



## Natjecanje timova studenata informatičara hrvatskih sveučilišta

Zagreb, Osijek, Dubrovnik, Rijeka, Pula

23. listopada 2016.

### Zadaci

A: Blokada . . . . .	1
B: Caki . . . . .	3
C: Gorivo . . . . .	4
D: Kocka . . . . .	5
E: Marica . . . . .	6
F: Ograda . . . . .	7
G: Pčele . . . . .	8
H: Roj . . . . .	9
I: Rudar . . . . .	11
J: Vjerojatnost . . . . .	13

## Zadatak A: Blokada

Vremensko ograničenje: 1 s

Memorijsko ograničenje: 512 MiB

Studenti nekog fakulteta su, zbog razmirica s dekanom, odlučili organizirati blokadu prostora fakulteta. To znači da određeni broj studenata mora biti fizički prisutan na fakultetu tijekom 24 sata. Organizatori bi među sobom željeli podijeliti dežurstva na blokadi što je moguće poštenije, pa su osmislili slijedeća pravila:

- Svaki će dan raspored dežurstva biti isti.
- Svaki student može biti dežuran samo kada je slobodan od drugih obaveza.
- Svaki student sam ograničava koliko može ukupno vremena biti dežuran u jednom danu (od ponoći do ponoći).
- Zbog jednostavnosti, dežurstvo može početi i završiti samo na pola sata ili puni sat (na primjer 16:00 ili 16:30, ali ne i 16:15).
- Student/ica moraju biti slobodni tijekom cijelog dežurstva (na primjer, ako studentica kaže da je slobodna tek od 16:05, ona u rasporedu ne može dežurati od 16:00).
- Minimalni broj studenata koji dežuraju tijekom bilo kojeg dijela dana je maksimiziran. To povećava izgleda za uspješnu blokadu.

Napišite program koji će omogućiti organizatorima blokade da utvrde koliki je maksimalan broj studenata  $M$  takav da je moguće napraviti raspored koji zadovoljava gornja ograničenja, a da u svakom trenutku dana (od ponoći do ponoći) dežura barem  $M$  studenata.

Možete pretpostaviti da je zamjena studenata koji dežuraju trenutna, tj. da ako dvoje studenata napuste blokadu a zamijene ih drugo dvoje, možete smatrati da je tijekom smjene dežurstva na blokadi cijelo vrijeme bilo dvoje studenata.

### Ulazni podaci

U prvom redu nalazi se cijeli broj  $n$  ( $1 \leq n \leq 50$ ), broj studenata koji žele sudjelovati u blokadi. Nakon toga slijedi  $n$  blokova gdje svaki blok opisuje želje i mogućnosti jednog studenta.

Svaki blok počinje s dva broja,  $k_i$  ( $1 \leq k \leq 50$ ) i  $m_i$  ( $1 \leq m_i \leq 1440$ ), redom broj vremenskih perioda u kojima je  $i$ -ti student slobodan od drugih obaveza i može sudjelovati u blokadi te najveći ukupni broj minuta koje je spreman posvetiti dežuranju u jednom danu.

U preostalih  $k_i$  redova u bloku, nalaze se vremenski trenunci početka i kraja uzastopnog perioda kada je student slobodan, odvojeni razmakom. Periodi se mogu preklapati, a njihova unija predstavlja vrijeme kada student može biti prisutan na blokadi.

Vremenski su trenunci formatirani kao "HH:MM" ( $00 \leq \text{HH} \leq 23$ ,  $00 \leq \text{MM} \leq 59$ ). Ponoć je predstavljena s "00:00". Kada je kraj prije početka, to znači da student može ostati na blokadi i preko ponoći. Na primjer, period "23:00 03:00" znači da student može biti prisutan od 23 sata do idućeg jutra u 3. Ako su početak i kraj jednaki, to znači da student može biti svih 24 sata na blokadi.

### Izlazni podaci

Potrebno je ispisati jedan cijeli broj – traženi najveći broj studenata  $M$  koji može biti u svakom trenutku na blokadi, uz ranije opisane pretpostavke.



