

ZADATAK	PERO	VATROGASCI	PIKULE
izvorni kôd	pero.pas pero.c pero.cpp	vatrogasci.pas vatrogasci.c vatrogasci.cpp	pikule.pas pikule.c pikule.cpp
ulazni podaci	standardni ulaz		
izlazni podaci	standardni izlaz		
vremensko ograničenje	1 sekunda		
memorijsko ograničenje	256 MB		
broj bodova	40	50	60
	150		



Pero, vaš najbolji prijatelj, radi na novoj aplikaciji za mobilne uređaje koja će omogućiti još jednostavnije i preglednije dopisivanje tekstualnim porukama. Zamolio vas je da mu pomognete napisati dodatak aplikaciji koji će omogućiti **poravnavanje teksta udesno**.

Vaš dio aplikacije učitat će rečenicu koju je potrebno poravnati, zajedno sa širinom ekrana, izraženom brojem znakova N koje je moguće prikazati u jednom retku ekrana. Program treba ispisati rečenicu *razlomljenu* na više redaka na sljedeći način:

- U prvi redak dodajemo riječ po riječ redoslijedom kojim dolaze u rečenici te između svake dvije riječi stavljamo točno jedan znak '.' (točka). Dodajemo što je više moguće riječi u prvi redak, ali tako da duljina retka (ukupan broj znakova uključujući točke) bude manja ili jednaka N .
- Ako je potrebno, prvi redak *poravnavamo udesno* tako da dodamo odgovarajući broj točaka na početak kako bi redak imao točno N znakova. Naravno, ako redak već ima točno N znakova nije potrebno dodavati točke.
- Prelazimo u drugi redak i nastavljamo isti postupak sve dok ne iskoristimo sve riječi u rečenici.
- Zadnji redak također treba poravnati udesno koristeći isti postupak.

Napiši program koji zadanu rečenicu ispisuje razlomljenu i poravnatu udesno tako da se u svakom retku nalazi točno N znakova kako je opisano u tekstu zadatka.

ULAZNI PODACI

U prvom retku ulaza nalazi se prirodan broj N ($1 \leq N \leq 80$), tražena duljina svakog retka. U sljedećem retku nalazi se jedna rečenica: niz riječi razdvojen razmacima. Između svake dvije riječi nalazi se točno jedan razmak, a riječ se sastoji samo od malih i velikih slova engleske abecede. Rečenica će se sastojati od najviše 1000 znakova, a svaka riječ će biti dugačka najviše N znakova.

IZLAZNI PODACI

Ispišite tekst razlomljen i poravnat udesno prema opisu iz teksta zadatka.

PRIMJERI TEST PODATAKA

ulaz

5

Prvi TEST nema rijec duzu od pet slova

izlaz

.Prvi
.TEST
.nema
rijec
.duzu
...od
..pet
slova

ulaz

10

Ovo je primjer RECENICE koja moze doci kao test ZA vas Program SRETNO

izlaz

....Ovo.je
...primjer
..RECENICE
.koja.moze
..doci.kao
...test.ZA
.....vas
...Program
....SRETNO

Goran se upravo uključio u dobrovoljno vatrogasno društvo i već je dobio prvi zadatak! Njegove algoritamske vještine nadaleko su poznate, pa je zamoljen da odredi lokacije za K budućih vatrogasnih domova kako bi sve kuće u selu bile što bolje zaštićene u slučaju požara. Svaki od K domova smjestit će se u jednoj od postojećih kuća. Kuće su označene prirodnim brojevima od 1 do N^1 (prema redoslijedu u ulaznim podacima) te su za svaku kuću A poznate njene koordinate - dva cijela broja, X_A i Y_A .

