

Secțiunea 9-10 avansați

PROBLEMA 1 TEXT

100 puncte

Un text este format din una sau mai multe propoziții separate pe linii. O propoziție este formată din două sau mai multe cuvinte separate prin câte un spațiu. Fiecare cuvânt este format doar din literele mici ale alfabetului englez.

Alinierea justify a unui text înseamnă inserarea de spații suplimentare între cuvintele unei propoziții astfel încât fiecare propoziție din text să ocupe același spațiu ca și cea mai lungă dintre propozițiile textului. Nu pot fi adăugate spații înainte de primul cuvânt și nici după ultimul, toate spațiile necesare vor fi inserate între cuvintele propoziției astfel încât diferența dintre numărul de spații dintre cuvintele alăturate să fie de cel mult 1. Dacă spațiile nu pot fi repartizate în număr egal între cuvintele propoziției, atunci între primele cuvinte se va insera câte un spațiu în plus.

Cerință

Pentru un text dat:

1. să se realizeze alinierea justify;
2. să se determine câte dintre subșirurile nevide ale primului cuvânt din text și diferite de acesta, se regăsesc ca și cuvinte în restul textului.

Date de intrare

Fișierul **text.in** conține pe prima linie o valoare naturală C care nu poate avea decât valoarea 1 sau 2, iar pe următoarele linii propozițiile unui text. Dacă C este 1 se rezolvă doar cerința 1, iar dacă C este 2 se rezolvă doar cerința 2.

Date de ieșire

Fișierul **text.out** va conține răspunsul la cerința rezolvată. Pentru prima cerință se va scrie textul aliniat, iar pentru cea de a doua cerință se va scrie numărul subșirurilor nevide ale primului cuvânt, regăsite ca și cuvinte în textul dat.

Restricții și precizări

- $1 \leq C \leq 2$
- Textul va conține cel mult 1000 de propoziții
- O propoziție are cel mult 100 de cuvinte
- Un cuvânt are cel mult 20 de litere

Exemple

text.in	text.out	Explicații
1 acest text trebuie aliniat justify pentru a arata mai bine	acest text trebuie aliniat justify pentru a arata mai bine	Cea mai lungă propoziție ocupa 23 de poziții. Astfel în prima propoziție a textului vor fi afișate 4 spații între primele două cuvinte și 3 spații între al doilea și al treilea cuvânt.
2 problema este usoara pentru aceasta proba nu este o lema	3	Există 3 subșiruri ale cuvântului problema care se regăsesc în text: lema o proba

Timp maxim de execuție: 0.7 secunde/test

Memorie totală disponibilă: 20 MB din care 20 MB pentru stivă

Dimensiunea maximă a sursei: 10Kb

Secțiunea 9-10 avansați

PROBLEMA 2 LOL

100 puncte

Octavian s-a apucat de un nou joc video, League of Legends. Din păcate, nu se pricepe prea bine și a primit diviza Iron, iar dorința lui este să ajungă Diamond sezonul acesta. Pentru a face acest lucru, trebuie să câștige cât mai multe meciuri. Pentru a câștiga, are nevoie de o echipă cât mai bună, pentru că nu se pricepe la acest joc și nu poate câștiga atunci când joacă de unul singur.

O echipă este formată din 5 membri, care joacă 5 roluri diferite: top, jungle, mid, ad carry și support. Octavian vrea să joace n meciuri la anumite ore și vrea să își formeze pentru fiecare meci câte o echipă cu 4 din cei k prieteni ai săi. Știind ce coeficient de skill au prietenii săi pentru fiecare rol și la ce ore sunt disponibili pentru a juca cu Octavian, calculați ce coeficient de skill are cea mai bună echipă pe care Octavian o poate forma pentru fiecare din cele n meciuri.

Cerința

Știind ce coeficient de skill are Octavian pentru fiecare rol și la ce ore vrea să joace cele n meciuri, afișați pentru fiecare meci care este cea mai bună echipă pe care Octavian o poate forma.

Date de intrare

Fișierul de intrare **lol.in** conține pe primul rând coeficienții de skill ai lui Octavian pentru fiecare rol. Pe al doilea rând este scris n, cu semnificația din enunț, iar pe următoarele n rânduri sunt scrise orele la care Octavian se va juca. Următorul rând conține numărul k, cu semnificația din enunț. Următoarele 2*k rânduri conțin coeficienții de skill ai prietenilor lui Octavian pentru fiecare rol, respectiv intervalul orar în care sunt disponibili să joace cu el.

Date de ieșire

Fișierul de ieșire **lol.out** conține n linii, pe linia i se află suma maximă a coeficienților de skill corespunzătoare echipelor pe care Mihai le poate forma în meciul i.

Restricții și precizări

- Pentru teste în valoare de 30 de puncte, $1 \leq n \leq 500$ și $4 \leq k \leq 2.500$
- Pentru alte teste în valoare de 30 de puncte, $1 \leq n \leq 1.000$ și $4 \leq k \leq 10.000$
- $1 \leq n \leq 10.000$ și $4 \leq k \leq 35.000$
- O zi are 2400 ore, o oră are 60 de minute și coeficienții de skill sunt numere între 1 și 3.500.000

Exemplu

lol.in	lol.out	Explicatii
5 2 2 3 1 1 14:06 6 14 12 14 12 19 13:47 13:51 10 18 20 16 19 14:07 20:12 6 2 7 6 4 12:15 14:11 6 15 6 5 20 14:04 14:15 3 1 4 16 12 14:04 14:07 11 9 5 14 10 14:03 14:17	57	Octavian va juca top și va face echipă cu prietenul 6 jungle, prietenul 3 mid, prietenul 5 ad carry și prietenul 4 support.

