog ulaza učitava se 5 celih brojeva, svaki u posebnom redu:  $a$ ,  $b$ ,  $c$ ,  $d$  ( $100 \leq a, b, c, d, \leq 200$ ) - težine, u kilogramima, 4 sumo rvača,  $L$  ( $200 \leq L \leq 700$ ) - nosivost lifta u kilogramima.

#### **Izlaz**

Na standardni izlaz ispisati jedan ceo broj - broj vožnji potrebnih da se rvači prevezu.

#### **Primer**

##### **Ulaz**

200  
103  
160  
154  
280

##### **Izlaz**

3

Moguće je da se u jednoj vožnji prevezu rvači od 103 i 154 kilograma (jer u zbiru imaju 258 kilograma, što je manje od 280 kilograma kolika je nosivost lifta), a da se rvači od 160 i 200 kilograma prevezu svaki posebno. Dve vožnje nisu moguće (rvači od 200, 160 i 154 ne mogu da stanu u lift jedan sa drugim).

### 3.0.3.7. Orijentacija trougla

U ravni ima puno tačaka i svaka od njih ima svoj redni broj (od 1 do  $n$ ). Trougao je određen sa tri tačke A, B i C, koje su određene svojim rednim brojevima. Međutim, problem je to što su tačke zadate u proizvoljnom redosledu. Na primer, trougao ABC može biti zadat tako što su date tačke B, A i na kraju C. Napiši program koji će tačke ispisati u osnovnom redosledu (od najmanjeg broja, do najvećeg) i uz to odrediti da li raspored tačaka koji je unet određuje trougao koji ima istu ili suprotnu orijentaciju od ovog dobijenog uređivanjem tačaka. Na primer, trouglovi ABC i CAB imaju istu orijentaciju (u smeru suprotnom kretanju kazaljke na satu), dok trougao BAC ima suprotnu orijentaciju.

#### **Ulaz**

Sa standardnog ulaza se unose tri različita prirodna broja, razdvojena razmakom, koji predstavljaju redne brojeve tačaka.

#### **Izlaz**

Na standardni izlaz ispisati tri broja razdvojena razmakom koji se dobijaju uređivanjem brojeva tačaka od najmanjeg do najvećeg, a u narednom redu ispisati isto ako polazni raspored određuje istu tj. suprotno ako određuje suprotnu orijentaciju od onog dobijenog uređivanjem.

### **Primer 1**

#### **Ulaz**

32 14 55

#### **Izlaz**

14 32 55

suprotno

### **Primer 2**

#### **Ulaz**

11 5 8

#### **Izlaz**

5 8 11

isto

### **3.0.4. Cifre broja**

#### **3.0.4.1. Zbir dva trocifrena**

Đaci uvežbavaju sabiranje trocifrenih brojeva. Učiteljica diktira zadatak redom cifru po cifru. Napiši program koji na osnovu učitanih cifara izračunava i ispisuje traženi zbir.

**Ulaz**

U šest linija standardnog ulaza zadato je šest cifara.

**Izlaz**

Na standardni izlaz ispisati jedan ceo broj - traženi zbir.

**Primer**

**Ulaz**

1  
2  
3  
4  
5  
6

**Izlaz**

579

#### **3.0.4.2. Obrni cifre**

Sa standardnog ulaza se unosi prirodan broj  $n$ . Napiši program koji na standardni izlaz ispisuje broj koji se dobija kada se obrne redosled poslednje 4 cifre broja  $n$ .

**Ulaz**

Pozitivan ceo broj manji od  $10^{**}9$ .

**Primer**

**Ulaz**

1234567

**Izlaz**

1237654

**Ulaz**

123

**Izlaz**

3210

### 3.0.4.3. Cifre sdesna

Petar i Marko se igraju igre memorije. Petar diktira Marku cifre jednog broja, jednu po jednu, otpozadi, krenuvši od cifre jedinica. Napiši program koji pomaže Marku da odredi broj čije cifre Petar diktira.

#### ***Ulaz***

Sa standardnog ulaza unosi se 6 dekadnih cifara, svaka u posebnom redu.

#### ***Izlaz***

Na standardni izlaz ispisati jedan ceo broj - broj čije cifre su unete sa standardnog ulaza.

#### **Primer 1**

##### ***Ulaz***

1  
2  
3  
4  
5  
6

##### ***Izlaz***

654321

#### **Primer 2**

##### ***Ulaz***

0  
1  
2  
2  
1  
0

##### ***Izlaz***

12210

## 3.1.serije

### 3.1.1. Pravilne serije

#### 3.1.1.1. Brojevi od a do b

Ako su dati celi brojevi a i b, napisati program koji ispisuje redom sve cele brojeve u zadatom intervalu [a,b].

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se ceo broj  $a$ , a u drugoj je ceo broj  $b$  ( $-1000 \leq a \leq 1000, -1000 \leq b \leq 1000$ ).

**Izlaz**

Na standardnom izlazu ispisuju se redom svi celi brojevi iz intervala, u svakoj liniji po jedan broj.

**Primer****Ulaz**

3  
9

**Izlaz**

3  
4  
5  
6  
7  
8  
9

**3.1.1.2. Brojevi trocifreni parni**

Za date cele brojeve  $a$  i  $b$ , napisati program koji ispisuje redom sve parne trocifrene brojeve koji pripadaju datom intervalu  $[a, b]$

**Ulaz**

Sa standardnog ulaza učitavaju se brojevi  $a$  i  $b$  ( $0 \leq a \leq 1500$  i  $a \leq b \leq 1500$ ).

**Izlaz**

Na standardnom izlazu ispisuju se redom (od najmanjeg do najvećeg) svi parni trocifreni brojevi, u svakoj liniji po jedan broj.

**Primer****Ulaz**

85  
109

**Izlaz**

100  
102  
104  
106  
108

**3.1.1.3. Brojevi obrnuti redosled**

Napisati program koji odbrojava unazad od datog broja do nule.

**Ulaz**

Sa standardnog ulaza unosi se prirodan broj  $a$  manji od 100 od kojeg počinje odbrojavanje.

**Izlaz**

Na standardnom izlazu ispisuju se redom (od najvećeg do najmanjeg) svi brojevi od  $a$  do nule. Svaki broj prikazati u posebnoj liniji.

**Primer****Ulaz**

5

**Izlaz**

5

4

3

2

1

0

**3.1.1.4. Najave emisije u pravilnim vremenskim intervalima**

Poznat je početak i kraj jednog filma (vremena u satima i minutima). U pravilnim vremenskim intervalima, prvo na početku filma i zatim nakon svakih  $m$  minuta u uglu ekrana se prikazuje najava sledeće emisije. Napiši program koji ispisuje vremena u kojima se prikazuje ta najava.

**Ulaz**

Na standardnom ulazu nalazi se vreme početka filma dato u obliku dva cela broja, svakog u zasebnoj liniji koji predstavljaju broj sati i broj minuta, zatim, u sledećoj liniji vreme završetka filma dato u istom obliku i na kraju jedan ceo broj koji predstavlja interval u minutima u kojem se prikazuje najava.

**Izlaz**

Na standardnom izlazu prikazati, vremena emitovanja najave u obliku  $h:m$ , svako u zasebnom redu.

**Primer****Ulaz**

12

0

13

53

15

**Izlaz**

12:0

12:15

12:30  
 12:45  
 13:0  
 13:15  
 13:30  
 13:45

### 3.1.1.5. Geometrijska serija

Napisati program koji za date prirodne brojeve  $a$ ,  $b$  ispisuje brojeve iz intervala  $[a,b]$ , od kojih je prvi broj koji se ispisuje jednak  $a$ , a svaki sledeći je tri puta veći od prethodnog. Na primer, za  $[a,b]=[5,50]$  treba ispisati 5, 15, 45.

#### **Ulaz**

Sa standardnog ulaza se učitavaju prirodni brojevi  $a$  ( $1 \leq a \leq 50$ ) i  $b$  ( $a \leq b \leq 10000$ ) svaki u posebnom redu.

#### **Izlaz**

Na standardnom izlazu ispisuju se svi traženi brojevi, redom (od najmanjeg do najvećeg). Svaki broj ispisati u posebnoj liniji.

#### **Primer**

##### **Ulaz**

5  
 50

##### **Izlaz**

5  
 15  
 45

### 3.1.1.6. Brojanje u igri žmurke

U igri žmurke deca obično broje po pet (5,10,15,20,...). Napiši program koji ispisuje baš te brojeve.

#### **Ulaz**

Sa standardnog ulaza unosi se broj  $x$  ( $100 \leq x \leq 1000$ ) deljiv sa 5.

#### **Izlaz**

Na standardni izlaz ispisati brojeve deljive sa 5, počevši od 5 i završivši sa  $x$ . Svaki broj ispisati u posebnom redu.

#### **Primer**

##### **Ulaz**

30

##### **Izlaz**

5

10  
15  
20  
25  
30

### 3.1.1.7. Podela intervala na jednake delove

Napisati program kojim se ispisuju vrednosti  $n$  ravnomerno razmaknutih realnih brojeva iz intervala  $[a,b]$ , tako da je prvi od tih brojeva  $a$ , a poslednji  $b$ .

#### **Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $n$  ( $1 < n \leq 20$ ), druga linija sadrži realan broj  $a$ , a treća linija realan broj  $b$ , pri čemu je  $a < b$ .

#### **Izlaz**

Na standardnom izlazu prikazati, na pet decimala, redom tražene brojeve svaki u posebnoj liniji.

#### **Primer**

##### **Ulaz**

5  
-1  
1

##### **Izlaz**

-1.00000  
-0.50000  
0.00000  
0.50000  
1.00000

## 3.1.2. Učitavanje

### 3.1.2.1. Zbir $n$ brojeva

Napisati program kojim se određuje zbir  $n$  datih celih brojeva.

#### **Ulaz**

U prvoj liniji standardnog ulaza nalazi se prirodan broj  $n$ . U svakoj od narednih  $n$  linija nalazi se po jedan ceo broj  $x_i$ .

#### **Izlaz**

U prvoj liniji standardnog izlaza prikazati zbir unetih  $n$  celih brojeva  $x_1, \dots, x_n$ .

#### **Primer**

##### **Ulaz**

5



10  
-3  
2  
4  
-1

**Izlaz**

12

### 3.1.2.2. Čitanje do nule

Unose se celi brojevi dok se ne unese nula. Napisati program kojim se prikazuje koliko je uneto brojeva, ne računajući nulu.

**Ulaz**

Svaka linija standardnog ulaza, izuzev poslednje, sadrži ceo broj različit od nule. Poslednja linija sadrži nulu.

**Izlaz**

Na standardnom izlazu u prvoj liniji prikazati kolko je učitano brojeva, ne računajući nulu.

#### Primer 1

**Ulaz**

5  
12  
-5  
675  
-123  
0

**Izlaz**

5

#### Primer 2

**Ulaz**

0

**Izlaz**

0

### 3.1.2.3. Čitanje do kraja ulaza

Sa ulaza se unose celi brojevi sve dok se ne dođe do kraja ulaza. Napisati program kojim se prikazuje koliko je uneto brojeva.

**Ulaz**

Svaka linija standardnog ulaza sadrži po jedan ceo broj. NAPOMENA: prilikom interaktivnog testiranja programa, kraj standardnog ulaza se označava

kombinacijom tastera ctrl + z ako se koristi operativni sistem Windows tj. ctrl + d ako se koristi operativni sistem Linux.

**Izlaz**

U prvoj liniji standardnog izlaza prikazati koliko je brojeva uneto.

**Primer**

**Ulaz**

20  
145  
-23  
12  
456  
13

**Izlaz**

6

### 3.1.2.4. Prosek svih brojeva do kraja ulaza

Sa standardnog ulaza se učitava broj poena takmičara na takmičenju iz programiranja. Napiši program koji izračunava prosečan broj poena svih takmičara.

**Ulaz**

Svaki red standardnog ulaza sadrži jedan ceo broj između 0 i 100. NAPOMENA: prilikom interaktivnog testiranja programa, kraj standardnog ulaza se označava kombinacijom tastera ctrl + z ako se koristi operativni sistem Windows tj. ctrl + d ako se koristi operativni sistem Linux.

**Izlaz**

Na standardni izlaz ispisati prosek zaokružen na 5 decimala.

**Primer**

**Ulaz**

1  
2  
3  
4  
5  
6

**Izlaz**

3.50000

### 3.1.2.5. Čitanje do minus 1 ili do n-tog

Pera se tokom časa igrao i seckao kvadrata od papira. Učiteljica je to primetila i dala mu je zadatak da izračuna ukupnu površinu svih kvadratića. Napiši

program koji Peri pomaže da to uradi. Pera, na samom početku unosi broj  $n$  za koji je siguran da je veći ili jednak broju kvadratića. Nakon toga, unosi najviše  $n$  brojeva koji predstavljaju dužine stranica kvadratića, pri čemu, ako primeti da kvadratića ima manje od broja  $n$ , on unosi  $-1$ , čime označava da je potrebno prekinuti unos.

#### **Ulaz**

Sa standardnog ulaza se učitava broj  $n$  ( $0 \leq n \leq 15$ ), a zatim  $1 \leq m \leq n$  celih brojeva između 1 i 10, pri čemu, ako je  $m < n$ , onda se nakon njih unosi  $-1$ .

#### **Izlaz**

Na standardni izlaz ispisati jedan ceo broj koji predstavlja ukupnu površinu svih kvadrata.

### **Primer 1**

#### **Ulaz**

5  
1  
2  
3  
4  
5

#### **Izlaz**

55

### **Primer 2**

#### **Ulaz**

5  
1  
2  
3  
-1

#### **Izlaz**

14

## **3.1.3. Preslikavanje**

### **3.1.3.1. Magacin sokova**

U magacin se na  $n$  polica slažu gajbice sa sokovima. U svaku gajbicu može da stane  $k$  flaša sokova. Odrediti najmanji ukupan broj gajbica potreban da se spakuju sve flaše iz magacina, ako je poznat broj flaša sokova na svakoj polici, pri čemu se u svaku gajbicu mogu pakovati isključivo flaše sa jedne police.

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se prirodan broj  $n$  ( $1 \leq n \leq 100$ ) koji predstavlja broj polica u magacinu. U drugoj je prirodan broj  $k$  ( $5 \leq k \leq 30$ ) koji predstavlja kapacitet gajbice. U svakoj od narednih  $n$  linija nalazi se prirodan broj, iz intervala  $[5, 500]$ , koji predstavlja broj flaša na svakoj polici.

**Izlaz**

U jednoj liniji standardnog izlaza ispisuje se najmanji broj upotrebljenih gajbica.

**Primer****Ulaz**

3  
8  
31  
25  
16

**Izlaz**

10

Za flaše sa prve police potrebno je 4 gajbice, sa druge isto 4, a sa treće 2.

**3.1.3.2. Tabeliranje funkcije**

Automobil se kreće ravnomerno ubrzano sa početnom brzinom  $v_0$  izraženom u metrima u sekundi (m/s) i ubrzanjem  $a$  izraženim u metrima u sekundi za sekundu ( $\text{m/s}^2$ ). Ukupno vreme do postizanja maksimalne brzine je  $T$  sekundi. Na svakih  $\Delta t$  sekundi od početka potrebno je izračunati pređeni put automobila. Napomena: za ravnomerno ubrzano kretanje pređeni put nakon proteklog vremena  $t$  izražava se sa  $s = v_0 \cdot t + a \cdot t^2 / 2$ .

**Ulaz**

Sa standardnog ulaza učitavaju se 4 realna broja:

$v_0$  ( $0 \leq v_0 \leq 5$ ) - početna brzina

$a$  ( $1 \leq a \leq 3$ ) - ubrzanje

$T$  ( $5 \leq T \leq 10$ ) - ukupno vreme

$\Delta t$  ( $0.1 \leq \Delta t \leq 2.5$ ) - interval

**Izlaz**

Na standardni izlaz ispisati seriju brojeva koji predstavljaju pređeni put u zadatim trenucima.

**Primer****Ulaz**

1  
1  
2  
0.5

**Izlaz**

0.00000  
 0.62500  
 1.50000  
 2.62500  
 4.00000

**3.1.3.3. Milje u kilometre**

Milja je engleska istorijska mera za dužinu koja iznosi 1609.344 m. Napiši program koji ispisuje tablicu preračunavanja milja u kilometre.

**Ulaz**

Sa standardnog ulaza se unosi broj  $a$  ( $1 \leq a \leq 10$ ),  $b$  ( $10 \leq b \leq 100$ ) i  $k$  ( $1 \leq k \leq 10$ ).

**Izlaz**

Na standardni izlaz ispisati tabelu konverzije milja u kilometre za svaki broj milja iz intervala  $[a, b]$ , sa korakom  $k$ . Broj kilometara zaokružiti na 6 decimala, a tabelu prikazati u formatu identičnom kao u primeru.

**Primer****Ulaz**

10  
 20  
 2

**Izlaz**

10 mi = 16.093440 km  
 12 mi = 19.312128 km  
 14 mi = 22.530816 km  
 16 mi = 25.749504 km  
 18 mi = 28.968192 km  
 20 mi = 32.186880 km

**3.1.4. Elementarne statistike****3.1.4.1. Faktorijel**

Date su cifre  $1, 2, \dots, n$ . Napiši program koji izračunava koliko se različitih  $n$ -tocifrenih brojeva sastavljenih od svih tih cifara može napraviti (na primer, od cifara  $1, 2, 3$  mogu se napraviti brojevi  $123, 132, 213, 231, 312$  i  $321$ ).

Napomena: Broj permutacija skupa od  $n$  elemenata jednak je faktorijelu broja  $n$  tj. broju  $n! = 1 \cdot 2 \cdot \dots \cdot n$ . Razmisli zašto je baš tako.

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $n$  ( $1 \leq n \leq 9$ ).

**Izlaz**

U prvoj liniji standardnog izlaza prikazati broj različitih brojeva koji se mogu napraviti od cifara 1,2,...,n.

**Primer 1****Ulaz**

5

**Izlaz**

120

**Primer 2****Ulaz**

9

**Izlaz**

362880

**3.1.4.2. Sredine**

Automobil putuje menjajući brzinu tokom putovanja. Poznato je da se jedan deo puta kretao ravnomerno brzinom  $v_1$  km/h, zatim se jedan deo puta kretao ravnomerno brzinom  $v_2$  km/h, i tako dalje, sve do poslednjeg dela puta gde se kretao ravnomerno brzinom od  $v_n$  km/h. Napisati program koji određuje prosečnu brzinu automobila na tom putu i to:

ako se pretpostavi da je svaki deo puta trajao isto vreme,

ako se pretpostavi da je na svakom delu puta automobil prešao isto rastojanje.

**Ulaz**

Sa standardnog ulaza unosi se  $n$  ( $2 \leq n \leq 10$ ) pozitivnih realnih brojeva:  $v_1, v_2, \dots, v_n$  (za svako  $v_i$  važi  $30 \leq v_i \leq 120$ ), nakon čega sledi kraj ulaza.

**Izlaz**

U prvoj liniji standardnog izlaza ispisati realan broj zaokružen na 2 decimale koji predstavlja prosečnu brzinu pod pretpostavkom da je svaki deo puta trajao isto vreme, a u drugom redu realan broj zaokružen na 2 decimale koji predstavlja prosečnu brzinu pod pretpostavkom da je na svakom delu puta automobil prešao isto rastojanje.

**Primer****Ulaz**

60

40

**Izlaz**

50.00

48.00

### 3.1.4.3. Prosečan rast cena

Potrošačka korpa je koštala 60000 dinara. Nakon godinu dana cene su porasle 21 procenat i potrošačka korpa je koštala 72600 dinara. Nakon druge godine cene su porasle još 44 procenta i potrošačka korpa je koštala 104544 dinara. Postavlja se pitanje koliko je prosečno godišnje porasla cena potrošačke korpe. Definišimo da će prosečan procenat porasta cene biti onaj koji, kada se primeni nakon svake godine, na kraju dati istu cenu kao i u slučaju polaznih porasta cena. U našem primeru, ako bi cena porasla 32 procenta nakon godinu dana potrošačka korpa bi koštala 79200 dinara, i ako bi nakon druge godine porasla još 32 procenta, na kraju bi koštala ponovo tačno 104544 dinara, tako da kažemo da je prosečan porast cene 32 procenta. Napiši program koji za date pojedinačne procenat porasta cene nakon svake godine izračunava prosečan procenat porasta cene tokom celog perioda.