## Primjeri test podataka

### ulaz

3  
1 540  
00:00 00:00  
3 480  
08:00 10:00  
09:00 12:00  
13:00 19:00  
1 420  
17:00 00:00

### izlaz

1

### ulaz

5  
1 720  
18:00 12:00  
1 1080  
00:00 23:00  
1 1080  
00:00 20:00  
1 1050  
06:00 00:00  
1 360  
18:00 00:00

### izlaz

2

### ulaz

3  
1 1440  
00:00 00:00  
1 720  
00:00 12:15  
1 720  
12:05 00:15

### izlaz

1

**Pojašnjenje 3. primjera:** Prvi student može biti na svih 24 sata blokade, a drugi i treći na 12. Međutim, kako dežurstva mogu početi i završiti samo svakih pola sata, niti drugi niti treći student ne mogu pokriti termin od 12:00 do 12:30.

## Zadatak B: Caki

Vremensko ograničenje: 1 s

Memorijsko ograničenje: 512 MiB

Mali Caki obožava utrke bolida Formule 1, a nova sezona upravo je počela. Prošle je nedjelje Caki sjeo ispred štale, izvukao antenu na svom radiju, usmjerio je u obližnji repetitor i pomno slušao prijenos utrke. I sve bi bilo krasno da se komentator utrke nije odlučio našaliti sa svojim slušateljima tako da je na početku prijenosa najavio da će jedine informacije koje će prenositi u eter biti *oznaka* bolida koji prestiže bolid ispred sebe i to upravo u trenutku prestizanja. Staza na kojoj se utrka odvija je kružna, a pretpostavljamo da su svi bolidi stigli na cilj nakon što su odvozili propisani broj krugova.

Iz dobivenih informacije odredite koji su bolidi zauzeli prvih šest mjesta na kraju utrke. Primijetite da je moguće da bolid prestigne druge bolide za cijeli krug ili više krugova.

### Ulazni podaci

U prvom redu nalaze se dva prirodna broja  $n$  i  $k$ , ( $2 \leq n \leq 1000$ ,  $1 \leq k \leq 10000$ ) — broj bolida na stazi te broj prestizanja tijekom utrke. Bolidi su označeni brojevima od 1 do  $n$ , i to tako da je svaki bolid označen upravo onim brojem koji predstavlja njegovu startnu poziciju.

U svakom od sljedećih  $k$  redaka nalazi se po jedan prirodni broj  $x$  ( $1 \leq x \leq n$ ) — oznaka bolida koji je prestigao bolid ispred sebe. Tih  $k$  brojeva poredani su upravo onim redoslijedom kako su se prestizanja i događala.

### Izlazni podaci

U prvi red ispišite oznake bolida koji su zauzeli prvih šest mjesta na kraju utrke. Prvi broj je oznaka bolida koji je zauzeo prvo mjesto, ..., šesti broj je oznaka bolida koji je zauzeo šesto mjesto, a ti su brojevi odvojeni s po jednim razmakom. Ako je broj bolida manji od šest, potrebno je ispisati točno onoliko brojeva koliko je bolida sudjelovalo u utrci.

### Primjeri test podataka

**ulaz**

4 3  
3  
4  
3

**izlaz**

3 1 4 2

**ulaz**

10 4  
4  
4  
4  
1

**izlaz**

1 4 2 3 5 6

**ulaz**

5 10  
1  
1  
1  
3  
2  
2  
2  
5  
4  
4

**izlaz**

2 1 3 4 5



## Zadatak C: Gorivo

Vremensko ograničenje: 1 s

Memorijsko ograničenje: 512 MiB

Mirko je nedavno otputovao u Ameriku i tamo je iznajmio moderan američki kabriolet narančaste boje s bijelim crtama, automatskim mjenjačem, navigacijom, i . . . prikaznikom na engleskom jeziku s imperijalnim sustavom mjera. Najviše ga muči to što je potrošnja goriva izražena u *miljama po galonu*, pa mu je teško procijeniti koliko mu mašina zapravo "guta" benzina.

Napišite program koji će preračunati potrošnju goriva iz broja milja po galonu u broj litara po 100 kilometara.