Goran vas je upravo nazvao i moli vas za pomoć, jer je otišao na put, a zaboravio ponijeti svoj laptop. Preko telefona ste dobili sljedeće objašnjenje algoritma koji Goran želi iskoristiti za odabir domova:

1. U prvom koraku, za privremena mjesta K vatrogasnih domova odabiru se lokacije prvih K kuća redom prema oznakama. Za privremenu lokaciju prvog vatrogasnog doma odabrat će se lokacija prve kuće, za lokaciju drugog vatrogasnog doma lokacija druge kuće itd.
2. Nakon toga je potrebno svaku kuću "pridružiti" **njoj najbližem** vatrogasnom domu. Udaljenost definiramo kao zbroj apsolutnih vrijednosti razlika koordinata (udaljenost kuće A i kuće B je $|X_A - X_B| + |Y_A - Y_B|$). Ukoliko postoji više domova s najmanjom udaljenosti, kuća se pridružuje domu s **najmanjom oznakom**. Kažemo da sve kuće pridružene i -tom vatrogasnom domu čine *i -tu vatrogasnu grupu*.
3. Za svaku vatrogasnu grupu, pronalazi se novoprivremena pozicija njenog vatrogasnog doma tako da se odabere ona kuća unutar grupe toga doma čiji je **ukupni zbroj udaljenosti** do svih ostalih kuća unutar promatrane vatrogasne grupe **najmanji**. Ponovo, u slučaju izjednačenja ukupnih udaljenosti, potrebno je birati kuću s **najmanjom oznakom**.
4. Ako se u koraku 3 pozicija vatrogasnog doma u **barem jednoj** od grupa promijenila, onda postupak ponavljamo tj. idemo na korak 2. U suprotnom, algoritam završava te pozicije vatrogasnih domova u tom trenutku postaju konačne.

Ovaj postupak ima svojstvo da sigurno završava i to nakon malog broja iteracija.

Napiši program koji za zadane lokacije kuća u selu te traženi broj vatrogasnih domova K pronalazi **konačne pozicije** vatrogasnih domova koje Goranov algoritam određuje. Test podaci će biti takvi da će Goranov algoritam za njih završavati nakon **najviše 30 iteracija** (jedna iteracija se sastoji od drugog, trećeg i četvrtog koraka).

ULAZNI PODACI

U prvom retku ulaza nalaze se prirodni brojevi N i K ($1 \leq K \leq N \leq 100$), broj kuća u Goranovom selu i broj vatrogasnih domova čije je lokacije potrebno odrediti. U svakom od sljedećih N redaka nalazi se po jedan par cijelih brojeva X_A , Y_A ($1 \leq X_A, Y_A \leq 1000$), koordinate A -te kuće. Niti jedan par kuća neće se nalaziti na istim koordinatama.

IZLAZNI PODACI

Potrebno je ispisati K redova. U A -ti red potrebno je ispisati par brojeva X_A , Y_A odvojenih razmakom - koordinate konačne pozicije A -tog vatrogasnog doma.

¹ U originalnom tekstu zadatka je greškom stajalo '... do K ...'.

PRIMJERI TEST PODATAKA

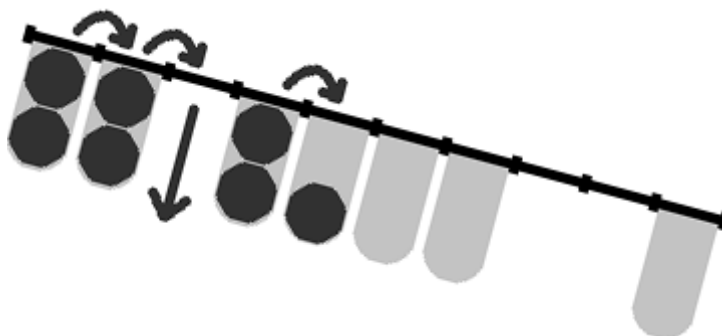
ulaz 4 2 4 5 9 8 4 2 10 7 izlaz 4 5 9 8	ulaz 9 3 8 4 10 2 8 10 5 6 1 5 6 5 7 5 7 9 10 4 izlaz 6 5 10 2 8 10
---	---

Mirku i Slavku su dosadile postojeće igračke, pa su odlučili sami konstruirati jednu novu. Skupili su nešto starih dasaka, čavala, epruveta i paket pikula (staklenih kuglica u nekim krajevima Hrvatske poznatih kao “franje” ili “špekule”), kojeg je Mirko dobio za neki od proteklih Božića. Igračku koju su sagradili možemo zamisliti kao rešetku širine 1 i dužine N . Za svaku su ćeliju u rešetki zatim pričvrstili epruvetu u koju stane točno M pikula. Željeli su se brzo početi igrati pa su iz nehaja razbili K epruveta.