#### **Ulaz**

Sa standardnog ulaza se unosi broj  $n$  (važi  $2 \leq n \leq 10$ ) a zatim  $n$  realnih brojeva  $p_1, p_2, \dots, p_n$  koji predstavljaju procenat porasat cene na kraju svake godine (za svako  $p_i$  važi da je  $5 \leq p_i \leq 50$ ).

#### **Izlaz**

Na standardni izlaz prikazati jedan realan broj zaokružen na dve decimale koji predstavlja prosečan procenat porasta cene.

#### **Primer 1**

##### **Ulaz**

2  
21  
44

##### **Izlaz**

32.00

#### **Primer 2**

##### **Ulaz**

5  
47.0  
13.0  
13.5  
26.5  
16.5

##### **Izlaz**

22.68

### 3.1.4.4. Jednakost rastojanja

Stanovnici jedne dugačke ulice žele da odrede položaj na kojem će biti napravljena antena za mobilnu telefoniju. Pošto žele da lokaciju odrede na

najpravedniji mogući način, dogovorili su se da antenu sagrađe na mestu na kom će zbir rastojanja svih onih koji se nalaze levo od antene do nje, biti jednak zbiru rastojanja svih onih koji se nalaze desno od antene do nje. Ako su poznate koordinate svih kuća u ulici (možemo zamisliti da su to koordinate tačaka na jednoj pravoj), napiši program koji određuje položaj antente.

#### **Ulaz**

Sa standardnog ulaza u prvoj liniji se unosi prirodan broj  $n$  ( $1 \leq n \leq 100$ ) koji predstavlja broj stanara, a u narednih  $n$  linija realni brojevi (od  $-1000$  do  $1000$ ) koji predstavljaju koordinate stanara ( $x$  koordinate tačaka na osi).

#### **Izlaz**

Na standardni izlaz ispisati jedan realan broj koji predstavlja traženi položaj antene (dopuštena je tolerancija greške  $10^{-5}$ ).

### **Primer**

#### **Ulaz**

5  
-7.34  
15.6  
3.67  
-22.17  
6.2

#### **Izlaz**

-0.80800

### **3.1.4.5. Težište**

Težište  $T$  trougla  $ABC$  je presek njegovih težišnih duži. Ta tačka ima i druga zanimljiva svojstva. Na primer, kada bismo izrezali trougao od papira, mogli bismo da ga balansiramo na vrhu olovke, samo ako bismo olovku postavili u težište trougla. Takođe, težište je tačka takva da je zbir kvadrata rastojanja između nje i temena trougla (tj. izraz  $|AT|^2 + |BT|^2 + |CT|^2$ ) najmanji moguć. Ujedno, to je jedina tačka takva da je zbir vektora  $\mathbf{AT} + \mathbf{BT} + \mathbf{CT} = \mathbf{0}$ . Koristeći poslednje pomenuto svojstvo, pojam težišta se može uopštiti na proizvoljan konačan skup tačaka (tada se ponekad naziva i baricentar ili centroid). Napiši program koji za  $n$  tačaka ravni zadatih svojim koordinatama određuje njihovo težište.

#### **Ulaz**

Sa standardnog ulaza učitava se broj  $3 \leq n \leq 100$  i zatim  $2n$  parova realnih brojeva  $(x_i, y_i)$ , pri čemu je svaki broj u posebnom redu, koji predstavljaju koordinate  $n$  tačaka ravni.

#### **Izlaz**

Na standardni izlaz ispisati koordinate težišta tog skupa tačaka, prvo koordinatu  $x$ , zatim koordinatu  $y$ , svaku u posebnom redu, zaokruženu na pet decimala.



**Primer****Ulaz**

3  
0  
0  
0  
1  
1  
0

**Izlaz**

0.33333  
0.33333

**3.1.4.6. Stepen**

Napiši program koji izračunava stepen  $x^n$ . Pokušaj da program napišeš bez upotrebe bibliotečkih funkcija i operatora za stepenovanje.

**Ulaz**

Sa standardnog ulaza se unosi realan broj  $x$  ( $0.8 \leq x \leq 1.2$ ) i ceo broj  $n$  ( $0 \leq n \leq 20$ ).

**Izlaz**

Na standardni izlaz ispiši vrednost  $x^n$  zaokruženu na pet decimala.

**Primer****Ulaz**

1.1  
5

**Izlaz**

1.61051

**3.1.5. Maksimum minimum****3.1.5.1. Najmanja temperatura**

Date su temperature u nekoliko gradova. Napisati program kojim se određuje, kolika je temperatura u najhladnijem gradu.

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $n$  ( $3 \leq n \leq 50$ ) koji predstavlja broj gradova, a u svakoj od narednih  $n$  linija nalazi se ceo broj  $t$  ( $t \in [-20, 40]$ ) koji predstavlja temperaturu u odgovarajućem gradu.

**Izlaz**

Na standardnom izlazu, u jednoj liniji, prikazati temperaturu u najhladnijem gradu.

**Primer****Ulaz**

5  
10  
-12  
22  
-13  
15

**Izlaz**

-13

**3.1.5.2. Prosek skokova**

Marko je veliki obožavatelj skokova u vodu i ne propušta nijedno takmičenje. Skok se ocenjuje tako što svaki od  $n$  sudija donese svoju ocenu, a zatim se najslabija i najbolja ocena odbace i odredi se prosek preostalih ocena. Napisati program kojim se određuje Markov rezultat na osnovu sudijskih ocena.

**Ulaz**

U  $n$  ( $2 < n \leq 10$ ) linija standardnog ulaza nalazi se po jedan ceo broj iz intervala  $[0,10]$ , ti brojevi predstavljaju ocene koje je Marko dobio. Nakon toga dolazi kraj ulaza (on se može uneti pomoću `ctrl + z` u sistemima Windows ili `ctrl + d` u sistemima Linux).

**Izlaz**

U prvoj liniji standardnog izlaza prikazati, na dve decimale, Markov rezultat.

**Primer****Ulaz**

8  
7  
9  
9  
5

**Izlaz**

8.00

**3.1.5.3. Redni broj maksimuma**

Na aukciji neke slike učestvuje  $n$  kupaca. Svaki kupac ponudi izvesni iznos novca, pri tome svi kupci su ponudili različite iznose. Napisati program kojim se određuje redni broj kupca koji je ponudio najveći iznos.

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $n$  ( $1 \leq n \leq 1000$ ) koji predstavlja broj kupaca. U narednih  $n$  linija nalazi se po jedan pozitivan realan broj, ti brojevi predstavljaju iznose novca koji su ponudili kupci redom.

**Izlaz**

U prvoj liniji standardnog izlaza prikazati redni broj kupca koji je ponudio najveći iznos novca.

**Primer****Ulaz**

5  
123.23  
234.45  
100.23  
345.00  
128.80

**Izlaz**

4

**3.1.5.4. Najbliži datom celom broju**

Učenci jednog odeljenja su organizovali nagradnu igru. Pobednik igre je onaj učenik koji što bolje proceni koliko metara je dugačka njihova učionica. Ako su dve procene takve da za isti iznos potcenjuju tj. precenjuju dužinu učionice, pobednik je onaj ko je dao veću procenu.

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se ceo broj  $x$  ( $10 \leq x \leq 20$ ), stvarna dužina učionice, u drugoj liniji prirodan broj  $n$  ( $3 \leq n \leq 30$ ) koji predstavlja broj učenika, dok sledećih  $n$  linija sadrže različite realne brojeve  $x_i$  ( $5 \leq x_i \leq 30$ ) koji predstavljaju procene učenika.

**Izlaz**

U prvoj liniji standardnog izlaza prikazati, najveći broj među učitanim realnim brojevima koji je najbliži broju  $x$  (zaokružen na dve decimale).

**Primer****Ulaz**

13.0  
4  
12.4  
8.4  
13.6  
7.5

**Izlaz**

13.60

**3.1.5.5. Broj maksimalnih**

Na takmičenju iz matematike učestvuje  $n$  učenika. Za svakog takmičara poznat je broj poena koje je osvojio na takmičenju. Napisati program kojim se određuje broj takmičara koji su postigli najbolji rezultat na takmičenju.

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $n$  ( $10 \leq n \leq 250$ ) koji predstavlja broj učenika. U narednih  $n$  linija nalazi se po jedan ceo broj iz intervala  $[0,100]$ , ti brojevi predstavljaju poene koje su takmičari osvojili.

**Izlaz**

U prvoj liniji standardnog izlaza prikazati ukupan broj takmičara koji su postigli najbolji rezultat na takmičenju.

**Primer****Ulaz**

10  
57  
43  
57  
90  
73  
89  
90  
75  
90  
85

**Izlaz**

3

**3.1.5.6. Drugi na rang listi**

Na osnovu rezultata takmičenju na kojem je učestvovalo  $N$  učenika formirana je rang lista. Rang lista se formira u nerastućem poretku po rezultatima, od najboljeg do najlošijeg rezultata. Ako dva takmičara imaju isti rezultat onda se oni prikazuju u abecednom poretku, po njihovom prezimenu i imenu. Napisati program kojim se određuje broj poena takmičara koji je drugi na rang listi.

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $N$  ( $1 \leq N \leq 50000$ ) koji predstavlja broj takmičara. U svakoj od narednih  $N$  linija nalazi se ceo broj iz intervala  $[0,50000]$ , ti brojevi predstavljaju rezultate takmičara, koji nisu sortirani po poenima, već po brojevima računara za kojima su takmičari sedeli.

**Izlaz**

U jednoj liniji standardnog izlaza prikazati broj poena drugog takmičara na rang listi.

**Primer****Ulaz**

5  
80  
95  
75  
50  
95

**Izlaz**

95

**3.1.5.7. Druga vrednost po veličini**

U studentskom udruženju se organizuje nagradna igra. Dogovor je da nagrade dobijaju dva člana udruženja koja imaju najveće brojeve indeksa. Problem je u tome što su na postojećem spisku neki studenti upisani i više puta. Napisati program koji na osnovu serije brojeva indeksa sa tog spiska određuje dva nagrađena studenta.

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $N$  ( $10 \leq N \leq 50000$ ) koji predstavlja broj studenata na spisku. Svaka od narednih  $N$  linija sadrži po jedan prirodan broj iz intervala  $[1, 50000]$ , koji predstavlja broj indeksa studenata iz udruženja. Može se pretpostaviti da će na spisku postojati bar dva različita indeksa.

**Izlaz**

Na standardnom izlazu prikazati indekse dva nagrađena studenta. U prvom redu najveću, a u drugom redu sledeću po veličini vrednost među brojevima indeksa.

**Primer****Ulaz**

7  
3  
15  
3  
15  
71  
71  
3

**Izlaz**

71

15

**3.1.5.8. Pobednik u tri discipline**

Takmičari su radili testove iz programiranja, matematike i fizike. Za svaki predmet dobili su određeni broj poena (ceo broj od 0 do 50). Takmičari se rangiraju po ukupnom broju poena iz sva tri predmeta. Ako dva takmičara imaju isti broj poena, pobednik je onaj koji ima više poena iz programiranja. Ako je i broj poena iz programiranja jednak, onda se posmatra broj poena iz matematike. Na kraju, ako je i broj poena iz matematike jednak, posmatra se broj poena iz fizike. Potrebno je napisati program koji određuje pobednika takmičenja.

**Ulaz**

Učitava se prvo broj takmičara, a zatim i podaci za svakog takmičara. Za svakog takmičara učitava se broj poena iz programiranja, a zatim broj poena iz matematike, i na kraju broj poena iz fizike, svi u istom redu.

**Izlaz**

Potrebno je ispisati redni broj pobednika i broj njegovih poena iz svakog predmeta. Brojanje takmičara počinje od 1. Ako su dva ili više takmičara ostvarila potpuno identičan uspeh, pobednik je onaj koji ima manji redni broj (jer je ostvario više poena na kvalifikacionom takmičenju).

**Primer****Ulaz**

4

20 30 40

10 20 30

20 40 30

10 50 20

**Izlaz**

3: 20 40 30

**3.1.5.9. Najmanji krug**

Za dati niz tačaka u ravni odrediti poluprečnik najmanjeg kruga, sa centrom u koordinatnom početku, koji sadrži sve date tačke.

**Ulaz**

Sa standardnog ulaza u prvoj liniji se unosi prirodan broj  $n$  ( $1 \leq n \leq 100$ ) koji predstavlja broj tačaka u ravni, a u narednih  $2 \cdot n$  linija u svakoj liniji po jedan realan broj (od  $-1000$  do  $1000$ ), koji predstavljaju koordinate tačke u ravni.  $x$  i  $y$  se naizmenično učitavaju.

**Izlaz**

Na standardni izlaz ispisati jedan realan broj (dopuštena je tolerancija greške  $10^{-5}$ ) - poluprečnik najmanjeg kruga sa centrom u koordinatnom početku, koji sadrži sve učitane tačke.

**Primer****Ulaz**

4  
2.6  
3.4  
1.2  
6.7  
3.33  
5.55  
2.57  
3.44

**Izlaz**

6.80661

**3.1.5.10. Razlika suma do max i od max**

Unosi se mase predmeta, odrediti razliku sume masa predmeta do prvog pojavljivanja predmeta najveće mase i sume masa predmeta posle prvog pojavljivanja predmeta najveće mase (predmet najveće mase nije uključen ni u jednu sumu).

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se broj predmeta  $n$  ( $1 \leq n \leq 50000$ ). Svaka od narednih  $n$  linija sadrži po jedan prirodan broj iz intervala  $[1, 50000]$ , ti brojevi predstavljaju mase svakog od  $n$  predmeta.

**Izlaz**

U prvoj liniji standardnog izlaza prikazati traženu razliku masa.

**Primer****Ulaz**

5  
10  
13  
7  
13  
4

**Izlaz**

-14

Predmet najveće mase je 13. Zbir masa pre njegovog prvog pojavljivanja je 10, a posle njegovog prvog pojavljivanja je 24. Zato je tražena razlika -14.

### 3.1.6. Pretraga

#### 3.1.6.1. Prva negativna temperatura

Poznate su temperature za svaki dan nekog perioda. Napisati program kojim se određuje redni broj dana u tom periodu kada je temperatura prvi put bila negativna.

##### *Ulaz*

Prva linija standardnog ulaza sadrži prirodan broj  $n$  ( $3 \leq n \leq 365$ ) koji predstavlja broj dana perioda. Svaka od narednih  $n$  linija sadrži po jedan ceo broj iz intervala  $[-5, 40]$ , brojevi predstavljaju temperature redom za  $n$  dana perioda.

##### *Izlaz*

U prvoj liniji standardnog izlaza prikazati redni broj dana perioda kada je temperatura prvi put bila negativna (dani se broje od 1 do  $n$ ), ako takav dan ne postoji prikazati -1.

#### Primer 1

##### *Ulaz*

5  
12  
10  
-2  
-3  
4

##### *Izlaz*

3

#### Primer 2

##### *Ulaz*

3  
12  
10  
14

##### *Izlaz*

-1

#### 3.1.6.2. Poslednja negativna temepartura

Poznate su temperature za svaki dan nekog perioda. Napisati program kojim se određuje redni broj dana u tom periodu kada je temperatura poslednji put bila negativna (dani se broje od 1 do  $n$ ).



**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $n$  ( $3 \leq n \leq 365$ ) koji predstavlja broj dana perioda. Svaka od narednih  $n$  linija sadrži po jedan ceo broj iz intervala  $[-10, 40]$ , brojevi predstavljaju temperature redom za  $n$  dana perioda.

**Izlaz**

U prvoj liniji standardnog izlaza prikazati redni broj dana perioda kada je temperatura poslednji put bila negativna, ako takav dan ne postoji prikazati -1.

**Primer 1****Ulaz**

5  
2  
-8  
-2  
-3  
4

**Izlaz**

4

**Primer 2****Ulaz**

3  
12  
10  
14

**Izlaz**

-1

**3.1.6.3. Deljiv brojevima od 1 do  $n$** 

Napisati program kojim se proverava da li je dati prirodan broj  $k$  deljiv svim prirodnim brojevima od 1 do  $n$ .

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $k$  ( $0 < k \leq 10^9$ ), a druga linija sadrži prirodan broj  $n$  ( $1 \leq n \leq 10$ ).

**Izlaz**

U prvoj liniji standardnog izlaza prikazati reč da ako je broj  $k$  deljiv svim prirodnim brojevima od 1 do  $n$ , inače prikazati reč ne.

**Primer 1****Ulaz**

60

6

**Izlaz**

da

**Primer 2****Ulaz**

90

5

**Izlaz**

ne

**3.1.6.4. Negativan broj**

Napisati program kojim se proverava da li među učitanim brojevima postoji negativan broj.

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $n$  ( $1 \leq n \leq 100$ ) koji predstavlja broj brojeva. Svaka od narednih  $n$  linija sadrži po jedan realan broj.

**Izlaz**

Na standardnom izlazu u jednoj liniji ispisti postoji ako među učitanim brojevima postoji negativan broj, inače ispisati ne postoji.

**Primer 1****Ulaz**

5

10.89

12.349

-5.9

2.3

-2.45

**Izlaz**

postoji

**Primer 2****Ulaz**

4

100.89

12.349

0

2.3

**Izlaz**

ne postoji

### 3.1.6.5. Parno neparni

Pera se igra sa kartama. Sve karte koje je imao u ruci je složio tako da prvo idu sve karte sa parnim brojevima, a zatim one sa neparnim brojevima (moguće je i da je Pera imao samo parne ili samo neparne karte). Napiši program koji proverava da li je Pera ispravno složio karte.

#### **Ulaz**

Sa standardnog ulaza učitavaju se brojevi karata (prirodni brojevi između 2 i 10) sve dok se ne dođe do kraja ulaza (on se može uneti sa Ctrl+Z tj. Ctrl+D). Karata ima najmanje dve, a najviše deset.

#### **Izlaz**

Na standardni izlaz ispisati da ako je Pera dobro složio karte tj. ne u suprotnom.

#### **Primer 1**

##### **Ulaz**

2  
6  
4  
5  
3

##### **Izlaz**

da

#### **Primer 2**

##### **Ulaz**

2  
6  
3  
5  
4

##### **Izlaz**

ne

### 3.1.6.6. Prvi i poslednji pristup

Sa računara u školi na sajt škole se loguju razni đaci. Svaki put kada se neko uloguje to se pamti u bazi. Napiši program koji za dati broj računara određuje koji je prvi, a koji je poslednji đak koji se logovao.

#### **Ulaz**

Sa standardnog ulaza se u prvoj liniji unosi redni broj računara koji nas zanima (ceo broj od 1 do 100), u drugoj liniji se nalazi ukupan broj  $n$  ( $1 \leq n \leq 50000$ ) logovanja, a zatim se u narednih  $n$  linija nalaze podaci o pojedinačnim logovanjima: broj računara sa kojeg se neki đak logovao (ceo broj od 1 do

100) i korisničko ime tog đaka (niska od najviše 20 malih slova engleske abecede).

#### **Izlaz**

Na standardni izlaz ispisati ime prvog i ime poslednjeg đaka koji se logovao na taj računar (svako u posebnom redu) ili tekst nema ako se sa tog računara niko nije logovao.

#### **Primer**

##### **Ulaz**

```
3
7
1 zika
3 jovana
2 ana
3 pera
1 lucija
2 ana
3 milica
```

##### **Izlaz**

```
jovana
milica
```

### **3.1.7. Filtriranje**

#### **3.1.7.1. Prosek odličnih**

Date su prosečne ocene  $n$  učenika jednog odeljenja. Napisati program kojim se određuje prosek prosečnih ocena svih odličnih učenika tog odeljenja.