**Napomena:** 1 galon jednak je točno 3.785411784 litara, a 1 milja jednaka je točno 1609.344 metara.

### Ulazni podaci

U prvom i jedinom retku ulaznih podataka nalazi se decimalni broj  $x$  ( $1 \leq x \leq 1000$ ) s dvije decimale, koji predstavlja koliko milja po galonu prijeđe Mirkov automobil.

### Izlazni podaci

Potrebno je ispisati jedan decimalan broj, koliko litara troši automobil na 100 kilometara.

Tolerirat će se apsolutno i relativno odstupanje od službenog rješenja za  $10^{-6}$ .

### Primjeri test podataka

**ulaz**

10.00

**izlaz**

23.521458

## Zadatak D: Kocka

Vremensko ograničenje: 1 s

Memorijsko ograničenje: 512 MiB

Potrebno je, pomoću ASCII znakova, napraviti grafički prostorni prikaz hrpe poslaganih kockica. Kocke su pravilno složene u  $m$  redaka i  $n$  stupaca, a na nekim kockama se nalazi jedna ili više drugih kocaka koje na taj način formiraju tornjeve. Redci su označeni redom brojevima od 1 do  $m$  tako da redak broj 1 označava najudaljeniji, a redak broj  $m$  najbliži redak na grafičkom prikazu. Stupci su označeni redom brojevima od 1 do  $n$ , slijeva na desno. Neke kocke zaklanjaju druge kocke koje su onda *djelomično ili potpuno skrivene*. Jednu kocku prikazujemo pomoću znakova "+" (plus), "-" (minus), "|" (okomita crta), "/" (kosa crta) i " " (razmak) u 6 redaka i 7 stupaca na sljedeći način:

```
  +---+
  /   /|
+---+ |
|   | +
|   | /
+---+
```

Napišite program koji će odrediti grafički prikaz cijele zadane konfiguracije, koristeći pri tome što je moguće manje redaka i stupaca. Prazna polja označite znakom "." (točka).

### Ulazni podaci

U prvom redu se nalaze prirodni brojevi  $m$  i  $n$  ( $1 \leq m, n \leq 50$ ). U svakom od sljedećih  $m$  redova nalazi se  $n$  prirodnih brojeva. Svaki od njih je manji od ili jednak 50, a označava visinu tj. ukupni broj naslaganih kockica na toj poziciji.

### Izlazni podaci

Ispišite grafički prikaz zadanih kocaka u prostoru kako je opisano u tekstu zadatka.

### Primjeri test podataka

**ulaz**1 2  
1 2**izlaz**

```
.....+---+
...../   /|
....+---+ |
..+ -|   | +
./   |   | /
+---+---+ |
|   |   | +
|   |   | /
+---+---+..
```

**ulaz**3 1  
2  
1  
3**izlaz**

```
.....+---+
..+---+ /|
./   /| -+ |
+---+ | | +
|   | + | /|
|   | /| -+ |
+---+ | /| +
|   | + | /
|   | /| +..
+---+ | /...
|   | +....
|   | /.....
+---+.....
```

**ulaz**2 3  
2 3 2  
1 2 1**izlaz**

```
.....+---+.....
...../   /|.....
.....+---+ |.....
....+ -|   | +---+
.../   |   | / /|
..+---+---+---+ |
..| /   /|   | +
..| +---+ |   | /|
..+ -|   | +---+ |
./   |   | / /| +
+---+---+---+ | /
|   |   |   | +..
|   |   |   | /...
+---+---+---+.....
```

## Zadatak E: Marica

Vremensko ograničenje: 1 s

Memorijsko ograničenje: 512 MiB

Maričina baka ima veliki voćnjak i svakog jutra na tržnicu nosi šljive. Marica je jutros baki nabrala  $n$  košara šljiva, ali baka se još nije probudila zbog sinoćnjeg tulumarenja, pa se Marica želi još malo poigrati. Neke će šljive iz košara pojesti, a potom ubrati još šljiva iz voćnjaka.