Poprečni presjek igračke dužine 10, s razbijenim epruvetama s oznakama 3, 8 i 9

Igra počinje! Mirko je igračku lagano nagnuo udesno, a zatim je Q puta zgrabio određenu količinu pikula te ih pokušao ubaciti u neku od epruveta. Naravno, moguće je da neke pikule neće stati u epruvetu u koju ih Mirko pokušava ubaciti pa će se višak “preliti” udesno. Pikule će tada završiti u prvoj epruveti udesno u kojoj ima slobodnog mjesta ili će pasti kroz prvu rupu ispod koje nije epruveta. Pikule se također mogu preliti preko krajnjeg desnog ruba igračke i pasti na tlo.



Rezultat nagnjanja igračke i ubacivanja pikula iznad određenih epruveta

Napišite program koji će na temelju opisa rešetke i opisa niza Mirkovih poteza ubacivanja izračunati broj epruveta u kojima se na kraju igre nalazi **barem jedna pikula** te ukupan broj pikula koje su tijekom igre **pale na tlo**.

ULAZNI PODACI

U prvom retku ulaza nalaze se prirodni brojevi N i M , ($1 \leq N \leq 1\,000\,000\,000$, $1 \leq M \leq 1\,000$) duljina rešetke i kapacitet svake epruvete.

U drugom retku ulaza nalaze se prirodni broj K ($1 \leq K \leq N$, $1 \leq K \leq 100\,000$), broj razbijenih epruveta, a zatim K prirodnih brojeva E_i ($1 \leq E_i \leq N$), redni brojevi razbijenih epruveta. Epruvete su označene brojevima od 1 do N s lijeva na desno. Brojevi E_i bit će poredani uzlazno.

U trećem retku ulaznih podataka nalazi se prirodni broj Q , broj Mirkovih ubacivanja pikula u epruvete ($1 \leq Q \leq 100\,000$). U sljedećih Q redova nalaze se po dva prirodna broja, X_i i S_i , redni broj epruvete i količina pikula koju Mirko pokušava ubaciti ($1 \leq X_i \leq N$, $1 \leq S_i \leq 100\,000$).

IZLAZNI PODACI

Potrebno je ispisati dva broja odvojena razmakom - broj epruveta koje nakon svih ubacivanja sadrže barem jednu pikulu te broj pikula koje su tijekom igre pale na tlo.

Napomena: Rješenje ne mora nužno stati u 32-bitni cjelobrojni tip podataka.

BODOVANJE

U skupu test podataka vrijednom ukupno 20% bodova, vrijedit će $N \leq 100$, $K \leq 100$ i $S_i \leq 100$.

U dodatnom skupu test podataka vrijednom ukupno 30% bodova, vrijedit će $N \leq 100\,000$.

PRIMJERI TEST PODATAKA

ulaz	ulaz
10 2	12 3
3 3 8 9	4 1 4 7 11
3	4
1 5	3 2
3 7	2 5
4 3	2 1
	8 8
izlaz	izlaz
4 8	5 2

Pojašnjenje prvog test primjera: *Primjer je prikazan na slikama. U prvom ubacivanju Mirko je zagrabilo pet pikula te time popunio prvu i drugu epruvetu, dok je peta pikula kroz rupu pala na tlo. U drugom ubacivanju Mirko je sve pikule ubacio u rupu na mjestu gdje je trebala biti treća epruveta. U trećem ubacivanju popunjena je četvrta epruveta te je jedna pikula završila u petoj epruveti. Konačno, četiri epruvete sadrže barem po jednu pikulu (one s oznakama 1, 2, 4 i 5), a ukupno je 8 pikula palo na tlo.*