##### **Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $n$  ( $1 \leq n \leq 100$ ) koji predstavlja broj učenika. U narednih  $n$  linija nalazi se po jedan realan broj iz intervala  $[2,5]$ . Ti brojevi predstavljaju proseke učenika.

##### **Izlaz**

U prvoj liniji standardnog izlaza prikazati prosek prosečnih ocena odličnih učenika odeljenja zaokružen na dve decimale. Ako odličnih učenika nema prikazati -.

#### **Primer 1**

##### **Ulaz**

```
4
3.5
4.75
3
```

4.5

**Izlaz**

4.62

**Primer 2****Ulaz**

4

3.5

3.75

2.80

4.35

**Izlaz**

-

**3.1.7.2. Brojevi deljivi sa 3**

Napiši program koji među unetim brojevima određuje i ispisuje one koji su deljivi sa 3.

**Ulaz**

Sa standardnog ulaza se najpre učitava prirodan broj  $n$  ( $1 \leq n \leq 1000$ ), a potom i  $n$  prirodnih brojeva iz intervala  $[1, 10^8]$ , svaki u posebnom redu.

**Izlaz**

Na standardnom izlazu ispisuju se redom svi učitani brojevi deljivi sa 3. Svaki broj ispisati u posebnoj liniji.

**Primer****Ulaz**

7

100

11

12

18

200

300

102

**Izlaz**

12

18

300

102

### 3.1.7.3. Broj pojavljivanja

Učenici u jednoj školi birali su svog omiljenog glumca. Đoku zanima koliko je glasova dobio njegov omiljeni glumac. Napiši program koji mu pomaže da to odredi.

#### **Ulaz**

Svaki glumac je označen prirodnim brojem od 1 do 10. Prva linija standardnog ulaza sadrži redni broj  $x$  Đokinog omiljenog glumca. Druga linija standardnog ulaza sadrži prirodan broj  $n$  ( $1 \leq n \leq 100$ ) - broj učenika u školi. Narednih  $n$  linija sadrže glasove svakog učenika (brojeve od 1 do 10, svaki u posebnom redu).

#### **Izlaz**

Na standardni izlaz ispisati jedan prirodan broj - broj glasova koje je dobio glumac  $x$ .

#### **Primer**

##### **Ulaz**

2  
6  
1  
2  
3  
2  
7  
2

##### **Izlaz**

3

### 3.1.7.4. Kategorije džudista

Na jednom turniru džudisti se takmiče u tri kategorije: do 50 kilograma, od 51 do 75 kilograma i od 76 kilograma naviše. Napiši program koji učitava broj džudista jednog kluba prijavljenog na taj turnir, a zatim težinu svakog od njih i za svaku kategoriju redom ispisuje koliko će se džudista tog kluba boriti u toj kategoriji.

#### **Ulaz**

Sa standardnog ulaza se učitava broj džudista  $n$  ( $1 \leq n \leq 100$ ), a zatim u  $n$  narednih redova težine džudista (celi brojevi između 30 i 120).

#### **Izlaz**

Na standardni izlaz ispisati tri broja, svaki u posebnom redu: broj džudista koji imaju manje ili jednako od 50 kilograma, broj džudista koji imaju između 51 i 75 kilograma i broj džudista koji imaju više ili jednako od 76 kilograma.

**Primer*****Ulaz***

5  
48  
51  
73  
82  
50

***Izlaz***

2  
2  
1

**3.1.8. Cifre broja****3.1.8.1. Broj i zbir cifara broja**

Napisati program kojim se određuje broj i zbir cifara dekadnog zapisa prirodnog broja.

***Ulaz***

Sa standardnog ulaza se učitava ceo broj od 0 do 1000000000.

***Izlaz***

Na standardni izlaz se ispisuju broj cifara i zbir cifara učitanoog broja.

**Primer*****Ulaz***

23645

***Izlaz***

5 20

***Ulaz***

0

***Izlaz***

1 0

**3.1.8.2. Ima li cifru**

Napisati program kojim se ispituje da li dekadni zapis prirodnog broja sadrži datu cifru?

***Ulaz***

U prvoj liniji standardnog ulaza unosi se prirodan broj od 0 do 1000000000, a u drugoj dekadna cifra.

**Izlaz**

Na standardni izlaz se ispisuje tekstualni odgovor, oblika koji je prikazan u narednim primerima.

**Primer 1****Ulaz**

29384

3

**Izlaz**

broj 29384 sadrzi cifru 3

**Primer 2****Ulaz**

29384

7

**Izlaz**

broj 29384 ne sadrzi cifru 7

**3.1.8.3. Transformacija broja u proizvod cifara**

Prirodan broj se transformiše tako što se zameni proizvodom svojih cifara. Napisati program kojim se ispisuju sve transformacije broja dok se ne dobije jednocifren broj. Odrediti i koliko transformacija je pri tome potrebno izvršiti.

**Ulaz**

Jedna linija standardnog ulaza sadrži jedan prirodan broj.

**Izlaz**

Na standardnom izlazu prikazati u svakoj liniji po jednu transformaciju datog broja (prirodan broj), do jednocifrenog broja. U poslednjoj liniji standardnog izlaza prikazati broj transformacija.

**Primer 1****Ulaz**

3274

**Izlaz**

168

48

32

6

4



**Primer 2****Ulaz**

5

**Izlaz**

0

**3.1.8.4. Najmanji broj sa najvećim zbirom parnih cifara**

Unose se prirodni brojevi (uključujući i 0) dok se ne unese -1. Među unetim brojevima koji sadrže bar jednu parnu cifru, naći najmanji broj sa najvećim zbirom parnih cifara.

**Ulaz**

Linije standardnog ulaza, njih najviše milion, sadrže po jedan prirodan broj. Poslednja linija standardnog ulaza sadrži broj -1.

**Izlaz**

Prva i jedina linija standardnog izlaza sadrži najmanji broj od unetih brojeva sa najvećim zbirom parnih cifara. Ako među unetim brojevima nema brojeve koji sadrže parne cifre na izlazu prikazati -1.

**Primer 1****Ulaz**

137

20

143

221

0

22

4

-1

**Izlaz**

4

**Primer 2****Ulaz**

137

39

155

791

731

31

-1

**Izlaz**

-1

### 3.1.8.5. Decimale broja $1/n$

Odredi prvih  $k$  decimalnih cifara vrednosti  $1/n$  za dati pozitivan prirodan broj  $n$ . Pretpostaviti da se nakon  $k$  decimala vrši odsecanje tj. da se broj zaokružuje naniže.

#### **Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $k$  ( $1 \leq k \leq 100$ ) a druga prirodan broj  $n$  ( $2 \leq n \leq 1000$ ).

#### **Izlaz**

Na standardnom izlazu, prikazujemo količnik sa traženim brojem decimala.

#### **Primer**

##### **Ulaz**

4  
33

##### **Izlaz**

0.0303

### 3.1.8.6. K decimala u binarnom zapisu

Napisati program kojim se ispisuje realan broj  $x$  ( $0 \leq x < 1$ ) u binarnom zapisu na  $k$  decimala.

#### **Ulaz**

Prva linija standardnog ulaza sadrži realan broj iz intervala  $[0,1)$ . Druga linija standardnog ulaza sadrži prirodni broj  $k$  ( $1 \leq k \leq 100$ ).

#### **Izlaz**

U prvoj i jedinoj liniji standardnog izlaza prikazati broj  $x$  u binarnom zapisu na  $k$  decimala.

#### **Primer**

##### **Ulaz**

0.8  
4

##### **Izlaz**

0.1100

### 3.1.8.7. Broj formiran od datih cifra s leva na desno

Napisati program kojim se formirati prirodan broj od učitanih cifara, ako se cifre broja učitavaju sleva na desno (redom od cifre najveće težine do cifre jedinica).

#### **Ulaz**

Svaka linija standardnog ulaza, njih najviše 9, sadrži po jednu cifru.

**Izlaz**

Na standardnom izlazu prikazati formirani broj.

**Primer****Ulaz**

4  
1  
0  
9

**Izlaz**

4109

**3.1.8.8. Broj formiran od datih cifara zdesna na levo**

Napisati program kojim se formirati prirodan broj od učitanih cifara, ako se cifre broja učitavaju zdesna na levo (redom od cifre jedinica do cifre najveće težine).

**Ulaz**

Svaka linija standardnog ulaza (njih najviše 9) sadrži po jednu cifru.

**Izlaz**

Na standardnom izlazu prikazati formirani broj.

**Primer****Ulaz**

0  
0  
3  
2  
9

**Izlaz**

92300

**3.1.8.9. Zameni cifre 0 sa 5**

Napisati program koji u datom prirodnom broju svako pojavljivanje cifre 0 zamenjuje cifrom 5.

**Ulaz**

Sa standardnog ulaza se učitava prirodan broj u granicama od 0 do 1000000000.

**Izlaz**

Na standardnom izlazu se ispisuje dobijeni broj.

**Primer****Ulaz**

20240607

**Izlaz**

25245657

**3.1.8.10. Kombinacija dva broja minimumom i maksimumom odgovarajućih cifara**

Dva data prirodna broja se kombinuju tako što se potpišu jedan ispod drugog, kao kada se sabiraju, ali se svaka cifra rezultata određuje kao manja ili kao veća od dve cifre (ako su brojevi različite dužine tada se ispred kraćeg broja dopiše onoliko nula koliko je potrebno da se brojevi izjednače po dužini). Npr. za brojeve 174 i 38, dobija se 34 ako se uzima manja od dve cifre tj. 178 ako se uzima veća.

**Ulaz**

Svaka od dve linije standardnog ulaza sadrži po jedan prirodan broj.

**Izlaz**

Na standardnom izlazu prikazati dva prirodna broja - jedan koji se dobija kombinovanjem dva broja uzimajući manju, a drugi uzimajući veću cifru na svakoj poziciji.

**Primer****Ulaz**

174

38

**Izlaz**

34

178

**3.1.8.11. Armstrongov broj**

Napisati program koji za dati prirodan broj  $n$  proverava da li je taj broj Armstrongov.  $k$ -to cifren broj je Armstrongov ako je jednak sumi  $k$ -tih stepena svojih cifara. Na primer, 370 je Armstrongov jer je  $370 = 3^3 + 7^3 + 0^3$ , 1634 je Armstrongov jer je  $1634 = 1^4 + 6^4 + 3^4 + 4^4$ , dok 12 nije Armstrongov jer je  $12 \neq 1^2 + 2^2$ .

**Ulaz**

Sa standardnog ulaza se učitava prirodan broj  $n$  ( $1 \leq n \leq 100000$ ).

**Izlaz**

Na standardnom izlazu ispisuju se poruka DA ako učitani broj jeste Armstrongov, tj. NE ako učitani broj nije Armstrongov.

**Primer 1****Ulaz**

1002

**Izlaz**

NE

**Primer 2****Ulaz**

370

**Izlaz**

DA

**3.1.8.12. Ne sadrže cifru 5**

Napiši program koji određuje koliko brojeva na standardnom ulazu ne sadrži cifru 5.

**Ulaz**

Svaka linija standardnog ulaza sadrži jedan prirodan broj iz intervala [0,1000000000].

**Izlaz**

Na standardni izlaz ispisati koliko od učitanih brojeva ne sadrži cifru 5.

**Primer****Ulaz**

12

15

155

14

**Izlaz**

2

**3.1.9. Odnos susednih elemenata serije****3.1.9.1. Rastuće cifre**

Napisati program kojim se ispituje da li su cifre dekadnog zapisa prirodnog broja rastuće idući od cifre najmanje težine (cifre jedinica)?

**Ulaz**

Sa standardnog ulaza se učitava prirodan broj od 0 do 100000.

**Izlaz**

Na standardnom izlazu se ispisuje tekstualni odgovor DA ili NE.

**Primer 1***Ulaz*

97642

*Izlaz*

DA

**Primer 2***Ulaz*

79653

*Izlaz*

NE

**3.1.9.2. Maksimalna razlika susednih**

Napisati program kojim se za  $n$  učitanih celih brojeva određuje po apsolutnoj vrednosti najveća razlika dva susedna elementa.

*Ulaz*

U prvoj ulaznoj liniji učitava se prirodan broj  $n$  ( $2 \leq n \leq 100$ ), a u sledećim  $n$  celih brojeva u intervalu  $[-100, 100]$ .

*Izlaz*

Ispisuje se traženi broj koji predstavlja najveću apsolutnu razliku dva susedna broja.

**Primer***Ulaz*

5  
-20  
30  
5  
90  
70

*Izlaz*

85

**3.1.9.3. Provera monotonosti**

Tokom nekoliko dana maratonac se sprema za predstojeću trku. Za svaki dan poznato je rastojanje koje je tog dana pretrčao. Odrediti da li je plivač pravio napredak tj. da li su učitani brojevi uređeni u strogo rastućem redosledu.

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $N$  ( $1 \leq N \leq 50$ ) koji predstavlja broj dana, a u svakoj od narednih  $N$  linija nalazi se po jedan prirodan broj koji predstavlja broj metara koji je trkač pretrčao odgovarajućeg dana.

**Izlaz**

Na standardnom izlazu ispisati odgovor Da ako su brojevi u strogo rastućem poretku i odgovor Ne u suprotnom.

**Primer 1****Ulaz**

4  
19  
20  
22  
23

**Izlaz**

Da

**Primer 2****Ulaz**

4  
20  
20  
23  
23

**Izlaz**

Ne

**3.1.9.4. Poređani datumi**

Milica je napravila spisak rođendana svojih drugarica. Poređala je datume njihovog rođenja hronološki (od najdavnijeg do najskorijeg). Napiši program koji proverava da li je Milica to dobro uradila.

**Ulaz**

Sa standardnog ulaza učitava se broj  $n$  ( $1 \leq n \leq 100$ ). U narednih  $n$  linija nalazi se  $n$  ispravnih, međusobno različitih datuma. Datumi su zapisani tako što su zapisani dan, mesec i godina, razdvojeni razmacima.

**Izlaz**

U jedinoj liniji standardnog izlaza ispisati DA ako su datumi ispravno (rastuće) poređani, tj. NE ako nisu.

**Primer****Ulaz**

3  
 3 7 2005  
 14 8 2006  
 10 5 2006

**Izlaz**

NE

**3.1.9.5. Testerast niz**

Za niz celih brojeva odrediti da li je testerast. Kažemo da je niz testerast ako za njegove članove važi poredak  $a_1 < a_2$ ,  $a_2 > a_3$ ,  $a_3 < a_4$ ,  $a_4 > a_5$  itd. ili  $a_1 > a_2$ ,  $a_2 < a_3$ ,  $a_3 > a_4$ ,  $a_4 < a_5$  itd.

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se prirodan broj  $N$  ( $2 \leq N \leq 100$ ). U sledećih  $N$  linija nalazi se po jedan ceo broj iz intervala od  $-10^{**9}$  do  $10^{**9}$ . Brojevi predstavljaju elemente niza.

**Izlaz**

Na standardni izlaz prikazati poruku DA ako niz jeste testerast. Inače prikazati poruku NE.

**Primer 1****Ulaz**

6  
 2  
 1  
 3  
 2  
 4  
 3

**Izlaz**

DA

**Primer 2****Ulaz**

6  
 1  
 2  
 3  
 4  
 5  
 6



**Izlaz**

NE

**3.1.10. Rekurentne serije****3.1.10.1. Dimenzija papira**

Standardne veličine papira se definišu na sledeći način. Površina papira A0 je  $1\text{m}^2$ . Odnos dimenzija stranica je  $1:\sqrt{2}$ . Jednostavnosti radi, dimenzije se zaokružuju na ceo broj milimetara. Papir A1 dobija se presavijanjem papira A0 na pola duž njegove duže stranice (proverite da se na taj način opet dobija odnos stranica  $1:\sqrt{2}$ ). Opet se jednostavnosti radi dimenzije zaokružuju na ceo broj milimetara. Ovaj postupak se ponavlja i tako se dobijaju papiri A2, A3, A4, A5 itd. Napiši program koji za dati broj  $n$  određuje dimenzije papira  $A_n$ .

**Ulaz**

Jedan broj  $n$  čija je vrednost između 0 i 10.

**Izlaz**

Dva prirodna broja odvojena razmakom koja predstavljaju kraću i dužu dimenziju papira  $A_n$

**Primer****Ulaz**

4

**Izlaz**

210 297

**3.1.10.2. Pčele i trutovi**

Pčela matica nosi jajašca. Ako trut oplodi jajašce pčele, tada se iz njega rađa ženska pčela. Ako se jajašce ne oplodi, onda se iz njega izleže trut. Dakle, ženska pčela ima dva roditelja, dok trut ima samo jednog (on nema oca, već samo majku). Pčela ima dve bake (maminu i tatinu mamu) i jednog dedu (maminog tatu), dok trut ima jednu baku i jednog dedu (mamine roditelje). Napiši program koji određuje koliko predaka u nekoj generaciji ima trut.

**Ulaz**

Sa standardnog ulaza se unosi broj  $n$  ( $1 \leq n \leq 20$ ) koji označava redni broj generacije: 0 je generacija samog truta, 1 je generacija njegove majke, 2 je generacija njegove bake i dede i tako dalje u prošlost.

**Izlaz**

Na standardni izlaz ispisati ukupan broj predaka truta u generaciji  $n$ .

**Primer****Ulaz**

5

**Izlaz**

8

**3.1.10.3. Procena kvadratnog korena Njutnovom metodom**

Iako smo navikli da kvadratni koren računamo pomoću digitrona i računara, pa ne razmišljamo puno kako se dolazi do precizne vrednosti korena nekog broja, u pozadini izračunavanja korena leže mnoge interesantne matematičke metode. Jedna od najjednostavnijih, a često korišćenih metoda je zasnovana na sledećem postupku. Pretpostavimo da je  $x_i$  trenutna procena vrednosti korena broja  $a$  tj.  $\sqrt{a}$ . Ako bi  $x_i$  zaista bio koren broja  $a$  tada, bi važno da su  $x_i$  i  $a/x_i$  jednaki. Ako nisu jednaki, jedan od njih je veći, a drugi manji i možemo očekivati da se vrednost korena nalazi negde između njih. Pretpostavićemo da će to biti tačno na sredini, tj. da ćemo narednu, bolju procenu te vrednosti možemo dobiti pomoću formule:

$$x_{i+1} = (x_i + a/x_i)/2.$$

Dakle, ako krenemo od neke početne vrednosti  $x_0$  (na primer, od vrednosti 1) i  $n$  puta novu vrednost izračunamo od prethodne primenom prethodne formula trebalo bi da dobijemo vrednost koja prilično precizno procenjuje  $\sqrt{a}$  (uslovi i razlozi pod kojima ovo važi zahtevaju određeno matematičko razmatranje koje prevazilazi opseg ove zbirke, ali ti preporučujemo da to samostalno proučiš). Recimo i da se ovaj postupak poklapa sa čuvenom Njutnovom metodom za određivanja nula funkcije, na osnovu koje bi važila sledeća veza:

$$x_{i+1} = x_i - (x_i^{*2} - a)/(2 \cdot x_i).$$

Napiši program koji prethodno opisanom tehnikom procenjuje koren datog broja.

**Ulaz**

Sa standardnog ulaza se unosi pozitivan realan broj  $a$  čiji se koren traži, a zatim i broj  $n$  ( $1 \leq n \leq 20$ ) koji određuje koliko puta treba izvršiti popravku vrednosti primenom prethodne formule.

**Izlaz**

Na standardni izlaz ispisati procenu vrednosti korena nakon  $n$  koraka, tj. vrednost  $x_n$ , ako je početna vrednost  $x_0=1$ . Rezultat prikazati zaokružen na 10 decimala.

**Primer****Ulaz**

10

5

**Izlaz**

3.1622777

### 3.1.10.4. Poslednje dve cifre Fibonačijevog broja

Niz Fibonačijevih brojeva 0, 1, 1, 2, 3, 5, 8, 13,... formira se na sledeći način: prva dva broja  $f[0]$  i  $f[1]$  su 0 i 1, svaki sledeći broj  $f[i]$  dobija se kao zbir prethodna dva broja  $f[i-2]+f[i-1]$ . Napisati program kojim se određuju poslednje dve cifre u Fibonačijevom broju  $f[n]$  (brojanje krećemo od 0).