Maričin je cilj da za svaki prirodan broj iz intervala  $[A, B]$  (uključivo) postoji barem jedna košara koja sadrži taj broj šljiva. Znajući trenutni broj šljiva u svakoj košari, odredite minimalan broj operacija koje Marica mora učiniti da bi ostvarila svoj zadatak, ako je svaka od operacija jedno od sljedećeg:

- pojedi šljivu iz neke košare,
- uberi šljivu iz voćnjaka i stavi je u neku košaru.

### Ulazni podaci

U prvom redu nalazi se prirodni broj  $n$  ( $n \leq 5000$ ) — broj košara. U sljedećem redu nalaze se prirodni brojevi  $A$  i  $B$  ( $1 \leq A \leq B \leq 10^6$ ,  $B - A + 1 \leq n$ ) iz teksta zadatka. U  $i$ -tom od sljedećih  $n$  redova nalazi se cijeli broj  $a_i$  ( $1 \leq a_i \leq 10^6$ ), broj šljiva u  $i$ -toj košari.

### Izlazni podaci

U prvi i jedini red ispišite traženi minimalan broj operacija.

### Primjeri test podataka

<b>ulaz</b>	<b>ulaz</b>
5	7
3 6	64 68
8	62
7	5
1	97
10	66
9	74
	47
<b>izlaz</b>	86
11	<b>izlaz</b>
	45

## Zadatak F: Ograda

Vremensko ograničenje: 3 s

Memorijsko ograničenje: 512 MiB

Pašnjak na kojem uspijevaju faktorijeli smješten je u standardni koordinatni sustav u kojem  $x$  koordinata raste nadesno, a  $y$  koordinata prema gore. *Polje* je kvadrat duljine stranice jedan, sa stranicama paralelnim s koordinatnim osima, takav da su koordinate njegovog donjeg lijevog vrha  $A = (x_A, y_A)$  oba nenegativni cijeli brojevi. *Vrijednost* svakog takvog polja je broj  $x_A!y_A!$ , gdje je  $a! = 1 \times 2 \times \dots \times a$  za  $a \geq 1$ , dok je  $0! = 1$ .

*Ograda* je poligon sa stranicama paralelnim s koordinatnim osima. Za zadanu ogradu izračunajte zbroj vrijednosti svih polja unutar ograde modulo  $10^9 + 7$ .

### Ulazni podaci

U prvom redu nalazi se prirodni broj  $n$  ( $n \leq 100$ ) — broj vrhova ograde. U  $k$ -tom od sljedećih  $n$  redova nalaze se cijeli brojevi  $x_k$  i  $y_k$  ( $0 \leq x_k, y_k \leq 10^9$ ) — koordinate  $k$ -tog po redu vrha ograde. Možete pretpostaviti da vrhovi čine poligon čije su stranice paralelne s koordinatnim osima. Između ostalog, poligon ne siječe samog sebe te dvije susjedne stranice nikada nisu paralelne.

### Izlazni podaci

U prvi red ispišite traženi zbroj modulo  $10^9 + 7$ .

### Primjeri test podataka

**ulaz**

4  
1 1  
1 2  
2 2  
2 1

**izlaz**

1

**ulaz**

4  
3 3  
1 3  
1 1  
3 1

**izlaz**

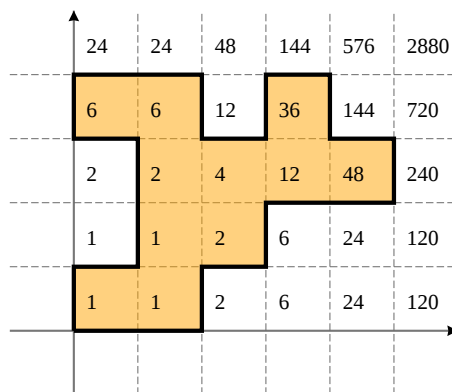
9

**ulaz**

18  
0 0  
2 0  
2 1  
3 1  
3 2  
5 2  
5 3  
4 3  
4 4  
3 4  
3 3  
2 3  
2 4  
0 4  
0 3  
1 3  
1 1  
0 1