#### **Ulaz**

Prva i jedina linija standardnog ulaza sadrži prirodan broj  $n$  ( $0 \leq n \leq 10^{**9}$ ).

#### **Izlaz**

U prvoj liniji standardnog izlaza prikazati cifru desetica, a u drugoj liniji cifru jedinica u Fibonačijevom broju  $f[n]$ .

#### **Primer 1**

##### **Ulaz**

16

##### **Izlaz**

8

7

#### **Primer 2**

##### **Ulaz**

5

##### **Izlaz**

0

5

#### **Primer 3**

##### **Ulaz**

300

##### **Izlaz**

0

0

#### **Primer 4**

##### **Ulaz**

301

##### **Izlaz**

0

1

## 3.2. Ugneždene petlje

### 3.2.0. Kombinatorni objekti

#### 3.2.0.1. Ciklične permutacije

Cikličnim pomeranje za jedno mesto ulevo niza brojeva  $x_1, x_2, x_3, \dots, x_n$  dobijamo  $x_2, x_3, \dots, x_n, x_1$ , ako vršimo ciklično pomeranje za dva mesta ulevo dobijamo  $x_3, \dots, x_n, x_1, x_2$ . Dobijeni nizovi predstavljaju ciklične permutacije polaznog niza.

Napisati program kojim se za dati prirodan broj  $n$  prikazuju nizovi brojeva dobijeni cikličnim pomeranjem niza brojeva  $1, 2, \dots, n$  redom za  $0, 1, 2, \dots, n-1$  mesta ulevo.

##### **Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $n \leq 30$ .

##### **Izlaz**

Standardni izlaz sadrži  $n$  linija, u kojima su prikazani traženi nizovi brojeva, brojevi u nizovima međusobno su odvojeni blanko znakom.

##### **Primer**

##### **Ulaz**

4

##### **Izlaz**

```
1 2 3 4
2 3 4 1
3 4 1 2
4 1 2 3
```

#### 3.2.0.2. Kockice za jamb

Ispisati sve rezultate bacanja tri kockice za jamb u kojima je zbir brojeva jednak zadatom broju  $X$  (prirodan broj od 3 do 18) ako je redosled kockica bitan (na primer, 2, 2, 3 nije isto kao 2, 3, 2).

##### **Ulaz**

U jednoj liniji standardnog ulaza navodi se prirodan broj  $X$  ( $3 \leq X \leq 18$ ) koji predstavlja zbir brojeva na kojima su se zaustavile kockice.

##### **Izlaz**

U svakoj liniji standardnog izlaza ispisuju se tri prirodna broja (brojevi od 1 do 6) koji predstavljaju rezultat bacanja tri kockice. Rezultate prikazati sortirane leksikografski.

**Primer****Ulaz**

7

**Izlaz**

```

1 1 5
1 2 4
1 3 3
1 4 2
1 5 1
2 1 4
2 2 3
2 3 2
2 4 1
3 1 3
3 2 2
3 3 1
4 1 2
4 2 1
5 1 1

```

**3.2.0.3. Trouglovi celobrojnih stranica, zadatog obima**

Napisati program kojim se određuju dužine stranica trouglova  $a$ ,  $b$ ,  $c$ , takve da su  $a$ ,  $b$  i  $c$  prirodni brojevi za koje važi da je  $a \leq b \leq c$  i da je obim trougla  $a+b+c$  jednak datom prirodnom broju  $O$ .

**Ulaz**

U jednoj liniji standardnog ulaza zadat je prirodan broj  $O$  ( $3 \leq O \leq 250$ ) koji predstavlja obim trougla.

**Izlaz**

U svakoj liniji standardnog izlaza nalaze se tri prirodna broja odvojena prazninom, koji predstavljaju moguće dužine stranica trougla zadatog obima. Trojke treba da budu leksikografski sortirane.

**Primer****Ulaz**

7

**Izlaz**

```

1 3 3
2 2 3

```

**3.2.0.4. Kombinacije poena**

Na košarkaškoj utakmici lopta ubačena u koš boduje se sa 1, 2 ili 3 poena. Potrebno je odrediti sve kombinacije broja koševa koji su bodovani sa 1

poenom, sa 2 i sa 3 poena ako je poznat ukupan broj poena jednog tima na košarkaškoj utakmici.

#### **Ulaz**

U prvoj liniji standardnog ulaza nalazi se ukupan broj poena na košarkaškoj utakmici (prirodan broj između 40 i 150).

#### **Izlaz**

Svaka linija standardnog izlaza sadrži, odvojeno sa po jednim razmakom (blanko karakterom), redom, broj koševa koju su bodovani sa 1 poenom, broj koševa koji su bodovani sa 2 poena i broj koševa koji su bodovani sa 3 poena. Bodovanja prikazati uređena po broju koševa sa 3 poena, od najmanjeg ka najvećem. Ako je u nekoj kombinaciji broj koševa sa 3 poena jednak, onda prvo prikazati kombinacije sa manjim brojem koševa sa 2 poena.

### **Primer**

#### **Ulaz**

10

#### **Izlaz**

```
0 0 10
0 1 8
0 2 6
0 3 4
0 4 2
0 5 0
1 0 7
1 1 5
1 2 3
1 3 1
2 0 4
2 1 2
2 2 0
3 0 1
```

### **3.2.0.5. Varijacije trojki**

Svaki od tri drugara ima određeni broj jabuka, ali nikoja dva od njih nemaju isti broj jabuka. Ako se zna najveći mogući broj jabuka koji svaki od drugara može da ima, napiši program koji ispisuje sve moguće trojke brojeva jabuka koje oni mogu da imaju.

#### **Ulaz**

Sa standardnog ulaza se unosi broj  $n$  ( $1 \leq n \leq 20$ ) - najveći broj jabuka koje svaki od drugara može da ima.

#### **Izlaz**

Na standardni izlaz ispisati sve moguće brojeve jabuka koje drugovi mogu da imaju, uređene leksikografski.

**Primer****Ulaz**

2

**Izlaz**

```

0 1 2
0 2 1
1 0 2
1 2 0
2 0 1
2 1 0

```

**3.2.0.6. Brojevi u datoj osnovi**

Brojevi u osnovi  $b$  se mogu zapisati pomoću cifara  $0, 1, \dots, b-1$ . Ako je osnova veća od 10, tada se umesto cifara koriste slova engleske abecede (cifra 10 zapisuje se sa  $a$ , cifra 11 sa  $b$  itd.). Napiši program koji ispisuje sve trocifrene brojeve u datoj osnovi.

**Ulaz**

Sa standardnog ulaza se učitava osnova  $b$  ( $2 \leq b \leq 16$ ).

**Izlaz**

Na standardni izlaz ispisati sve trocifrene brojeve u osnovi  $b$ , poređane rastući po vrednosti (brojeve ispisati sa vodećim nulama).

**Primer****Ulaz**

2

**Izlaz**

```

000
001
010
011
100
101
110
111

```

**3.2.0.7. Mali loto**

U jednom odeljenju su odlučili da u sklopu novogodišnje priredbe organizuju malo izvlačenje igre loto. Da bi povećali šanse za dobitak, odlučili su da se izvlače samo tri kuglice. Napiši program koji ispisuje koje sve kombinacije mogu biti izvučene, ako se zna da u bubnju ima  $n$  različitih kuglica obeleženih brojevima od 1 do  $n$ .

**Ulaz**

Sa standardnog ulaza se unosi broj  $n$  ( $4 \leq n \leq 20$ ).

**Izlaz**

Na standardni izlaz ispiši sve kombinacije, pri čemu su brojevi u svakoj kombinaciji sortirani rastuće, a kombinacije su leksikografski sortirane.

**Primer****Ulaz**

5

**Izlaz**

```
1 2 3
1 2 4
1 2 5
1 3 4
1 3 5
1 4 5
2 3 4
2 3 5
2 4 5
3 4 5
```

**3.2.1. Segmenti****3.2.1.1. Svi sufiksi niza brojeva od 1 do  $n$** 

Svojevremeno je postojala reklama za firmu "Jugodravo" u kojoj se pojavljivala pesmica "Jugodravo-ugodravo-godravo-odravo-dravo-rvo-vo-o" u kojoj su se određivali svi sufiksi date reči. Slično, ali malo jednostavnije je ispisati sve sufikse niza prirodnih brojeva od 1 do  $n$ . Napiši program koji za dato  $n$  ispisati sve članove niza  $1, 2, 3, \dots, n, 2, 3, 4, \dots, n, 3, 4, \dots, n, \dots, n-1, n, n$ .

**Ulaz**

Sa standardnog ulaza se učitava prirodan broj  $n$  u granicama od 1 do 30.

**Izlaz**

Brojevi se ispisuju na standardni izlaz, svaki u posebnom redu.

**Primer****Ulaz**

4

**Izlaz**

```
1
2
3
```



4  
2  
3  
4  
3  
4  
4

### 3.2.1.2. Svi prefiksi niza brojeva od 1 do n

Za dato n ispisati sve prefikse niza prirodnih brojeva koji počinju sa 1 i završavaju se sa brojem manjim od n.

#### *Ulaz*

Sa standardnog ulaza se učitava ceo broj n ( $1 \leq n \leq 20$ ).

#### *Izlaz*

Na standardni izlaz ispisati sve tražene segmente svaki u posebnom redu. Iza svakog elementa svakog segmenta treba da bude ispisan po jedan razmak.

#### **Primer**

#### *Ulaz*

3

#### *Izlaz*

1

1 2

1 2 3

### 3.2.1.3. Sve podreči

Napiši program koji će ispisati sve podreči date reči i to u rastućem redosledu početnih pozicija i rastućem redosledu dužine.

#### *Ulaz*

Sa standardnog ulaza se učitava jedna reč sastavljena samo od malih slova engleske abecede.

#### *Izlaz*

Na standardni izlaz ispisati tražene podreči, svaku u posebnom redu.

#### **Primer**

#### *Ulaz*

abcd

#### *Izlaz*

a

ab

abc

```
abcd
b
bc
bcd
c
cd
d
```

#### 3.2.1.4. Sve podreči po opadajućoj dužini

Za unetu reč ispisati sve podreči u redosledu opadajuće dužine i rastućih levih granica za reči iste dužine.

##### **Ulaz**

Sa standardnog ulaza se unosi jedna reč sastavljena samo od malih slova engleske abecede.

##### **Izlaz**

Na standardni izlaz ispisati tražene podreči, svaku u posebnom redu.

#### **Primer**

##### **Ulaz**

```
abcd
```

##### **Izlaz**

```
abcd
abc
bcd
ab
bc
cd
a
b
c
d
```

### 3.2.2. ASCII crteži

#### 3.2.2.1. Trougao od zvezdica

Napiši program koji iscrtava trougao kakav je prikazan u primeru.

##### **Ulaz**

Sa standardnog ulaza se učitava jedan prirodan broj  $n$  ( $1 \leq n \leq 30$ ).

##### **Izlaz**

Na standardni izlaz se ispisuje traženi trougao, ispisivanjem karaktera \*, razmaka (karaktera blanko) i prelaska u novi red. Posle svake poslednje

zvezdice u svakoj vrsti preći u naredni red (ne ispisivati razmake posle zvezdica).

### Primer

**Ulaz**

6

**Izlaz**

```

      *
     ***
    *****
   *********
  ***********
 *****
*****

```

### 3.2.2.2. Kvadrat od zvezdica

Napiši program koji iscrtava kvadrat od karaktera \* (kao što je prikazano u primerima).

**Ulaz**

Sa standardnog ulaza se učitava prirodan broj  $n$  ( $1 \leq n \leq 20$ ).

**Izlaz**

Na standardni izlaz ispisati traženi crtež.

### Primer

**Ulaz**

4

**Izlaz**

```

****
****
****
****

```

### 3.2.2.3. Romb od zvezdica

Napiši program koji ispisuje romb napravljen od zvezdica (kako je prikazano u primeru).

**Ulaz**

Sa standardnog ulaza se unosi prirodan broj  $n$  ( $3 \leq n \leq 20$ ).

**Izlaz**

Na standardni izlaz iscrtati romb.

**Primer****Ulaz**

5

**Izlaz**

```

*****
*****
*****
*****
*****

```

**3.2.2.4. Trougao od reči**

Napiši program koji iscrtava trougao čije su ivice sastavljene od karaktera date reči. Reč se dobija čitanjem slova sa leve i desne ivice trougla naniže, dok se na donjoj ivici nalazi palindrom čija je desna polovina ta data reč.

**Ulaz**

Sa standardnog ulaza se učitava reč dužine između 3 i 20 karaktera.

**Izlaz**

Na standardni izlaz ispisati traženi trougao.

**Primer****Ulaz**

trougao

**Izlaz**

```

  t
 r r
o  o
u   u
g   g
a   a
oaguortrougao

```

**3.2.4. Razno****3.2.3.1. Serije 123...**

Za dato  $n$  ispisati članove niza 1,2,2,3,3,3,4,4,4,4,...,n,...,n (niz sadrži jednu jedinicu, dve dvojke, tri trojke itd.).

**Ulaz**

Sa standardnog ulaza se učitava prirodan broj  $n$  ( $1 \leq n \leq 30$ ).

**Izlaz**

Članove traženog niza ispisati na standardni izlaz, svaki u posebnom redu.

**Primer****Ulaz**

4

**Izlaz**

1

2

2

3

3

3

4

4

4

4

**3.2.3.2. Serije neparni parni**

Za dato  $n$  ispisati elemente niza 1, 2, 4, 5, 7, 9, 10, 12, 14, 16, 17, ... koji se formira tako što se polazeći od broja 1 – prikazuje jedan neparan prirodni broj, zatim sledeća dva parna – 2, 4; pa sledeća 3 neparna – 5, 7, 9; sledeća 4 parna – 10, 12, 14, 16, ... itd. Poslednja serija sadrži  $n$  elemenata.

**Ulaz**

Sa standardnog ulaza se učitava prirodan broj  $n$  u granicama od 1 do 40.

**Izlaz**

Na standardni izlaz ispisati elemente traženog niza, svaki u posebnom redu.

**Primer****Ulaz**

5

**Izlaz**

1

2

4

5

7

9

10

12

14

16

17

19

21

23

25

**3.2.3.3. Tablica množenja**

Napiši program koji ispisuje tablicu množenja.

**Ulaz**

Sa standardnog ulaza se unose dva cela broja  $m$  i  $n$  ( $1 \leq m, n \leq 9$ ), svaki u posebnom redu.

**Izlaz**

Na standardni izlaz ispisati tablicu množenja sa  $m$  vrsta i  $n$  kolona, kako je prikazano u primeru. Između kolona štampati tabulator (karakter Tab).

**Primer****Ulaz**

5  
5

**Izlaz**

```
1 * 1 = 1  1 * 2 = 2  1 * 3 = 3  1 * 4 = 4  1 * 5 = 5
2 * 1 = 2  2 * 2 = 4  2 * 3 = 6  2 * 4 = 8  2 * 5 = 10
3 * 1 = 3  3 * 2 = 6  3 * 3 = 9  3 * 4 = 12  3 * 5 = 15
4 * 1 = 4  4 * 2 = 8  4 * 3 = 12  4 * 4 = 16  4 * 5 = 20
5 * 1 = 5  5 * 2 = 10  5 * 3 = 15  5 * 4 = 20  5 * 5 = 25
```

**3.3. Podserije****3.3.2. Podela na uzastopne segmente****3.3.2.1. Nedelje sa negativnim temperaturama**

Date su maksimalne dnevne temperature svakog dana u  $K$  uzastopnih nedelja. Odrediti koliko ima "ledenih" nedelja u kojima je maksimalna temperatura svakog dana bila negativna.

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se broj nedelja koje analiziramo,  $K$  (prirodan broj  $3 \leq K \leq 100$ ). U svakoj od narednih  $7 \cdot K$  linija nalazi se maksimalna dnevna temperatura odgovarajućeg dana (ceo broj).

**Izlaz**

U jednoj liniji standardnog izlaza nalazi se broj nedelja u kojima je maksimalna temperatura svakog dana bila negativna (prirodan broj, uključujući nulu).

**Primer*****Ulaz***

5  
 2  
 3  
 2  
 -4  
 -3  
 0  
 1  
 -1  
 -2  
 -1  
 -1  
 -2  
 -3  
 -2  
 0  
 0  
 -1  
 2  
 -1  
 0  
 -1  
 -1  
 -2  
 -1  
 -1  
 -1  
 -2  
 -3  
 -2  
 -1  
 -2  
 1  
 1  
 2  
 3  
 -2

***Izlaz***

2

**3.3.2.2. Suma segmenata niza između nula**

Niz celih brojeva može se podeliti u segmente (podnizove sastavljene od uzastopnih elemenata niza) tako da je prvi segment od početka niza do prve nule u nizu, drugi segment je od prve do druge nule, ..., poslednji segment je od poslednje nule u nizu do kraja niza. Na primer, niz 3,1,0,4,2,5,0 se deli na

segment koji sadrži elemente 3,1, segment koji sadrži elemente 4,2,5 i prazan segment na kraju. Napisati program kojim se na osnovu unetog niza celih brojeva dužine N prikazuju sume segmenata redom. Suma praznog segmenta je nula.

#### ***Ulaz***

Prva linija standarnog ulaza sadrži prirodan broj  $N$  ( $2 \leq N \leq 50000$ ) koji predstavlja broj elemenata niza a u svakoj od narednih  $N$  linija nalazi se po jedan član niza (svi celi brojevi iz intervala  $[-100,100]$ ). Pretpostavimo da prvi broj nije nula i da postoji najmanje jedna nula u nizu.

#### ***Izlaz***

Na standarnom izlazu, u svakoj liniji, nalaze se, redom, sume segmenata niza (celi brojevi).

### **Primer 1**

#### ***Ulaz***

11  
5  
4  
0  
2  
33  
-5  
0  
2  
4  
0  
-6

#### ***Izlaz***

9  
30  
6  
-6

### **Primer 2**

#### ***Ulaz***

5  
2  
0  
0  
3  
0

#### ***Izlaz***

2  
0



3  
0

### 3.3.2.3. Parno neparne serije

Niz prirodnih brojeva se može razložiti na segmente (podnizove uzastopnih elemenata) koje čine elementi iste parnosti. Na primer, niz 3,5,2,4,6,7,1 se može razložiti na segmente 3,5, zatim 2,4,6 i na kraju 7,1. Napiši program koji za dati niz prirodnih brojeva izračunava zbrove svih tako dobijenih segmenata.

#### *Ulaz*

Sa standardnog ulaza se učitava broj  $n$  ( $0 \leq n \leq 50000$ ) a zatim  $i$   $n$  prirodnih brojeva, svaki u posebnom redu.

#### *Izlaz*

Na standardni izlaz ispisati tražene zbrove, svaki u posebnom redu.

#### **Primer**

#### *Ulaz*

7  
1  
3  
4  
6  
8  
5  
1

#### *Izlaz*

4  
18  
6

### 3.3.2.4. Najduža serija pobeda

Košarkaški tim je igrao puno utakmica u sezoni. U svakoj utakmici ostvario je ili pobjedu ili poraz. Napiši program koji određuje dužinu najduže serije pobjeda u uzastopnim mečevima tokom sezone.

#### *Ulaz*

Sa standardnog ulaza se unosi prirodan broj  $N$  ( $5 \leq N \leq 50000$ ), a zatim  $i$   $N$  brojeva -1 (što označava poraz) ili 1 (što označava pobjedu).

#### *Izlaz*

Na standardni izlaz ispisati jedan prirodan broj koji predstavlja traženu dužinu najduže serije uzastopnih pobjeda.

**Primer****Ulaz**

8  
1  
1  
-1  
1  
1  
1  
-1  
-1

**Izlaz**

3

**3.3.2.5. Najduža rastuća serija**

Data je dužina priprema za jednog atletičara, u danima. Za svaki dan unosi se broj metara koje je atletičar pretrčao tog dana. Odrediti kog dana počinje i koliko dana traje najduži period u toku priprema u kome je atletičar svakog dana pretrčao više metara od prethodnog (ako je atletičar trčao samo jedan dan, za taj dan se smatra da predstavlja period dužine jedan koji zadovoljava uslov zadatka). Ako ima više takvih perioda odrediti prvi.

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $N$  ( $1 \leq N \leq 50000$ ) koji predstavlja broj dana a u svakoj od narednih  $N$  linija nalazi se po jedan prirodan broj koji predstavlja broj pretrčanih metara odgovarajućeg dana.

**Izlaz**

Na standardnom izlazu, u dve linije, nalaze se, redom, početak i dužina najdužeg perioda u kome je atletičar svakog dana trčao više metara nego prethodnog (prirodni brojevi između 3000 i 6000). Dani se broje od 1 do  $N$ .

**Primer****Ulaz**

11  
5003  
4125  
4320  
4335  
4568  
4633  
4882  
4800  
4850  
4932  
4911

**Izlaz**2  
6**3.3.2.6. Serije kutija**

Na pokretnoj traci nalaze se kutije, za svaku kutiju poznate su njene dimenzije: dužina, širina i visina ( $a$ ,  $b$ ,  $c$ ). Radnik na kraju trake uzima redom kutiju po kutiju. Kutiju koju uzme radnik stavlja u prethodnu kutiju ako je to moguće. Kutija se stavlja u prethodnu tako da su im ivice paralelne, i ivice nove kutije su strogo manje od ivica prethodne kutije. Na taj način radnik slaže seriju kutija, a ako nije moguće novu kutiju staviti u prethodnu radnik počinje da slaže novu seriju kutija. Potrebno je odrediti koliko najviše kutija ima u nekoj seriji.

**Ulaz**

Za svaku kutiju (njih najviše 50000) date su njene dimenzije, tri cela broja (između 1 i 5000), svaka dimenzija u posebnoj liniji. Kraj unosa je kada korisnik unese kutiju čija je bar jedna dimenzija jednaka 0.

**Izlaz**

Broj kutija u najbrojnijoj seriji kutija.

**Primer****Ulaz**5  
3  
7  
6  
1  
4  
9  
5  
6  
3  
7  
2  
1  
5  
2  
4  
0  
6**Izlaz**

3

### 3.3.2.7. Raspored paketa na policama

Na police u magacinu se raspoređuju redom predmeti. Police se popunjavaju redom, i to tako što na svaku policu stavljamo po  $K$  (unapred zadat prirodan broj) predmeta. Napisati program koji za svaku policu ispisuje ukupnu masu predmeta stavljenih na nju a zatim i redni broj police na kojoj je ukupna masa raspoređenih predmeta najveća.

#### **Ulaz**

U prvoj liniji standardnog ulaza nalazi se broj predmeta koje stavljamo na jednu policu  $K$  ( $1 \leq K \leq 1000$ ). U svakoj narednoj liniji (ima ih između 1 i 50000) nalazi se masa predmeta  $m$  ( $1 \leq m \leq 100$ ) koji treba postaviti na policu koja je na redu. Ulaz se završava linijom u kojoj se nalazi broj 0.

#### **Izlaz**

Na standardnom izlazu se prvo, za svaku policu, u po jednoj liniji, nalazi ukupna masa predmeta stavljenih na nju, a zatim u poslednjoj liniji i redni broj police sa najvećom sumom (police se broje od 1).

### **Primer**

#### **Ulaz**

```
3
5
10
8
12
3
5
15
12
7
18
2
6
22
9
0
```

#### **Izlaz**

```
23
20
34
26
31
3
```

### 3.3.2.8. Najstabilniji temperaturni period

Date su maksimalne dnevne temperature za  $N$  uzastopnih dana u godini. Odrediti početak i dužinu temperaturno najstabilnijeg perioda (najduži period u kome temperatura ni u jednom danu nije varirala za više od  $K$  stepeni u odnosu na prethodni dan). Ako je takvih perioda više, odrediti poslednji.

#### *Ulaz*

U prvoj liniji standardnog ulaza nalazi se broj  $N$  (prirodan broj  $5 \leq N \leq 50000$ ), u drugoj broj  $K$  (prirodan broj  $1 \leq K \leq 10$ ), a u svakoj od narednih  $N$  linija nalazi se maksimalna dnevna temperatura određenog dana (prirodan broj između  $-30$  i  $45$ ).

#### *Izlaz*

U prvoj liniji standardnog izlaza ispisuje se redni broj dana kada počinje temperaturno najstabilniji period (prirodan broj) a u drugoj dužina tog perioda (prirodan broj). Dani se broje od 1 do  $N$ .

#### **Primer**

##### *Ulaz*

15  
3  
12  
15  
14  
12  
17  
15  
18  
16  
19  
21  
16  
18  
20  
20  
17

##### *Izlaz*

5  
6

### 3.3.2.9. Utovar transportnog broda

U transportni brod prenose se redom paketi zadatih masa kolicima date nosivosti (ona je veća ili jednaka masi svakog paketa). Pri tome pakete uvek prenosimo u celosti. Kada neki paket ne može stati u kolica zbog trenutnog prekoračenja nosivosti kolica, ta kolica prevozimo do broda, i počinjemo punjenje novih kolica. Napisati program koji prikazuje redom, za svaka kolica,

broj paketa i ukupnu masu paketa prenetih njima. Na kraju prikazati redni broj kolica koja sadrže najveći broj paketa.

#### **Ulaz**

U prvoj liniji standardnog ulaza nalazi se nosivost kolica  $N$  (prirodan broj  $10 \leq N \leq 500$ ). U svakoj narednoj liniji nalazi se masa paketa  $m_i$  (prirodan broj  $1 \leq m_i \leq N$ ) koji treba preneti. Ulaz se završava linijom u kojoj se nalazi broj 0 (njih ima najviše 50000).

#### **Izlaz**

Na standardnom izlazu se prvo, za svaka kolica, u po jednoj liniji, nalazi broj paketa na kolicima i ukupna masa paketa stavljenih na njih (razdvojeni razmakom), a zatim u poslednjoj liniji i redni broj kolica sa najvećim brojem predmeta.

#### **Primer**

##### **Ulaz**

30  
5  
10  
8  
12  
3  
5  
15  
13  
5  
15  
2  
4  
22  
9  
0

##### **Izlaz**

3 23  
3 20  
2 28  
4 26  
1 22  
1 9  
4

#### **3.3.2.10. Najduži segment uzastopnih brojeva**

Naći dužinu najdužeg segmenta celobrojnog niza koji sadrži uzastopne brojeve. Na primer, za niz 1,56,57,58,90,91,92,93,94,45 rezultat je 5.

**Ulaz**

U prvoj liniji standardnog ulaza unosi se broj elemenata niza  $n$  ( $1 \leq n \leq 50000$ ), a zatim u sledećih  $n$  linija celobrojni elementi niza  $-100000 \leq a_i \leq 100000$ .

**Izlaz**

Dužina najdužeg segmenta uzastopnih brojeva.

**Primer****Ulaz**

10  
1  
56  
57  
58  
90  
91  
92  
93  
94  
45

**Izlaz**

5

**3.3.2.11. Planiranje izgradnje prodavnica**

Za  $N$  kuća u jednom selu koje se nalaze kraj glavnog puta data su rastojanja u metrima od ulaza u selo, i to redom, od najbliže do najudaljenije kuće. Za dve kuće kažemo da su bliske ako njihovo međusobno rastojanje nije veće od zadate vrednosti  $R$ . U selu se u nekim kućama grade prodavnice koje treba da opsluže sve kuće. Prva prodavnica se postavlja u najbližu kuću ulazu u selo i ona opslužuje sve kuće iza nje koje su joj bliske, naredna prodavnica se postavlja u prvu sledeću neopsluženu kuću i ona opslužuje sve kuće iza nje koje su joj bliske i tako dalje. Za svaku prodavnicu odrediti broj kuća koje će opsluživati kao i ukupan broj prodavnica.

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se broj kuća duž puta,  $N$  (prirodan broj manji od 50000), a u drugoj rastojanje  $R$  (prirodan broj). U svakoj od narednih  $N$  linija nalazi se rastojanje određene kuće (prirodan broj) od ulaza u selo i to redom, od najbliže do najudaljenije kuće.

**Izlaz**

Na standardnom izlazu se prvo, za svaku prodavnicu, u po jednoj liniji, navodi broj kuća koje opslužuje (prirodan broj), a zatim u poslednjoj liniji i ukupan broj prodavnica.

**Primer*****Ulaz***

10  
50  
5  
19  
31  
39  
52  
66  
100  
118  
140  
166

***Izlaz***

5  
2  
3  
3

**3.4. datumi****3.4.1. Datum u dan godine**

Napisati program koji određuje koji dan u godini predstavlja uneti datum.

***Ulaz***

Sa standardnog ulaza se unose tri pozitivna cela broja (svaki u zasebnom redu) koja predstavljaju dan, mesec i godinu jednog ispravnog datuma.

***Izlaz***

Ispisuje se redni broj dana u godini od 1 do 365, odnosno 366 ako je godina prestupna.

**Primer*****Ulaz***

28  
2  
2016

***Izlaz***

59



### 3.4.2. Dan godine u datum

Napisati program koji na osnovu rednog broj dana u godini i godine određuje datum.

#### *Ulaz*

Sa standardnog ulaza se unose dva pozitivna cela broja (svaki u zasebnom redu) koja predstavljaju redni broj dana u godini (od 1 do 365, odnosno 366 za prestupnu godinu) i godinu.

#### *Izlaz*

Na standardni izlaz ispisati tri cela broja koja predstavljaju dan, mesec i godinu. Svi brojevi se ispisuju u jednom redu, a iza svakog broja navodi se tačka.

#### **Primer**

##### *Ulaz*

60  
2016

##### *Izlaz*

29.2.2016.

### 3.4.3. Broj dana između dva datuma

Napisati program kojim se određuje broj dana između dva datuma.

#### *Ulaz*

U prva tri reda standardnog ulaza unose se tri pozitivna cela broja koji predstavljaju dan, mesec i godinu početka vremenskog perioda, a u sledeća tri takođe tri pozitivna cela broja koji predstavljaju dan, mesec i godinu kraja vremenskog perioda. Oba datuma su korektna.

#### *Izlaz*

Broj dana zadatog vremenskog perioda.

#### **Primer**

##### *Ulaz*

7  
4  
1953  
9  
12  
2016

##### *Izlaz*

23257

### 3.4.4. Dan u nedelji

Ako se zna da je 1. januar 2000. godine bila subota, za dati datum u 21. veku odrediti koji je dan u nedelji.

#### **Ulaz**

Sa standardnog ulaza se unose tri pozitivna cela broja (svaki u zasebnom redu) koja predstavljaju dan, mesec i godinu jednog ispravnog datuma.

#### **Izlaz**

Dan u nedelji: nedelja, ponedeljak, utorak, sreda, cetvrtak, petak, subota.

#### **Primer**

##### **Ulaz**

9  
8  
2015

##### **Izlaz**

nedelja

## 3.5. Teorija brojeva

### 3.5.1. Prost broj

Napiši program koji ispituje da li je uneti prirodan broj prost (veći je od 1 i nema drugih delilaca osim 1 i samog sebe).

#### **Ulaz**

Sa standardnog ulaza se unosi prirodan broj  $n$  ( $1 \leq n \leq 10^9$ ).

#### **Izlaz**

Na standardni izlaz ispisati DA ako je broj  $n$  prost tj. NE ako nije.

#### **Primer**

##### **Ulaz**

17

##### **Izlaz**

DA

#### **Primer 2**

##### **Ulaz**

903543481

##### **Izlaz**

NE

### 3.5.2. Eratostenovo sito

Napiši program koji određuje broj prostih brojeva u intervalu  $[a,b]$  i njihov zbir (ako zbir ima više od 6 cifara, ispisati samo poslednjih 6).

#### **Ulaz**

Sa standardnog ulaza unose se brojevi  $a$  i  $b$  ( $1 \leq a \leq b \leq 10^{**7}$ ), svaki u posebnoj liniji.

#### **Izlaz**

Na standardnom izlazu prikazati u jednoj liniji, odvojeni jednim blanko znakom, broj prostih brojeva is intervala  $[a,b]$  i traženi zbir.

#### **Primer**

##### **Ulaz**

1  
1000

##### **Izlaz**

168 76127

### 3.5.3. Najbliži prost broj

Autor zadatka za takmičenje treba da sastavi test primer za program koji ispituje da li je dati broj prost. Pošto želi da ispita program na primerima razne težine, potrebno je da odredi neki prost broj koji je blizu milijarde, neki koji je blizu miliona i slično. Pomozi mu tako što ćeš napisati program koji određuje najbliži prost broj unetom broju.

#### **Ulaz**

Sa standardnog ulaza unosi se broj  $n$  ( $1 \leq n \leq 10^{**9}$ ).

#### **Izlaz**

Ako je broj  $n$  prost, ispiši poruku: prost broj i vrednost broja  $n$  odvojene razmakom.

Ako postoje dva broja koja su na istom rastojanju od broja  $n$  ispiši poruku: dva broja i proste brojeve, odvojene razmacima. Ispisati prvo manji, pa onda veći prost broj.

Ako je jedinstven najbliži prost broj manji od broja  $n$  ispiši poruku: manji broj i vrednost tog broja, razdvojene razmakom.

Ako je jedinstven najbliži prost broj veći od broja  $n$  ispiši poruku: veci broj i vrednost tog broja, razdvojene razmakom.

#### **Primer 1**

##### **Ulaz**

12

##### **Izlaz**

dva broja 11 13

**Primer 2****Ulaz**

19

**Izlaz**

prost broj 19

**Primer 2****Ulaz**

24

**Izlaz**

manji broj 23

**3.5.4. Rastavljanje na proste činioce**

Ako je dato nekoliko prostih brojeva, njihov proizvod se može veoma lako i brzo odrediti. Međutim, ako je dat proizvod, često je veoma teško odrediti proste brojeve koji ga sačinjavaju. Napiši program koji što efikasnije rešava taj problem.

**Ulaz**Sa standardnog ulaza se unosi jedan prirodan broj  $n$  ( $1 \leq n \leq 2 \cdot 10^9$ ).**Izlaz**Na standardni izlaz ispisati proste činioce broja  $n$ , uređene od najmanjih do najvećih, razdvojene razmakom.**Primer****Ulaz**

900

**Izlaz**

2 2 3 3 5 5

**3.5.5. Euklid**

Mravi, pčele i komarci organizuju sportski turnir i žele da se podele u timove, tako da se svaki tim sastoji samo od jedne vrste insekata, da svi timovi imaju isti broj članova (da bi se nakon runde kvalifikacija unutar svake vrste mogli svoje predstavnike da pošalju na zajednički turnir) i da je svaki insekt uključen tačno u jedan tim. Ako se zna broj insekata svake od tri date vrste, napiši program koji određuje najveću moguću veličinu timova.

**Ulaz**Sa standardnog ulaza se unose tri broja iz intervala  $[1, 2 \cdot 10^9]$ , svaki u posebnom redu: broj mrava, pčela i komaraca.

**Izlaz**

Na standardni izlaz ispisati jedan ceo broj - traženu veličinu tima.

**Primer****Ulaz**

20

30

40

**Izlaz**

10

**3.5.6. Modularni inverz**

Mile i Tanja moraju da razmene tajne poruke koje se sastoje od puno velikih brojeva. Mile je želeo da se dodatno zaštiti i da te brojeve pre slanja izmeni tako što je svaki broj pomnožio tajnim brojem  $a$  i izračunao ostatak pri deljenju sa tajnim brojem  $n$ . Pošto je dobar matematičar, Mile zna da će Tanja lako moći da dešifruje poruku ako zna brojeve  $a$  i  $n$  i ako su oni uzajamno prosti i zato ih je izabrao baš na taj način i unapred ih je dogovorio sa Tanjom. Napiši program koji Tanji pomaže da dešifruje brojeve koje joj je Mile poslao.

**Ulaz**

Sa standardnog ulaza se unose brojevi  $a$  i  $n$  ( $2 \leq a, n \leq 10^{**9}$ ), za koje se zna da su uzajamno prosti. Nakon toga unose se brojevi  $x_i$  ( $0 \leq x_i < n$ ), njih najviše 10, svaki u posebnom redu sve do kraja standardnog ulaza, koji predstavljaju brojeve koje je Mile dobio nakon šifrovanja originalnih brojeva.

**Izlaz**

Na standardni izlaz ispisati originalne (dešifrovane) brojeve, svaki u posebnom redu.

**Primer****Ulaz**

3

5

1

2

3

4

**Izlaz**

2

4

1

3

Zaista, važi da je  $(3 \cdot 2) \bmod 5 = 6 \bmod 5 = 1$ ,  $(3 \cdot 4) \bmod 5 = 12 \bmod 5 = 2$ ,  $(3 \cdot 1) \bmod 5 = 3 \bmod 5 = 3$  i  $(3 \cdot 3) \bmod 5 = 9 \bmod 5 = 4$ .

### 3.5.7. Dopuna do punog kvadrata

Napiši program koji za uneti prirodni broj  $n$  određuje najmanji broj  $m$  takav da je  $n \cdot m$  potpun kvadrat.

#### **Ulaz**

Sa standardnog ulaza se unosi prirodni broj  $n$  ( $1 \leq n \leq 2 \cdot 10^9$ ).

#### **Izlaz**

Na standardni izlaz ispisati traženi broj  $m$ .

#### **Primer**

##### **Ulaz**

104

##### **Izlaz**

26

### 3.5.8. Broj deljiv sa 1 do $n$

Najmanji broj koji je deljiv svim brojevima od 1 do 10 je broj 2520. Napiši program koji za dati prirodan broj  $n$  određuje koji je najmanji broj deljiv brojevima od 1 do  $n$ .

#### **Ulaz**

Sa standardnog ulaza učitava se broj  $n$  ( $1 \leq n < 23$ ).

#### **Izlaz**

Na standardni izlaz se ispisuje traženi broj.

#### **Primer**

##### **Ulaz**

10

##### **Izlaz**

2520

### 3.5.9. Kineska teorema

Pera pokušava da paralelizuje svoj program koji radi nad  $x$  podataka tako da svaki procesor obrađuje isti broj podataka. Ako rasporedi podatke na  $n_1$  procesora, ostaje mu  $a_1$  podataka viška, ako rasporedi podatke na  $n_2$  procesora, ostaje mu  $a_2$  podataka viška, a ako ih rasporedi na  $n_3$  procesora, ostaje mu  $a_3$  podataka viška. Ako se znaju brojevi  $a_1$ ,  $n_1$ ,  $a_2$ ,  $n_2$ ,  $a_3$  i  $n_3$  i ako se zna da su brojevi  $n_i$  uzajamno prosti, napiši program koji određuje  $x$ .

#### **Ulaz**

Sa standardnog ulaza unose se brojevi  $a_1$ ,  $n_1$ ,  $a_2$ ,  $n_2$ ,  $a_3$  i  $n_3$  ( $2 \leq n_i \leq 5 \cdot 10^4$ ,  $0 \leq a_i < n_i$ ). Svaki par se navodi u posebnom redu, a brojevi su razdvojeni razmakom. Brojevi  $n_1$ ,  $n_2$  i  $n_3$  su uzajamno prosti.

**Izlaz**

Na standardni izlaz ispisati jedan prirodan broj - jedinstven prirodan broj manji od proizvoda  $n_1 \cdot n_2 \cdot n_3$  koji zadovoljava date uslove.

**Primer****Ulaz**

2 3  
3 5  
2 7

**Izlaz**

23

Kada se 23 podatka podeli na 3 procesora, svaki procesor dobija 7 podatka (ukupno 21) i 2 podatka ostaju neraspoređena. Kada se podeli na 5 procesora svaki procesor dobija po 4 podatka (ukupno 20) i tri podatka ostaju neraspoređena. Kada se podeli na 7 procesora, svaki procesor dobija po 3 podatka (ukupno 21) i opet 2 podatka ostaju neraspoređena. Slično bi važno i za 128 podataka, 233 podatka itd., ali 23 je jedini broj manji od  $3 \cdot 5 \cdot 7 = 105$  za koji ovo važi.