**izlaz**

119



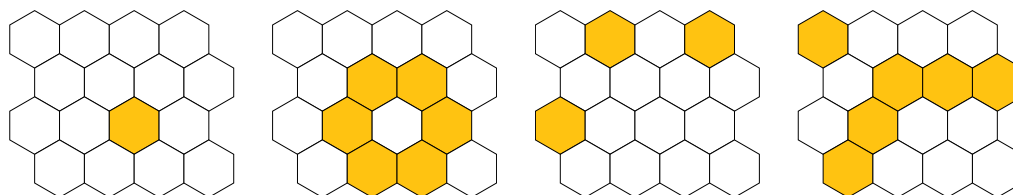


## Zadatak G: Pčele

Vremensko ograničenje: 1 s

Memorijsko ograničenje: 512 MiB

Pčele svakog dana pohranjuju med u saće – mrežu voštanih ćelija koja se sastoji od pravilnih šesterokuta organiziranih u retke i stupce. Tijekom noći, ličinke pojedu sav med iz saća tako da je na početku svakog dana saće potpuno prazno. Malo je poznata činjenica da pčele ne pohranjuju med u nasumične ćelije, budući da raspored ovisi izravno o tome koje su ćelije sadržavale med prethodnog dana. Ćelija će, naime, sadržavati med ako i samo ako je broj susjednih ćelija (ne uključujući i promatranu ćeliju) koje su sadržavale med prethodnog dana bio neparan. U suprotnom će ćelija biti prazna.



Ilustracija prvog primjera test podataka

### Ulazni podaci

U prvom se redu nalaze tri prirodna broja,  $n$ ,  $m$  i  $k$  ( $2 \leq n, m \leq 10$ ,  $1 \leq k \leq 2^{63} - 1$ ), redom broj redaka i stupaca saća te broj dana.

U idućih se  $n$  redaka nalazi po  $m$  znakova '#' (ljestve) i '.' (točka) koji označuju ima li (ljestve) ili nema (točka) meda u danoj ćeliji saća na početku. Saće je uvijek zadano tako da je prva ćelija prvog retka "gore lijevo" od prve ćelije drugog retka, kao na slici u tekstu zadatka.

### Izlazni podaci

Potrebno je ispisati stanje saća nakon  $k$  dana u istom formatu u kojem je saće zadano na ulazu.

### Primjeri test podataka

**ulaz**

4 4 3

....

....

..#.

....

**izlaz**

#...

.###

.#..

#...

**ulaz**

3 10 8

###.#...

###.#...

...#.#...

**izlaz**

###...##.

.#.#...###

...###.###.

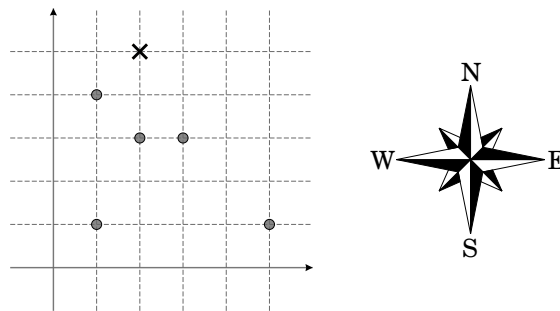
## Zadatak H: Roj

Vremensko ograničenje: 10 s

Memorijsko ograničenje: 512 MiB

Olujno nevrijeme razbacalo je roj pčela po koordinatnoj ravnini te se svaka pčela trenutno nalazi u nekoj točki s cjelobrojnim koordinatama. Sve pčele žele se naći u istoj točki, također sa cjelobrojnim koordinatama. Pčele iz ovog roja dijele genetsku mutaciju zbog koje mogu letjeti samo u određenim smjerovima. Dozvoljeni smjerovi su podskup skupa svih glavnih i sporednih strana svijeta. Kao i obično, sjever je smjer rastaće  $y$  koordinate dok je istok smjer rastaće  $x$  koordinate.

U svakom koraku, jedna pčela može se pomaknuti na susjednu točku sa cjelobrojnim koordinatama u jednom od dozvoljenih smjerova. Pronađite najmanji ukupan broj koraka potreban da se sve pčele nađu u istoj točki.