**3.5.10. Prijateljski brojevi**

Brojevi su prijateljski, ako je zbir delilaca prvog broja jednak drugom, a zbir delilaca drugog broja jednak prvom broju (u zbir delilaca broja se ubraja broj 1, ali ne i sam taj broj). Napiši program koji ispisuje sve parove prijateljskih brojeva takve da oba broja leže u datom intervalu.

**Ulaz**

Sa standardnog ulaza se učitavaju brojevi  $a$  i  $b$  ( $1 \leq a \leq b \leq 500000$ ).

**Izlaz**

Na standardni izlaz ispisati sve tražene parove uređene rastući po prvom elementu, tako je u svakom paru prvi broj manji ili jednak od drugoga.

**Primer 1****Ulaz**

1  
1000

**Izlaz**

6 6  
28 28  
220 284  
496 496

**Primer 2****Ulaz**

300000

400000

**Izlaz**

308620 389924

356408 399592

**3.5.11. Učenici na istim sedištima**

Učenici gledaju dva filma u bioskopskoj sali u kojoj su sedišta raspoređena u  $m$  vrsta i  $n$  kolona (učenika ima tačno  $m \cdot n$ ). Kada su gledali prvi film, nastavnici su ih rasporedili tako što su ih ređali po azbučnom redosledu popunjavajući vrstu po vrstu, a kada su gledali drugi film, ponovo su bili raspoređeni po azbučnom redosledu, ali ovaj put vrstu po vrstu. Napiši program koji određuje koliko učenika je sedelo na istom mestu tokom gledanja oba filma.

**Ulaz**

Sa standardnog ulaza se unose dva cela broja  $m$  i  $n$  ( $1 \leq m, n \leq 10000$ ), svaki u posebnoj liniji.

**Izlaz**

Na standardni izlaz ispisati jedan ceo broj koji predstavlja broj učenika koji su tokom gledanja oba filma sedeli na istom sedištu.

**Primer****Ulaz**

3

5

**Izlaz**

3

Objašnjenje: Prilikom prvog filma učenici su ređani na sledeći način:

0 1 2

3 4 5

6 7 8

9 10 11

12 13 14

A prilikom gledanja drugog filma, učenici su ređani na sledeći način:

0 5 10

1 6 11

2 7 12

3 8 13

4 9 14

Na istom mestu sedeli su učenici 0, 7 i 14.



## 4. Nizovi

### 4.0. Nizovi vektori liste

#### 4.0.1. Elementarno koriscenje nizova

##### 4.0.1.1. Ispis u obratnom redosledu

Napisati program kojim se unosi  $n$  celih brojeva, a zatim se uneti brojevi prikazuju u obratnom redosledu od redosleda unošenja.

##### *Ulaz*

U prvoj liniji standardnog ulaza nalazi se prirodan broj  $n$  ( $1 \leq n \leq 1000$ ). U sledećih  $n$  linija nalaze se celi brojevi između -1000 i 1000.

##### *Izlaz*

Na standardnom izlazu prikazati unete brojeve, svaki u posebnoj liniji, u obratnom redosledu od redosleda unošenja.

##### **Primer**

##### *Ulaz*

5  
10  
-123  
67  
14  
987

##### *Izlaz*

987  
14  
67  
-123  
10

##### 4.0.1.2. Prosečno odstupanje od minimalnog

Date su cene uređaja u  $n$  prodavnica. Napisati program kojim se određuje koliko su u proseku cene uređaja skuplje od najmanje cene uređaja u tim prodavnicama.

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se prirodan broj  $n$  ( $1 \leq n \leq 200$ ). U sledećih  $n$  linija nalaze se pozitivni realni brojevi koji predstavljaju cene uređaja u prodavnicama.

**Izlaz**

Na standardnom izlazu prikazati na dve decimale zaokruženo prosečno odstupanje cena od minimalne cene.

**Primer****Ulaz**

4  
100  
95  
120  
95

**Izlaz**

7.50

**4.0.1.3. Minimalno odstupanje od proseka**

Za dati niz od  $n$  realnih brojeva odrediti vrednost najmanjeg apsolutnog odstupanja od proseka vrednosti unetih brojeva (to je najmanja apsolutna vrednost razlika između elemenata i proseka niza).

**Ulaz**

U prvoj liniji standardnog ulaza unosi se broj elemenata niza  $n$  ( $1 \leq n \leq 100$ ), a zatim u sledećih  $n$  linija elementi niza  $-10000 \leq a_i \leq 10000$ .

**Izlaz**

Realan broj na dve decimale koji predstavlja vrednost najmanjeg apsolutnog odstupanja od srednje vrednosti.

**Primer****Ulaz**

6  
2.8  
19.3  
-4.2  
7.5  
-11.1  
7.17

**Izlaz**

0.78

#### 4.0.1.4. Translacija tačaka

Dane su koordinate  $N$  tačaka u ravni. Translirati tačke tako da im težište bude u koordinatnom početku.

##### **Ulaz**

U prvoj liniji standardnog ulaza nalazi se prirodan broj  $n$  ( $1 \leq n \leq 100$ ). U sledećih  $n$  linija nalaze se po dva realna broja, koji predstavljaju  $x$  i  $y$  koordinate tačaka.

##### **Izlaz**

Na standardnom izlazu prikazati koordinate tačaka posle translacije, za svaku tačku u jednoj liniji njenu  $x$  pa  $y$  koordinatu, koordinate odvojiti jednom prazninom i prikazati ih na dve decimale.

#### **Primer**

##### **Ulaz**

```
3
0 0
1 0
2 3
```

##### **Izlaz**

```
-1.00 -1.00
0.00 -1.00
1.00 2.00
```

#### 4.0.1.5. Parni i neparni elementi

Poštara ide ulicom i treba da rasporedi pošiljke. Odlučio je da prvo obiđe jednu stranu ulice (na kojoj su kuće sa parnim brojevima), a da u povratku obiđe drugu stranu ulice (na kojoj su kuće sa neparnim brojevima). Napiši program koji za unete brojeve kuća određuje koje su na parnoj, a koje na neparnoj strani ulice.

##### **Ulaz**

Sa standardnog ulaza se unosi broj pošiljki (prirodan broj  $k$  manji od 100), a zatim i  $k$  prirodnih brojeva koji predstavljaju brojeve kuća na pošiljkama. Svaki broj je zadat u posebnom redu.

##### **Izlaz**

Na standardni izlaz ispisati dva reda brojeva. U prvom redu se nalaze svi parni brojevi (ispisani u istom redosledu u kojem su uneti), a u drugom svi neparni brojevi (ispisani u istom redosledu u kojem su uneti). Brojevi u svakom redu treba da su razdvojeni sa po jednim razmakom, a razmak može biti naveden i iza poslednjeg broja.

**Primer****Ulaz**

5  
1  
2  
3  
4  
5

**Izlaz**

2 4  
1 3 5

**4.0.2. Transformacije nizova****4.0.2.1. Izbacivanje elemenata**

Broj je nepoželjan u nizu celih brojeva ako deli ukupan broj elemenata (npr. u nizu dužine 10 su nepoželjni elementi koji dele broj 10, a to su 1, 2, 5 i 10). Potrebno je pronaći sve nepoželjne elemente u nizu i ukloniti ih. Nakon toga se broj elemenata može promeniti i neki drugi elementi mogu postati nepoželjni. Postupak se ponavlja dok se ne dobije niz bez nepoželjnih elemenata. Napiši program koji za dati niz određuje zbir preostalih elemenata, nakon uklanjanja nepoželjnih.

**Ulaz**

Sa standardnog ulaza se unosi broj  $n$  ( $1 \leq n \leq 50000$ ), a zatim i  $n$  elemenata niza iz raspona od 1 do 100.

**Izlaz**

Na standardni izlaz ispisati jedan ceo broj koji predstavlja zbir preostalih elemenata u nizu, nakon uzastopnog uklanjanja svih nepoželjnih elemenata.

**Primer****Ulaz**

10  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10

**Izlaz**

24

Prvo se uklanjaju elementi 1, 2, 5 i 10 koji dele dužinu 10, zatim se uklanjaju elementi 3 i 6 koji dele dužinu 6 i na kraju se uklanja element 4, tako da ostaju elementi 7, 8 i 9 čiji je zbir 24.

**4.0.2.2. Ciklično pomeranje za jedno mesto**

Napisati program koji učitava niz celih brojeva a zatim ga transformiše tako što se ciklično pomeraju zadati delovi niza od pozicije p do pozicije q sve dok se ne unesu dve jednake pozicije. Pri tome vršiti ciklično pomeranje udesno ako je  $p < q$ , a pomernje ulevo vršiti ako je  $p > q$ .

**Ulaz**

U jednoj liniji standardnog ulaza nalazi se broj elemenata niza  $n$  ( $1 < n \leq 200$ ), a zatim se, u svakoj od  $n$  narednih linija standardnog ulaza, nalazi po jedan član niza. U narednim redovima se unose po dva cela broja,  $p$  i  $q$  ( $0 \leq p, q < n$ ), odvojena prazninom dok se ne unese red u kome su brojevi jednaki.

**Izlaz**

U svakoj liniji standardnog izlaza ispisuje se po jedan element transformisanog niza.

**Primer****Ulaz**

```
4
1
2
3
4
2 3
2 0
1 2
0 0
```

**Izlaz**

```
2
1
4
3
```

**4.0.2.3. Različiti elementi niza**

Napisati program koji učitani niz celih brojeva transformiše tako da se svaki element niza pojavljuje tačno jednom, pri čemu se čuva redosled elemenata (redosled njihovog prvog pojavljivanja).

**Ulaz**

U jednoj liniji standardnog ulaza nalazi se broj elemenata niza  $N$  ( $0 < N \leq 10000$ ), a zatim se, u svakoj od  $N$  narednih linija standardnog ulaza, nalazi po jedan član niza.

**Izlaz**

U svakoj liniji standardnog izlaza ispisuje se po jedan element transformisanog niza.

**Primer****Ulaz**

10  
1  
3  
5  
3  
1  
5  
7  
2  
3  
5

**Izlaz**

1  
3  
5  
7  
2

**4.0.2.4. Neobrisani brojevi**

Posmatrajmo skup prirodnih brojeva 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, ... Prvo izbrišemo svaki drugi broj i posmatrani skup postaje 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, ... zatim iz dobijenog skupa brišemo svaki treći broj i dobijamo skup 1, 3, 7, 9, 13, 15, 19, 21, ... zatim iz dobijenog skupa brišemo svaki četvrti broj i tako dalje. Skup neobrisanih brojeva je 1, 3, 7, 13, 19, ... Napisati program kojim se za dati prirodan broj  $n$  proverava da li pripada skupu neobrisanih brojeva.

**Ulaz**

Prva linija standardnog ulaza sadrži prirodan broj  $n$  ( $n < 5 \cdot 10^{**5}$ ).

**Izlaz**

Reč da ako broj  $n$  pripada skupu neobrisanih brojeva, u suprotnom reč ne.

**Primer 1****Ulaz**

39

**Izlaz**

da

**Primer 2****Ulaz**

41

**Izlaz**

ne

**4.0.3. Obilazak sa dva kraja****4.0.3.1. Niska palindrom**

Napisati program kojim se proverava da li je data reč sastavljena isključivo od malih slova palindrom. Reč je palindrom ako se jednako čita slevo na desno i sdesna na levo.

**Ulaz**

Prva i jedina linija standardnog ulaza sadrži reč.

**Izlaz**

Na standardnom izlazu prikazati reč da ako reč predstavlja palindrom inače prikazati reč ne.

**Primer****Ulaz**

madam

**Izlaz**

da

**4.0.3.2. Palindromska rečenica**

Napisati program kojim se proverava da li je data rečenica palindrom. Rečenica je palindrom ako se jednako čita slevo na desno i sdesna na levo, pri čemu se razmatraju samo slova i ne pravi se razlika između velikih i malih slova.

**Ulaz**

Prva i jedina linija standardnog ulaza sadrži rečenicu (sastavljenu od slova, praznina i interpukcijskih znakova).

**Izlaz**

Na standardnom izlazu prikazati reč da ako rečenica predstavlja palindrom  
inače prikazati reč ne.

**Primer****Ulaz**

Ana voli Milovana!!!

**Izlaz**

da

**4.0.5. sortiranje****4.0.5.0. sortiranje****4.0.5.0.1. Sortiranje brojeva**

Napiši program koji uređuje (sortira) niz brojeva neopadajuće (svaki naredni mora da bude veći ili jednak od prethodnog).

**Ulaz**

Sa standardnog ulaza se unosi broj  $n$  ( $1 \leq n \leq 5 \cdot 10^4$ ) a zatim  $n$  prirodnih brojeva manjih od  $2n$ , svaki u posebnom redu.

**Izlaz**

Na standardni izlaz ispisati učitane brojeve u sortiranom redosledu.

**Primer****Ulaz**

5  
3  
1  
6  
8  
1

**Izlaz**

1  
1  
3  
6  
8

**4.0.5.0.2. Najvredniji predmeti**

Za svaki predmet koji je na prodaju data je šifra i cena. Kupac ima na raspolaganju određeni iznos dinara i želi da kupi što skuplje predmete. Redom



uzima predmete počev od najskupljeg, dok ima novca. Ako nema novca za najskuplji, uzima najskuplji za koji ima novca. Prikazati šifre predmeta koje kupac kupuje i, ako mu je ostalo, preostali iznos novca. Napomena: ova strategija ne garantuje da će predmeti koje kupi biti ukupno najveće moguće vrednosti (npr. ako ima 5 dinara i ako su cene predmeta 4, 3 i 2 dinara, on će kupiti predmet samo predmet od 4 dinara, a mogao bi da kupi predmete od 3 i 2 dinara).

### ***Ulaz***

U prvoj liniji standardnog ulaza nalazi se iznos novca (realan broj) koji ima kupac, u drugoj broj predmeta, N, a zatim se, u svake dve linije standardnog ulaza, unose, redom, šifra (niska karaktera) pa cena (realan broj) predmeta, svaka u posebnom redu, za svih N predmeta.

### ***Izlaz***

U svakoj liniji standardnog izlaza ispisuju se šifre i cene kupljenih predmeta (razdvojene razmakom), ako ih ima. U poslednjoj liniji prikazuje se preostali iznos novca, ako postoji.

## **Primer1**

### ***Ulaz***

```
1250.75
5
predmet1
1010.30
predmet2
357.35
predmet3
725.45
predmet4
1125.5
predmet5
115.75
```

### ***Izlaz***

```
predmet4 1125.5
predmet5 115.75
9.50
```

## **Primer2**

### ***Ulaz***

```
10000
6
predmet1
3010
predmet2
3005
```

predmet3  
5725  
predmet4  
1265  
predmet5  
2075  
predmet6  
385

**Izlaz**

predmet3 5725.00  
predmet1 3010.00  
predmet4 1265.00

**Primer3****Ulaz**

1000  
6  
predmet1  
3010  
predmet2  
3005  
predmet3  
5725  
predmet4  
1265  
predmet5  
2075  
predmet6  
3850

**Izlaz**

1000.00

**4.0.5.1. objedinjavanje****4.0.5.0.1. Objedinjavanje**

U školi malih žutih mrava nastavnik je pregledao kontrolni zadatak. Prvo je pregledao đake koji su radili grupu A, a zatim one koji su radili grupu B, sredio je rezultate za svaku grupu i mrave poređao na osnovu broja poena koji su osvojili. Napiši program koji mu pomaže da od uređenog spiska učenika koji su radili zadatke iz grupe A i od uređenog spiska učenika koji su radili zadatke iz grupe B dobije jedinstven uređen spisak svih učenika.

**Ulaz**

Sa standardnog ulaza se unosi broj đaka  $m$  koji su radili grupu A ( $5 \leq m \leq 25000$ ), a zatim neopadajuće sortiran niz poena tih đaka, svaki u

posebnoj liniji. Nakon toga se unosi broj  $n$  đaka koji su radili grupu B ( $5 \leq n \leq 25000$ ), a zatim neopadajuće sortiran niz poena tih đaka, svaki u posebnoj liniji.

#### **Izlaz**

Na standardni izlaz ispisati neopadajuće sortirani niz poena svih đaka zajedno, svaki u posebnoj liniji.

#### **Primer**

##### **Ulaz**

4  
1  
3  
5  
7  
3  
2  
4  
5

##### **Izlaz**

1  
2  
3  
4  
5  
5  
7

#### **4.0.5.0.2. Zajednički elementi tri uređena niza**

Tri lokalne prodavnice prodaju različite proizvode. Svaki proizvod je okarakterisan jedinstvenim bar-kodom (prirodnim brojem manjim od milijardu). Ako su poznati spiskovi proizvoda koji se prodaju u te tri prodavnice (sortirani rastuće po bar-kodovima), napiši program koji određuje proizvode koji se prodaju u sve tri prodavnice.

##### **Ulaz**

U prvoj linija standardnog ulaza nalazi se prirodan broj  $n$  ( $n \leq 50000$ ) koji predstavlja broj proizvoda u prvoj prodavnici, u sledećih  $n$  linija nalaze se bar-kodovi proizvoda iz te prodavnice (uređeni rastuće). Na isti način su zadati i proizvodi iz druge dve prodavnice.

##### **Izlaz**

Na standardnom izlazu prikazati bar-kodove proizvoda koji se prodaju u svakoj od te tri prodavnice, uređene rastući.

**Primer*****Ulaz***

5  
1  
5  
6  
7  
10  
4  
5  
9  
10  
17  
4  
5  
6  
10  
13

***Izlaz***

5  
10

**4.0.5.2. binarna\_pretraga****4.0.5.0.1. Kružne zone**

Kvalitet signala zavisi od udaljenosti tačke od predajnika. Prostor je podeljen u zone oblika kružnih prstenova, pri čemu su širine prstenova međusobno različite. Napiši program koji za datu tačku određuje zonu kojoj pripada.

***Ulaz***

Sa standardnog ulaza unosi se broj  $n$  ( $1 \leq n \leq 50000$ ), a zatim  $n$  realnih brojeva zaokruženih na dve decimale, svaki u posebnom redu, koji predstavljaju širine svih kružnih prstenova. Nakon toga se unosi broj  $m$  ( $1 \leq m \leq 50000$ ) i zatim  $m$  parova koordinata tačaka (u svakom redu se nalaze dva realana broja zaokružena na dve decimale, razdvojena sa po jednim razmakom).

***Izlaz***

Na standardni izlaz ispisati  $m$  linija. U svakoj liniji ispisati ili indeks zone (broje se od nule) kojoj tačka pripada ili tekst izvan ako je tačka izvan poslednje zone. Ako je tačka na granici dve zone, smatrati da pripada unutrašnjoj.

**Primer*****Ulaz***

3  
2.0

3.0  
7.0  
5  
1.0 1.0  
2.0 3.0  
8.0 7.0  
13.2 14.5  
0.0 12.0

**Izlaz**

0  
1  
2  
izvan  
2

**4.0.5.0.2. Binarna pretraga**

U prodavnici se nalazi puno vrsta proizvoda i poznati su njihovi bar-kodovi. Proizvođač želi da sazna koliko se vrsta njegovih proizvoda prodaje u toj prodavnici. Ako je spisak svih kodova proizvoda u prodavnici dat u sortiranom obliku, a spisak svih kodova proizvoda proizvođača je dostavljen nesortiran, napiši program koji određuje traženi broj.

**Ulaz**

Sa standardnog ulaza učitava se broj  $n$  ( $1 \leq n \leq 50000$ ), a zatim u narednih  $n$  linija po jedan prirodan broj (najviše šestocifren). Ti brojevi predstavljaju bar-kodove proizvoda u prodavnici i sortirani su rastuće. Nakon toga se do kraja ulaza učitavaju bar-kodovi proizvoda koje je proizvođač dostavio (najviše šestocifreni prirodni brojevi, svaki u posebnom redu).

**Izlaz**

Na standardni izlaz ispisati broj proizvoda proizvođača koji se već prodaju u prodavnici.

**Primer****Ulaz**

5  
1  
3  
5  
6  
7  
2  
3  
4  
5  
8

**Izlaz**

2

**4.0.7. vektori\_i\_polinomi****4.0.7.1. Vrednost polinoma**

Sa standardnog ulaza se unose stepen  $n$  i realni koeficijenti polinoma  $y=a[n] \cdot x^{**n}+a[n-1] \cdot x^{**}(n-1)+\dots+x \cdot a[1]+a[0]$ . Napiši program koji izračunava vrednost tog polinoma u  $k$  ravnomerno raspoređenih tačaka intervala  $[p,q]$ .

**Ulaz**

U prvoj liniji standardnog ulaza uneti  $n$  ( $2 \leq n \leq 9$ ) - stepen polinoma, u sledećih  $n+1$  linija realne vrednosti koeficijenata polinoma, zatim, u narednoj liniji  $k$  ( $2 \leq k \leq 40$ ) - broj ravnomerno raspoređenih tačaka na intervalu  $[p,q]$ , u narednoj liniji realnu vrednost  $p$  - početak intervala, i u narednoj liniji realna vrednost  $q$  - kraj intervala.

**Izlaz**

U  $k$  linija ispisati vrednost polinoma u ravnomerno raspoređenim tačakama intervala  $[p,q]$  zaokruženu na dve decimale.

**Primer****Ulaz**

2

1.0

2.0

1.0

10

1.0

10.0

**Izlaz**

4.00

9.00

16.00

25.00

36.00

49.00

64.00

81.00

100.00

121.00

#### 4.0.7.2. Aritmetika nad polinomima

Data su dva polinoma,  $P$  i  $Q$  stepenima i nizovima svojih koeficijenata. Odrediti njihov zbir i proizvod.

##### **Ulaz**

U prvoj liniji standardnog ulaza nalazi se stepen  $n$  ( $0 \leq n \leq 20$ ) prvog polinoma, a u sledećih  $n+1$  linija realni koeficijenti prvog polinoma i to redom počev od koeficijenta uz najveći stepen. Zatim se na standardnom ulazu nalazi stepen  $m$  ( $0 \leq m \leq 20$ ) drugog polinoma, a u sledećih  $m+1$  linija realni koeficijenti drugog polinoma i to redom počev od koeficijenta uz najveći stepen.

##### **Izlaz**

Prikazati redom zbir i proizvod, svaki u posebnoj liniji. Za svaki polinom prikazati u jednoj liniji njegove koeficijente, na dve decimale i to redom počev od koeficijenta uz najveći stepen.

#### **Primer**

##### **Ulaz**

2  
2  
1  
2  
1  
1  
-1

##### **Izlaz**

2.00 2.00 1.00  
2.00 -1.00 1.00 -2.00

## 4.2. Niske

### 4.2.0. Podniske

#### 4.2.0.1. Reč Frankenštajn

Napiši program koji od unete reči pravi novu reč sastavljenu iz delova početne reči. Na primer, za uneto:

dabrakabmra

6 2  
3 3  
1 1  
0 3  
9 2

program gradi i ispisuje reč abrakadabra (prvo idu 2 karaktera krenuvši od pozicije 6 tj. ab, zatim 3 karaktera od pozicije 3 tj. rak, zatim 1 karakter od pozicije 1 tj. a, zatim tri karaktera od pozicije 0 tj. dab i na kraju dva karaktera od pozicije 9 tj. ra).

#### **Ulaz**

U prvoj liniji standardnog ulaza je string koji predstavlja datu reč (dužina mu je najviše 50000 karaktera) i sadrži samo slova engleske abecede. U narednim linijama (njih najviše 10000) se unose parovi: pozicija u datom stringu i dužina podstringa (pozicija i dužina ispravno zadaju podnisku).

#### **Izlaz**

Reč koja se dobija navedenim postupkom.

### **Primer**

#### **Ulaz**

maranaakkabbikopa

13 3

16 1

8 3

3 3

#### **Izlaz**

kopakabana

### **4.2.0.2. Izbacivanje podniski**

Napisati program kojim se određuje tekst koji se dobija brisanjem iz datog teksta svih pojavljivanja podreči iz datog skupa. Brišu se prvo sva pojavljivanja prve reči, zatim druge, treće i tako do kraja. Taj postupak se ponavlja sve dok je se tekst njime menja. Prilikom brisanja svih pojavljivanja reči postupak se iscrpno ponavlja sve dok je brisanje moguće.

Na primer, za tekst babrarkadabrabrr i skup reči {br, ka, aa}, prvo se iscrpno briše br i dobija se baarkadaa, zatim se briše ka i dobija se baardaa, zatim se briše aa i dobija se brd. Nakon toga se kreće iz početka, iscrpno se briše br i dobija d, pokušava se sa brisanjem ka i aa koje ne uspeva, prolazi se kroz treći krug u kojem reč ostaje ista i prijavljuje se rezultat d.

#### **Ulaz**

U prvoj liniji standardnog ulaza je tekst, dužine najviše  $5 \cdot 10^{*5}$  karaktera iz kojeg se brišu reči koje se učitavaju iz narednih linija. Reči se učitavaju do kraja standardnog ulaza i ima ih najviše  $10^{*5}$ , a svaka je dužine najviše 10 karaktera.

#### **Izlaz**

Tekst koji se dobija nakon brisanja podreči koje sadrži.



**Primer****Ulaz**

babrarkadabrabbrr  
br  
ka  
aa

**Izlaz**

d

**4.2.0.3. Prezime pa ime**

Učenici koji su zainteresovani na takmičenju su napravili spisak u elektronskom obliku, ali su na tom spisku uvek pisali prvo ime, pa onda prezime. Međutim, elektronski sistem prijavljivanja zahteva da se unese prvo prezime, pa onda ime. Napiši program koji popravljja spisak u skladu sa tim zahtevom.

**Ulaz**

Svaka linija standardnog ulaza, sve do njegovog kraja, sadrži ime i prezime učenika (koja su otkucana samo korišćenjem slova engleske abecede), razdvojene sa tačno jednim razmakom. Napomena: pri interaktivnom testiranju kraj ulaza se može uneti tasterima ctrl+z tj. ctrl+d.

**Izlaz**

Na standardni izlaz ispisati prezimena i imena svih učitanih učenika razdvojene sa po jednim razmakom, u istom redosledu u kojem su učitani.

**Primer****Ulaz**

Petar Petrovic  
Ana Anic  
Jelena Jelenkovic  
Mika Mikic

**Izlaz**

Petrovic Petar  
Anic Ana  
Jelenkovic Jelena  
Mikic Mika

**4.2.0.4. Podniska**

Napisati program kojim se za dve date reči proverava da li je druga sadržana u prvoj, ako jeste odrediti prvu poziciju na kojoj se druga reč pojavljuje u prvoj.

**Ulaz**

Na standardnom ulazu nalaze se dve reči svaka u posebnoj liniji. Svaka reč ima najviše 20 karaktera.

**Izlaz**

Na standardnom izlazu prikazati prvu poziciju (pozicije su numerisane od 0) na kojoj se nalazi druga reč u prvoj, ako se druga reč ne nalazi u prvoj prikazati -1.

**Primer 1****Ulaz**

banana  
ana

**Izlaz**

1

**Primer 2****Ulaz**

branka  
ana

**Izlaz**

-1

**4.2.1. poredjenje****4.2.1.1. Leksikografski minimum**

Napiši program koji određuje koja bi od nekoliko datih reči trebalo da bude navedena prva u rečniku ako se ne pravi razlika između velikih i malih slova (za tu reč kažemo da je leksikografski ispred ostalih reči, tj. da je leksikografski najmanja među svim rečima).

**Ulaz**

Sa standardnog ulaza se učitava linija teksta koja sadrži više reči razdvojenih sa po tačno jednim razmakom (u liniji postoji bar jedna reč, a iza poslednje reči nema razmaka).

**Izlaz**

Na standardni izlaz ispisati reč iz učitane linije koja je leksikografski ispred ostalih reči.

**Primer****Ulaz**

Ko rano rani dve sreće grabi

**Izlaz**

dve

## 4.2.2. transformisanje

### 4.2.2.1. Cezarov kod

Zadata je reč sastavljena od malih slova rimske abecede (koja je ista kao engleska). Potrebno je šifrovati odnosno dešifrovati poruku. Cezarova šifra je tip šifre zamenjivanja u kome se svako slovo teksta koji se šifruje menja slovom abecede koje se dobije pomeranjem polaznog slova za određeni broj mesta, ciklično po abecedi. Na primer, ako se vrši pomeranje za dva mesta, slovo a bi se šifrovalo slovom c, slovo b slovom d, itd., dok bi se slovo z šifrovalo slovom b, slovo x slovom a, slovo y slovom b i slovo z slovom c. Na primer, reč papagaj bi se šifrovala sa sdsdjdmd.

#### **Ulaz**

U prvoj liniji standardnog ulaza nalazi se reč ne duža od 100 slova, u drugoj liniji se nalazi ceo broj  $N$ , koji predstavlja pomak ( $1 \leq N < 26$ ), a u trećoj ceo broj  $S$  koji predstavlja smer šifrovanja. Ako je  $S=1$  potrebno je šifrovati, a ako je  $S=2$  potrebno je dešifrovati reč.

#### **Izlaz**

U prvoj liniji standardnog izlaza prikazati reč koja se dobije nakon šifrovanja tj. dešifrovanja zadate reči.

### **Primer 1**

#### **Ulaz**

abcdefghijklmnopqrstuvwxyz

3

1

#### **Izlaz**

defghijklmnopqrstuvwxyzabc

### **Primer 2**

#### **Ulaz**

twornuvkpemkp

2

2

#### **Izlaz**

rumplstinckin

## 4.2.3. Zapis brojeva i izraza

### 4.2.3.1. Arapski u rimski

Napiši program koji prevodi uneti arapski broj u rimski.

**Ulaz**

Sa ulaza se unosi prirodan broj  $n$  ( $1 \leq n \leq 2000$ )

**Izlaz**

Na standardni izlaz se ispisuje rimski zapis broja  $n$ .

**Primer****Ulaz**

1978

**Izlaz**

MCMLXXVIII

**4.2.3.2. Rimski u arapski**

Napiši program koji konvertuje rimske u arapske brojeve.

**Ulaz**

Jedina linija standardnog ulaza sadrži jedan rimski broj iz intervala od 1 do 2000.

**Izlaz**

Na standardni izlaz ispisati uneti broj zapisan u uobičajenom (arapskom) zapisu.

**Primer****Ulaz**

MCMLXXVIII

**Izlaz**

1978

**4.2.3.3. Vrednost izraza**

Napiši program koji izračunava vrednost jednostavnog aritmetičkog izraza (npr.  $123 * 456$ ).

**Ulaz**

Sa standardnog ulaza učitava se jedna linija teksta koja sadrži jednostavni aritmetički izraz dobijen primenom operacija sabiranja, oduzimanja ili množenja na dva prirodna broja. Učitani tekst ne sadrži razmake. Napomena: vežbe radi, učitati ceo string koji sadrži izraz, a zatim ga raščlaniti na delove.

**Izlaz**

Na standardni izlaz ispisati vrednost izraza (ona je sigurno između  $-10^{**9}$  i  $10^{**9}$ ).

**Primer****Ulaz**

32+73

**Izlaz**

105

**4.2.3.4. Brojevne osnove**

Napisati program kojim se uneti prirodan broj prikazuje u binarnom, oktalnom i heksadekadnom sistemu.

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se prirodan broj  $n$  ( $0 \leq n \leq 2 \cdot 10^9$ ).

**Izlaz**

U prvoj liniji standardnog izlaza nalazi se binarni zapis, u drugoj liniji oktalni zapis i u trećoj liniji heksadekadni zapis broja  $n$ . U heksadekadnom zapisu koristiti velika slova.

**Primer****Ulaz**

234

**Izlaz**

11101010

352

EA

**4.2.3.5. Spreadsheet kolone**

U programima za tabelarna izračunavanja (Microsoft Excel, LibreOffice Calc i slično) kolone su obeležene slovima i to kao A, B, ..., Z, AA, AB, ..., AZ, BA, BB, ..., ZZ, AAA, ... Napiši program koji omogućava konverziju rednog broja (od 1 pa naviše) u tekstualnu oznaku kolone i obratno.

**Ulaz**

Sa standardnog ulaza se unosi broj  $n$  ( $1 \leq n \leq 10$ ), a zatim  $n$  stringova (svaki u posebnom redu) koji sadrže velika slova engleskog alfabeta (najviše 5). Nakon toga se unosi  $n$  brojeva (svaki u posebnom redu) iz intervala od 1 do 12356630.

**Izlaz**

Na standardni izlaz ispisati prvo  $n$  brojeva, pa zatim  $n$  stringova (svaki u posebnom redu) koji odgovaraju unetim stringovima tj. brojevima.

**Primer****Ulaz**

3  
D  
AB  
ZZZZZ  
18  
100  
1000000

**Izlaz**

4  
28  
12356630  
R  
CV  
BDWGN

**4.3. Matrice****4.3.1. Turnir**

Tokom jedne košarkaške sezone neki timovi su se sastajali više puta. Ako je poznat rezultat svakog njihovog susreta, napiši program koji određuje ukupan rezultat svih njihovih susreta nakon te sezone.

**Ulaz**

U prvom redu standardnog ulaza unosi se ukupan broj odigranih utakmica  $n$  ( $1 \leq n \leq 100$ ). U sledećih  $n$  redova unose se po 4 nenegativna cela broja  $p, q, r, s$  koji opisuju jednu utakmicu, tako da su  $p$  i  $q$  ( $1 \leq p, q \leq 10$ ) redni brojevi košarkaških timova koji su odigrali tu utakmicu s rezultatom  $r:s$  ( $0 \leq r, s \leq 150$ ) tj. tim  $p$  je dao  $r$  koševa, dok je tim  $q$  dao  $s$  koševa. Nakon toga se unosi prirodni broj  $m$  ( $1 \leq m \leq 20$ ), a potom  $m$  parova prirodnih brojeva  $p$  i  $q$  ( $1 \leq p, q \leq 10$ ).

**Izlaz**

Za svaki par timova  $p$  i  $q$ , u posebnoj liniji ispiši koliko ukupno koševa je tim  $p$  dao timu  $q$  i koliko je tim  $q$  dao timu  $p$ , razdvojene dvotačkom.

**Primer****Ulaz**

5  
1 2 15 70  
2 3 90 88  
1 2 23 9  
3 1 88 86

2 3 42 50

2

1 2

3 2

**Izlaz**

38:79

138:132

#### 4.3.2. Broj bombi u okolini

U igrici Minesweeper na poljima se nalaze sakrivene bombe i zadatak igrača je da ih pronađe. Igraču se prikazuje tabla sa brojevima gde svaki broj predstavlja broj bombi koje se nalaze u okolini datog polja (gledaju se okolna polja u svih 8 smerova). Tvoj zadatak je da započneš programiranje ove igre tako što ćeš napisati program koji za dati raspored bombi određuje ove brojeve.

**Ulaz**

Sa standardnog ulaza se učitavaju dva broja  $m$  i  $n$  koji predstavljaju dimenzije table ( $3 \leq m, n \leq 100$ ) i nakon toga matrica koja ima  $m$  vrsta i  $n$  kolona koja sadrži nule i jedinice (gde jedinica označava bombu).

**Izlaz**

Matrica dimenzije  $m \times n$  koja određuje broj bombi u okolini svakog polja polazne matrice.

#### Primer

**Ulaz**

3 4

0 1 0 1

1 0 1 0

0 1 0 0

**Izlaz**

2 2 3 1

2 4 3 2

2 2 2 1

#### 4.3.3. Preslikavanja matrice

Deca su spremala priredbu u školskom dvorištu. Svako je obukao kostim različite boje i poredali su se u kvadratnu matricu. Tokom tačke deca su se premeštala i to na veoma pravilne načine. Prvo su se preslikali matricu oko glavne dijagonale. Zatim su zarotirali matricu za 90 stepeni udesno (u smeru kazaljke na satu). Nakon toga su je preslikali horizontalno i na kraju su je preslikali oko sporedne dijagonale. Napiši program koji ispisuje raspored boja dečijih kostima nakon svakog koraka njihove koreografije.

**Ulaz**

Sa standardnog ulaza se učitava broj [Math Processing Error] ([Math Processing Error]), a zatim i kvadratna matrica dimenzije [Math Processing Error] koja sadrži brojeve između 0 i 10 (svaki broj označava jednu boju kostima).

**Izlaz**

Na standardni izlaz ispisuje se 5 matrica. Polazna, učitana i zatim matrica dobijena nakon svake transformacije.

**Primer****Ulaz**

```
3
1 2 3
4 5 6
7 8 9
```

**Izlaz**

```
1 2 3
4 5 6
7 8 9
1 4 7
2 5 8
3 6 9
3 2 1
6 5 4
9 8 7
9 8 7
6 5 4
3 2 1
1 4 7
2 5 8
3 6 9
```

**4.3.4. Da li se dame napadaju**

Data je šahovska tabla na kojoj je raspoređeno osam dama. Napiši program koji proverava da li se neke dve dame napadaju (dve dame se napadaju ako se nalaze u istoj vrsti, istoj koloni ili na istoj dijagonali).

**Ulaz**

Sa standardnog ulaza učitava se [Math Processing Error] matrica dimenzije [Math Processing Error] čijih 8 jedinica opisuje položaj 8 dama.

**Izlaz**

Na standardnom izlazu ispisati tekst NE ako se dame ne napadaju ili DA ako se neke dve dame napadaju.



**Primer 1***Ulaz*

```

0 0 0 0 0 1 0 0
0 0 0 1 0 0 0 0
0 0 0 0 0 0 1 0
1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0
0 0 0 0 1 0 0 0
0 0 1 0 0 0 0 0

```

*Izlaz*

NE

**Primer 2***Ulaz*

```

0 0 0 0 0 1 0 0
0 0 0 1 0 0 0 0
0 0 0 0 0 0 1 0
1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 1
0 1 0 0 0 0 0 0
0 0 1 0 0 0 0 0
0 0 0 0 1 0 0 0

```

*Izlaz*

DA

**4.3.5. Najveća dijagonala**

Kvadratna matrica sadrži cele brojeve. Napiši program koji pronalazi najveći zbir brojeva na nekoj od dijagonala matrice (u obzir se uzimaju glavna dijagonala, sporedna dijagonala i sve dijagonale njima paralelne).

*Ulaz*

Sa standardnog ulaza se učitava broj  $n$  ( $2 \leq n \leq 100$ ) i nakon toga elementi kvadratne matrice dimenzije  $n \times n$  (celi brojevi između  $-10$  i  $10$ ).

*Izlaz*

Na standardni izlaz ispisati jedan ceo broj - najveći zbir elemenata na nekoj od dijagonala.

**Primer***Ulaz*

```

3
10 -10 2
1 -9 3

```

-2 9 9

**Izlaz**

12

Najveći zbir ima sporedna dijagonala koja sadrži elemente 9 i 3.

#### 4.3.6. Spiralni ispis matrice

Napiši program koji ispisuje elemente matrice spiralno, krenuvši od gornjeg levog ugla i kružeći u smeru kazaljke na satu.

**Ulaz**

Sa standardnog ulaza učitava se dimenzija kvadratne matrice [Math Processing Error], a zatim i elementi matrice.

**Izlaz**

Na standardni izlaz ispisati niz elemenata matrice koji se dobijaju spiralnim obilaskom.