Ilustracija prvog primjera test podataka

### Ulazni podaci

U prvom redu nalazi se cijeli broj  $d$  ( $1 \leq d \leq 8$ ) — broj dozvoljenih smjerova. Drugi red sadrži  $d$  različitih nizova znakova odvojenih s po jednim razmakom. Svaki niz znakova je jedan od N, NW, W, SW, S, SE, E, NE koji redom označavaju sjever, sjeverozapad, zapad, jugozapad, jug, jugoistok, istok i sjeveroistok.

U sljedećem redu nalazi se prirodni broj  $n$  ( $1 \leq n \leq 50$ ) — broj pčela. U  $k$ -tom od sljedećih  $n$  redova nalaze se cijeli brojevi  $x_k$  i  $y_k$  ( $-10^6 \leq x_k, y_k \leq 10^6$ ) — koordinate točke u kojoj se trenutno nalazi  $k$ -ta pčela. Možete pretpostaviti da se sve pčele nalaze u različitim točkama.

### Izlazni podaci

Ispišite traženi najmanji ukupan broj koraka. Možete pretpostaviti da rješenje uvijek postoji.



## Primjeri test podataka

**ulaz**

3  
N NE NW  
5  
1 1  
5 1  
1 4  
3 3  
2 3

**izlaz**

13

**ulaz**

3  
N SE SW  
3  
-4 1  
-4 -2  
4 -2

**izlaz**

16

**ulaz**

6  
N E W SW NE NW  
3  
249 218  
443 -290  
231 -444

**izlaz**

1013

**ulaz**

3  
S SE NE  
4  
437 -368  
387 -47  
-422 136  
-407 -300

**izlaz**

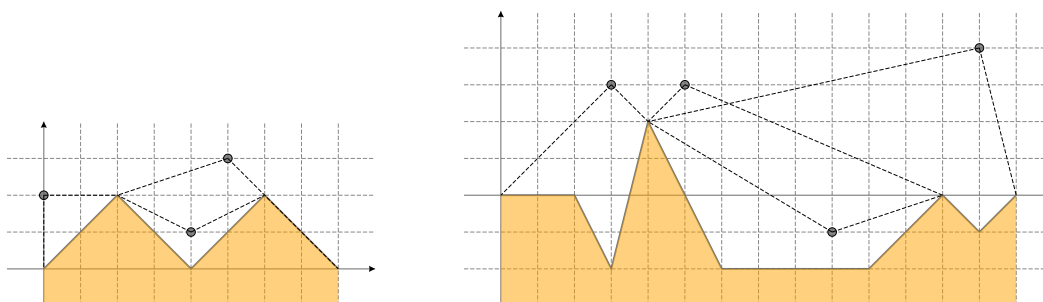
2025

## Zadatak I: Rudar

Vremensko ograničenje: 1.5 s

Memorijsko ograničenje: 512 MiB

Pod rudarske špilje zamišljamo kao izlomljenu liniju u koordinatnom sustavu koja se sastoji od  $n + 1$  vrhova  $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$  te  $n$  dužina koje povezuju uzastopne vrhove. Pod počinje u ishodištu, proteže se slijeva nadesno te završava na  $x$ -osi ( $0 = x_0 < x_1 < \dots < x_n, y_0 = y_n = 0$ ). Dužina koja povezuje vrh  $(x_{i-1}, y_{i-1})$  sa vrhom  $(x_i, y_i)$  je opisana sa dva cijela broja  $d_i$  i  $k_i$  — redom duljinom projekcije dužine na  $x$ -os te koeficijentom smjera pravca određenog dužinom. Dakle, točka  $(x_i, y_i)$  je jednaka  $(x_{i-1} + d_i, y_{i-1} + k_i d_i)$ .



Ilustracije primjera test podataka

Rudari mogu objesiti lampu u bilo koju točku  $A$  iznad poda špilje. Kažemo da lampa osvjetljava točku  $B$  na podu špilje ako dužina  $\overline{AB}$  ne siječe pod špilje (dozvoljeno je da dužina  $\overline{AB}$  dodiruje pod špilje u točkama ili uzduž segmenata). Jasno je da lampa osvjetljava točku  $A'$  koja se nalazi na podu špilje neposredno ispod  $A$ . Glavni interval lampe u točki  $A$  je područje poda špilje do kojeg rudari mogu doći šetnjom po podu špilje počevši iz točke  $A'$ , a da cijelo vrijeme budu osvjetljeni lampom u točki  $A$ .