#### Primer

**Ulaz**

5  
1 2 3 4 5  
6 7 8 9 10  
11 12 13 14 15  
16 17 18 19 20  
21 22 23 24 25

**Izlaz**

1 2 3 4 5 10 15 20 25 24 23 22 21 16 11 6 7 8 9 14 19 18 17 12 13

#### 4.3.7. Množenje matrica

Jovani je rođendan i odlučila je da napravi tortu da počasti svoje drugarice i drugare. Ona zna recepte za  $t$  različitih torti. U svaku tortu ide s različitih sastojaka (za svaku tortu poznata je količina svakog od tih sastojaka). Jovana će sve stvari nabaviti u jednoj od  $p$  prodavnica. Za svaku prodavnicu poznate su cene svakog od tih sastojaka. Napiši program koji pomaže Jovani da odredi koju tortu da pravi i u kojoj prodavnici da kupuje sastojke da bi joj ostalo što više para za ekskurziju.

**Ulaz**

Sa standardnog ulaza unose se brojevi  $t$ ,  $s$  i  $p$  (svi između 2 i 10), a zatim i dve matrice. Prva, dimenzije  $t \times s$  određuje količinu sastojaka za svaku od torti (količina je ceo broj između 1 i 3), a druga, dimenzije  $s \times p$  određuje cenu svakog sastojka u svakoj od prodavnica (cena je ceo broj između 100 i 300).

**Izlaz**

Na standardni izlaz ispisati tri cela broja - redni broj torte i redni broj prodavnice (oba se broje od 0), kao i ukupnu cenu najjeftinije torte.

**Primer****Ulaz**

```

3 4 2
2 1 3 2
3 2 2 1
2 1 2 3
250 170
160 120
135 142
145 155

```

**Izlaz**

```
1 1 1189
```

**4.3.8. Relacija zavisnosti**

Na fakultetu postoje zavisnosti između ispita koji se polažu. Na primer, da bi se polagao ispit "Programiranje 2", prethodno je neophodno da je položen ispit "Programiranje 1". Zavisnosti su zadate matricom logičkih vrednosti koja predstavlja jednu relaciju. Napiši program koji proverava da li su ispunjeni sledeći uslovi:

relacija je antirefleksivna tj. ne postoji ni jedan predmet koji zavisi sam od sebe,

relacija je antisimetrična tj. nije moguće da jedan predmet zavisi od nekog drugog predmeta, a da taj drugi predmet istovremeno zavisi od tog prvog predmeta,

relacija je tranzitivna tj. ako jedan predmet zavisi od drugog, a taj drugi zavisi od trećeg, onda je neophodno da i prvi predmet zavisi od trećeg.

**Ulaz**

Sa standardnog ulaza unosi se ukupan broj predmeta  $n$  ( $5 \leq n \leq 10$ ), a zatim i matrica dimenzije  $n \times n$  koja sadrži samo nule i jedinice - jedinica u vrsti  $v$  i koloni  $k$  označava da predmet sa rednim brojem  $v$  zavisi od predmeta sa rednim brojem  $k$ .

**Izlaz**

Na standardni izlaz ispisati DA ako matrica ispunjava sve zadate uslove tj. NE ako narušava bilo koji od njih.

**Primer****Ulaz**

```

6
0 0 0 0 0 0
1 0 1 0 1 0
1 0 0 0 1 0
1 1 1 0 1 0
1 0 0 0 0 0
0 0 0 0 0 0

```

**Izlaz**

DA

**4.3.9. Trouglovi i široka dijagonala**

Učenici tri odeljenja su sela na stolice u bioskopskoj sali kvadratnog oblika. Učenici jednog odeljenja su popunili mesta na glavnoj dijagonali sale i nekoliko dijagonala uz nju, učenici drugog odeljenja su popunili gornji, a trećeg odeljenja donji trougao sale. Na primer, ako je dimenzija sale 7, a ako su učenici prvog odeljenja seli na dve dijagonale uz glavnu, onda je raspored sedenja sledeći:

1112222

1111222

1111122

3111112

3311111

3331111

3333111

Poznata je visina svakog učenika. Napiši program koji određuje prosečnu visinu učenika svakog odeljenja.

**Ulaz**

Sa standardnog ulaza učitava se broj  $n$  koji predstavlja dimenziju sale ( $5 \leq n \leq 10$ ), zatim, kvadratna matrica dimenzije  $n \times n$  koja sadrži visine učenika (realne brojeve između 1.1 i 1.80) i na kraju, u sledećem redu, broj  $m$  koji određuje broj dijagonala oko glavne na kojima su raspoređeni učenici prvog odeljenja ( $0 \leq m \leq 3$ ).

**Izlaz**

Na standardni izlaz ispisati prosečnu visinu učenika odeljenja 1, odeljenja 2 i odeljenja 3 (svaki broj u posebnom redu, zaokružen na dve decimale).

**Primer****Ulaz**

7

1.31 1.58 1.13 1.49 1.70 1.17 1.36

1.39 1.78 1.42 1.38 1.42 1.54 1.68

1.28 1.10 1.42 1.52 1.79 1.19 1.33

1.18 1.44 1.33 1.23 1.55 1.33 1.59

1.24 1.79 1.17 1.33 1.76 1.19 1.44

1.63 1.50 1.41 1.11 1.56 1.61 1.80

1.51 1.76 1.48 1.33 1.74 1.30 1.55

2

**Izlaz**

1.43

1.45

1.48

#### 4.3.10. Vrhovi planine

Planinarska ekspedicija se kretala pravougaonim terenom i u pravilnim intervalima merila je nadmorsku visinu. Pretpostavljamo da su prikupljene visine u matrici dimenzija  $m \times n$ . Kada se nalaze u nekoj tački, planinari mogu da gledaju na istok, zapad, sever i jug, ali i na severoistok, severozapad, jugoistok i jugozapad. U svakom od tih 8 smerova planinari mogu da vide nekoliko vrhova. Tačka se smatra vidljivom, ako se ispred nje nalaze samo tačke koje imaju strogo manju nadmorsku visinu od nje (ili ako ispred nje nema drugih tačaka).

##### **Ulaz**

Opis ulaznih podataka.

##### **Izlaz**

Opis izlaznih podataka.

#### **Primer**

##### **Ulaz**

```
4 5
3 8 5 4 2
4 1 7 6 5
2 1 4 8 9
3 3 7 5 2
1 2
```

##### **Izlaz**

```
11
```

Planinar iz tačke na kordinatama (1,2) koja je na visini 7, u pravcu ka zapadu vidi tačke sa nadmorskim visinama 1 i 4, u pravcu zapada samo tačku sa nadmorskom visinom 6, u pravcu ka severu opet vidi tačku sa nadmorskom visinom 5, u pravcu ka jugu tačke sa visinama 4 i 7, u pravcu ka severozapadu vidi tačku sa nadmorskom visinom 8, kao severoistoku tačku sa visinom 4, u pravcu ka jugozapadu vidi tačke sa visinama 1 i 3, a u pravcu ka jugoistoku vidi samo tačku sa nadmorskom visinom 8. Dakle, ukupno vidi 11 tačaka.

### 4.4. Asocijativni nizovi

#### 4.4.1. Različite cifre

Napisati program kojim se ispituje da li su sve cifre u dekadnom zapisu datog prirodnog broja različite?

##### **Ulaz**

Sa standardnog ulaza unosi se prirodan broj od 0 do  $2 \cdot 10^{**9}$ .

**Izlaz**

Na standardnom izlazu ispisuje se tekstualni odgovor DA ili NE.

**Primer 1****Ulaz**

67569

**Izlaz**

NE

**Primer 2****Ulaz**

1234567890

**Izlaz**

DA

**4.4.2. Svega sa različitim ciframa**

Napisati program kojim se određuje koliko ima prirodnih brojeva u intervalu  $[m,n]$  kojima su sve cifre različite.

**Ulaz**

Sa standardnog ulaza u prvoj liniji se učitava prirodan broj  $m$ , a u sledećoj prirodan broj  $n$ . Broj  $n$  je veći od broja  $m$ . Vrednosti učitanih brojeva su u granicama od 100 do 10000.

**Izlaz**

Na standardnom izlazu se ispisuje koliko je brojeva sa različitim ciframa u zadatom intervalu.

**Primer****Ulaz**

120

130

**Izlaz**

9

**4.4.3. Frekvencija znaka**

Napisati program koji čita jednu reč teksta koju čine samo velika slova i ispisuje slovo koji se najčešće pojavljuje i koliko puta se pojavljuje. Ako se više slova najčešće pojavljuje, ispisuje se slovo koji se pre pojavilo u reči.

**Ulaz**

U jednoj liniji standardnog ulaza nalazi se jedna reč teksta sa ne više od 20 slova.

**Izlaz**

U prvoj liniji standardnog izlaza prikazati slovo koje se najčešće pojavljuje, a u drugoj liniji standardnog ulaza ispisati i koliko puta se pojavljuje.

**Primer 1****Ulaz**

POPOKATEPETL

**Izlaz**

P  
3

**Primer 2****Ulaz**

BACACB

**Izlaz**

B  
2

**4.4.4. Dvojke i trojke deljive sa 3**

U datom nizu prirodnih brojeva dužine  $n$ , odredi broj grupa od dva ili tri elementa takvih da je suma svih elemenata grupe deljiva sa 3. Na primer u nizu  $[2,1,3,10]$  imamo 4 takve grupe:  $[2,1]$ ,  $[2,10]$ ,  $[2,1,3]$ ,  $[2,3,10]$ .

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se broj elemenata niza  $n$  ( $1 \leq n \leq 1000$ ). U sledećih  $n$  linija nalazi se redom elementi niza (prirodni brojevi između 0 i 1000).

**Izlaz**

Na standardnom izlazu u jednoj liniji prikazati broj traženih grupa u nizu.

**Primer****Ulaz**

4  
2  
1  
3  
10

**Izlaz**

4

#### 4.4.5. Histogram

Grupa biologa amatera je posmatrala populaciju žaba. Za svaku žabu su zabeležili veličinu. Biloge je zanimala raspodela veličina žaba (koliko je bilo malih žaba, koliko malo većih, koliko najvećih i slično). Zato su interval veličina  $[a,b]$  podelili na  $n$  delova jednake dužine i izbrojali koliko žaba je pripadalo svakom od tih delova (ako je veličina žabe tačno na granici između dva dela, pridružuje se desnom). Napiši program koji im u tome pomaže i rezultat prikazuje grafički, u obliku histograma.

##### **Ulaz**

Sa standardnog ulaza se učitavaju brojevi  $a$  i  $b$  razdvojeni jednim razmakom, a zatim u sledećem redu i broj  $n$ . Nakon toga unosi se broj žaba  $k$ , a zatim i  $k$  brojeva iz intervala  $[a,b)$  koje predstavljaju veličine žaba.

##### **Izlaz**

Na standardni izlaz se prikazuju se podeoci (sa granicama zaokruženim na tri decimale), nakon čega se prikazuje broj žaba čija je veličina u tom podeoku, a zatim i procentualni udeo žaba u tom podeoku u odnosu na ukupan broj žaba (tako što se taj procenat zaokurži na najbliži ceo broj i zatim ispiše toliko karaktera  $*$ ). Zvezdice su od broja žaba odvojene jednim tabulatorom (znakom Tab).

#### **Primer 1**

##### **Ulaz**

```
0.0 1.0
4
10
0.05
0.25
0.95
0.93
0.56
0.73
0.10
0.47
0.23
0.15
```

##### **Izlaz**

```
[0.000, 0.250): 4 *****
[0.250, 0.500): 2 *****
[0.500, 0.750): 2 *****
[0.750, 1.000): 2 *****
```

U prvom podeoku se nalaze žabe čije su veličine 0.05, 0.23, 0.10 i 0.15, u drugom 0.25 i 0.47, u trećem 0.56 i 0.73, a u četvrtom 0.95 i 0.93. U prvom podeoku se nalazi 40% žaba, a u ostalim po 20%, pa je ispisan i odgovarajući broj zvezdica.



**Primer 2****Ulaz**

10.0 20.0

3

3

15.3

17.2

19.1

**Izlaz**

[10.000, 13.333): 0

[13.333, 16.667): 1 \*\*\*\*\*

[16.667, 20.000): 2

\*\*\*\*\*

\*\*\*\*

U drugom podeoku je žaba veličine 15.3, a u trećem 17.2 i 19.1. U prvom podeoku se nalazi trećina ukupnog broja žaba pa su ispisane 33 zvezdice dok se u drugom podeoku nalazi dve trećine ukupnog broja žaba pa je ispisano 67 zvezdica.

**4.4.6. Frekvencije reči**

Milica je kupovala razne proizvode i svaki put kada se vratila iz kupovine dopisivala je sve proizvode koje je kupila na spisak. Napiši program koji pomaže Milici da odredi proizvod koji je najčešće kupovala tokom prethodne godine.

**Ulaz**

Sa standardnog ulaza učitavaju se nazivi proizvoda, sve dok se ne dođe do kraja ulaza. Svaki naziv je reč koja sadrži najviše 10 malih slova engleskog alfabeta.

**Izlaz**

Na standardni izlaz ispisati jedan red koji sadrži naziv najčešće kupovanog proizvoda, jedan razmak i broj puta koliko je taj proizvod kupljen. Ako je više proizvoda kupljeno isti broj puta, ispisati onaj koji je prvi po abecednom redosledu.

**Primer****Ulaz**

jabuka

hleb

kruska

jabuka

sljiva

hleb

mleko

jabuka

hleb

**Izlaz**

hleb 3

#### 4.4.7. Izomorfne niske

Dve niske su izomorfne ako se druga može dobiti nekim 1-1 preslikavanjem slova prve - svakom slovu abecede odgovara neko slovo (njegova slika u tom preslikavanju), svako pojavljivanje tog slova u prvoj niski se zamenjuje tom slikom, pri čemu ne postoje dva slova koja imaju istu sliku. Na primer, reči filip i cilim su izomorfne (f se slika u c, i u i, l u l i p u m), dok reči filip i madam nisu izomorfne (slovo m bi trebalo da bude slika i slova f i slova p, što nije dopušteno). Napiši program koji ispituje da li su dve unete reči izomorfne.

**Ulaz**

Sa standardnog ulaza se učitavaju dve reči sastavljene od malih slova engleske abecede, svaka u posebnom redu.

**Izlaz**

Na standardni izlaz ispisati da ako su reči izomorfne, tj. ne u suprotnom.

#### Primer 1

**Ulaz**

filip  
leden

**Izlaz**

da

#### Primer 2

**Ulaz**

filip  
melem

**Izlaz**

ne

## 5. Tipovi

### 5.1. Celobrojni

#### 5.1.1. Sredina intervala

Na planeti M u ulici M žive mravi. Sve kućice u ulici su označene prirodnim brojevima. Mravica Ana živi u kućici broj  $a$ , a mrav Boris iza nje, u kućici broj  $b$ . Mrav Viktor živi u kućici tačno na sredini između Ane i Borisa. Napiši program koji određuje broj kućice u kojoj živi Viktor (ako je između Ane i Borisa paran broj kućica, Viktor živi u onoj koja je bliža Ani).

##### *Ulaz*

Sa standardnog ulaza unose se dva cela broja:

$a$  ( $0 \leq a \leq 2 \cdot 10^9$ ) - broj kućice u kojoj živi Ana

$b$  ( $a+2 \leq b \leq 2 \cdot 10^9$ ) - broj kućice u kojoj živi Boris

##### *Izlaz*

Na standardni izlaz ispisati jedan ceo broj - broj kućice u kojoj živi Viktor.

#### **Primer 1**

##### *Ulaz*

432457

562321

##### *Izlaz*

497389

#### **Primer 2**

##### *Ulaz*

2

9

##### *Izlaz*

5

#### **Primer 3**

##### *Ulaz*

2000000000

1999999998

##### *Izlaz*

1999999999

### 5.1.2. Provera prekoračenja

Napiši program koji određuje da li će doći do prekoračenja ili potkoračenja prilikom sabiranja i oduzimanja dva data broja tipa int.

#### *Ulaz*

Sa standardnog ulaza se učitavaju dva broja, svaki u posebnoj liniji. Svaki broj se ispravno može reprezentovati tipom int tj. Integer na računaru na kojem se vrši testiranje.

#### *Izlaz*

U prvoj liniji standardnog izlaza ispisati da ako se ta dva broja mogu ispravno sabrati tj. ne ako prilikom sabiranja dolazi do prekoračenja ili potkoračenja. U drugoj liniji standardnog izlaza ispisati da ako se ta dva broja mogu ispravno oduzeti tj. ne ako prilikom oduzimanja dolazi do prekoračenja ili potkoračenja.

#### **Primer 1**

##### *Ulaz*

```
1000000000
1000000000
```

##### *Izlaz*

```
da
da
```

#### **Primer 2**

##### *Ulaz*

```
2000000000
2000000000
```

##### *Izlaz*

```
ne
da
```

#### **Primer 3**

##### *Ulaz*

```
2000000000
-2000000000
```

##### *Izlaz*

```
da
ne
```

U primerima se pretpostavlja da se za zapis tipa int koriste 32 bita (to ne mora uvek biti tako).

### 5.1.3. Faktorijel 64bit

Napiši program koji izračunava vrednosti faktorijela datog celog broja.

**Ulaz**

Sa standardnog ulaza unosi se broj  $n$  ( $1 \leq n \leq 20$ ).

**Izlaz**

Na standardni izlaz ispisuje se jedan broj  $n!$ .

**Primer****Ulaz**

20

**Izlaz**

2432902008176640000

**5.1.4. Jabuke u trouglu**

Jabuke su poslagane u trougao sledećeg oblika:

\*

\*\*

\*\*\*

\*\*\*\*

Napiši program koji određuje ukupan broj jabuka u trouglu.

**Ulaz**

Sa standardnog ulaza unosi se ceo broj  $n$  ( $3 \cdot 10^{**9} \leq 6 \cdot 10^{**9}$ ) koji određuje broj redova trougla.

**Izlaz**

Na standardni izlaz ispisati jedan ceo broj koji predstavlja ukupan broj jabuka u trouglu.

**Primer 1****Ulaz**

3500000000

**Izlaz**

6125000001750000000

**Primer 2****Ulaz**

5000000000

**Izlaz**

12500000002500000000

## 5.2. Karakteri

### 5.2.1. Klasifikacija karaktera

Napisati program koji za učitani ASCII karakter ispituje da li je malo slovo, veliko slovo, cifra ili nešto četvrto

*Ulaz*

U jednoj liniji standardnog ulaza nalazi se jedan karakter.

*Izlaz*

U jednoj liniji standardnog izlaza prikazati jednu od sledećih informacija: MALO SLOVO, VELIKO SLOVO, CIFRA, OSTALO.

#### Primer 1

*Ulaz*

a

*Izlaz*

MALO SLOVO

#### Primer 2

*Ulaz*

B

*Izlaz*

VELIKO SLOVO

#### Primer 3

*Ulaz*

7

*Izlaz*

CIFRA

#### Primer 4

*Ulaz*

:

*Izlaz*

OSTALO

### 5.2.2. Transformacija karaktera

Napisati program kojim se transformiše učitani karakter tako što se malo slovo prebacuje u veliko, veliko slovo u malo, a ostali karakteri se ne menjaju.

**Ulaz**

U prvoj liniji standardnog ulaza nalazi se jedan karakter.

**Izlaz**

Na standardnom izlazu prikazati transformisan karakter.

**Primer****Ulaz**

t

**Izlaz**

T

**5.2.3. Abecedno ogledalo**

Katarina je odlučila da svojoj drugarici pošalje šifrovanu poruku, koja sadrži samo slova engleske abecede, cifre i interpunkcijske znake. Svako slovo će šifrovati posebno na osnovu narednih pravila. Mala slova se šifruju velikim slovima tako što se slovo a šifruje slovom Z, slovo b šifruje slovom Y, c slovom X itd., sve do slova y koje se šifruje slovom B i z koje se šifruje slovom A. Velika slova se šifruju potpuno analogno - od A koje se šifruje sa z do Z koje se šifruje sa a. Ostali karakteri se ne menjaju.

**Ulaz**

Sa standardnog ulaza unosi se jedna linija teksta, završena karakterom tačka (karakterom .).

**Izlaz**

Na standardni izlaz ispisati šifrovani tekst (bez karaktera tačka).

**Primer****Ulaz**

Zdravo svima.

**Izlaz**

aWIZEL HERNZ