Zadano je  $q$  mogućih pozicija lampe. Za svaku poziciju odredite lijevi i desni rub glavnog intervala lampe na toj poziciji.

### Ulazni podaci

U prvom redu nalaze se prirodni brojevi  $n$  i  $q$  ( $1 \leq n, q \leq 100\,000$ ) — broj segmenata od kojih se sastoji pod špilje te broj mogućih pozicija lampe.

U  $i$ -tom od sljedećih  $n$  redova nalaze se dva cijela broja  $d_i$  i  $k_i$  ( $1 \leq d_i \leq 10^9, -10^9 \leq k_i \leq 10^9$ ) — duljina projekcije i koeficijent smjera  $i$ -tog segmenta. Možete pretpostaviti da su parametri takvi da za vrhove poda špilje vrijedi  $0 = x_0 < x_1 < \dots < x_n \leq 10^9, y_0 = y_n = 0, -10^9 \leq y_i \leq 10^9$ .

U  $i$ -tom od sljedećih  $q$  redova nalaze se dva cijela broja  $x'_i, y'_i$  ( $0 \leq x'_i \leq x_n, -10^9 \leq y'_i \leq 10^9$ ) — koordinate  $i$ -te potencijalne pozicije. Sve pozicije se nalaze strogo iznad poda špilje.

### Izlazni podaci

Ispišite  $q$  redova. U  $i$ -ti red ispišite dva cijela broja — redom  $x$ -koordinate lijevog i desnog ruba glavnog intervala lampe u  $i$ -toj potencijalnoj poziciji. Možete pretpostaviti da su tražene koordinate uvijek cijeli brojevi.



## Primjeri test podataka

**ulaz**

4 3  
2 1  
2 -1  
2 1  
2 -1  
0 2  
5 3  
4 1

**izlaz**

0 2  
2 8  
2 6

**ulaz**

8 4  
2 0  
1 -2  
1 4  
2 -2  
4 0  
2 1  
1 -1  
1 1  
5 3  
9 -1  
3 3  
13 4

**izlaz**

4 12  
4 12  
0 4  
4 14



## Zadatak J: Vjerojatnost

Vremensko ograničenje: 1.5 s

Memorijsko ograničenje: 512 MiB

Mali Perica oduševljen je novom računalnom igrom "Slovčeka". U toj je igri na dnu ekrana niz od  $n$  (početno praznih) polja, a na vrhu ekrana redom se pojavljuju nezavisno slučajno odabrana slova iz skupa  $\{A, B, C, D\}$ . Čim se pojavi neko slovo, Perica ga treba smjestiti u neko polje niza koje je još uvijek prazno. Igra završava kada su sva polja popunjena, a Peričin cilj je da slova u popunjenom nizu budu poredana uzlazno po abecedi.

Znajući vjerojatnosti pojavljivanja slova  $A, B, C$  i  $D$  na ekranu, izračunajte vjerojatnost da će Perica, igrajući optimalno, uspjeti ostvariti cilj igre.

### Ulazni podaci

U prvom redu nalazi se prirodni broj  $n$  ( $n \leq 250$ ) — broj polja u nizu, ujedno i broj padajućih slova.

U drugom redu nalaze se cijeli brojevi  $a, b, c, d$  ( $0 \leq a, b, c, d \leq 100$ ,  $a + b + c + d = 100$ ) — postotne vjerojatnosti pojavljivanja slova  $A, B, C, D$  redom.

### Izlazni podaci

U prvi i jedini red ispišite traženu vjerojatnost.

Tolerirat će se apsolutno i relativno odstupanje od službenog rješenja za  $10^{-6}$ .

### Primjeri test podataka

**ulaz**

3  
25 25 25 25

**izlaz**

0.75

**ulaz**

10  
0 50 0 50

**izlaz**

1

**ulaz**

5  
10 20 30 40

**izlaz**

0.62116