Timp maxim de execuție: 1 secundă/test.

Memorie totală disponibilă 64 MB din care 64 MB pentru stivă

Dimensiunea maximă a sursei: 5 Kb.

Secțiunea 9-10 avansați

PROBLEMA 3 FLEX

100 puncte

După ce Octavian a pierdut multe meciuri de LOL cu cea mai bună echipă pe care o putea alege, acesta decide să-și creeze propriul joc, "Flex,, ce constă în următoarele reguli:

1. Există un singur jucător, așa că nu poate pierde!
2. Jocul se desfășoară pe o hartă, în formă de pătrat de dimensiuni infinite, în care sunt plasați N campioni. Poziția fiecărui campion este dată prin coordonatele acestuia față de „Baza echipei”, considerată cu coordonatele $(0,0)$.
3. Jucătorul își alege un campion, și primește un coeficient de flex, în funcție de cum folosește abilitatea campionului respectiv.
4. Fiecare campion are abilitatea să-și schimbe poziția pe hartă cu orice alt campion și primește un coeficient de flex egal cu distanța dintre acești 2 campioni (campionul de pe poziția $(1,1)$ primește coeficientul 4 dacă își folosește abilitatea pe campionul de pe poziția $(1,5)$)

Octavian joacă un meci de Flex și dorește să aibă un coeficient de flex cât mai mare .

În cazul în care toți cei N campioni și-ar folosi abilitatea într-un mod optim pentru a obține un coeficient maxim, numărul de campioni diferiți ce sunt folosiți pentru schimbul de poziții este notat cu H .

Cerință

Octavian alege Q campioni și vă roagă, pentru fiecare din aceștia, să îi spuneți care este coeficientul de flex maxim pe care îl poate avea.

Date de intrare

Din fișierul **flex.in**, se citesc pe prima linie numerele naturale N și Q , cu semnificația din enunț. Pe fiecare dintre următoarele N linii se află coordonatele fiecărui campion. Pe linia $K+1$, se află coordonatele campionului V_k . Pe următoarele Q linii, se află indicele campionilor cu care vrea să joace Octavian. Pe linia $N+K+1$ se află indicele campionului Q_k

Date de ieșire

În fișierul **flex.out** se vor afișa Q valori. pe o singură linie, separate printr-un spațiu. A K -a valoare va reprezenta coeficientul de flex maxim pe care îl poate obține Octavian, alegând campionul Q_k .

Restricții și precizări

- $1 \leq N \leq 50000$
- $1 \leq Q \leq 100000$.
- $0 \leq X_i \leq 10000$, $0 \leq Y_i \leq 10000$ pentru orice valoare a lui i între 1 și N , unde X_i și Y_i reprezintă coordonatele campionului V_i și sunt numere reale, cu 2 zecimale
- $1 \leq N \leq 1000$ și $1 \leq Q \leq 1000$, pentru 20% din teste
- Numerele V_1, V_2, \dots, V_N sunt distincte.
- Numerele Q_1, Q_2, \dots, Q_N nu sunt neapărat distincte.
- $H \leq 100$
- Se vor afișa doar primele 2 zecimale ale fiecărui număr din fișierul de ieșire.
- Există cel puțin un set de 3 campioni care nu sunt coliniari

Secțiunea 9-10 avansați

Exemple

flex.in	flex.out	Explicații
3 3 0.00 1.00 0.00 2.00 1.00 1.00 1 2 3	1.00 1.41 1.41	1 ar folosi abilitatea pe 2 , 2 ar folosi abilitatea pe 3, 3 ar folosi abilitatea pe 2 , deci $H=3$
4 2 1.00 1.00 2.00 1.00 1.00 2.00 5.00 5.00 1 2	5.65 5.00	1 ar folosi abilitatea pe 4, 2 ar folosi abilitatea pe 4, 3 ar folosi abilitatea pe 4, 4 ar folosi abilitatea pe 1, deci $H=2$

Timp maxim de execuție: 0.7 secunde/test.

Memorie totală disponibilă 8 MB, din care 8 MB pentru stivă

Dimensiunea maximă a sursei: 5 Kb.

PROBLEMA 4 CONSILIU**100 puncte**

Zilele acestea, Bucureștiul găzduiește o mulțime de conferințe, manifestări culturale care țin de Președinția Consiliului UE. La una din conferințe, organizatorul a primit o cerință din partea cabinetului președintelui de consiliu.

Acesta a dorit o reșezare a persoanelor aflate pe primul rând din sala de conferințe, cu scopul de a genera o atmosferă diferită. Protocolul impune că așezarea persoanelor din primul rând să respecte principiul rangului - începând din partea dreaptă, toate persoanele trebuie așezate în ordinea descrescătoare a rangului protocolar.

Pentru schimbarea acestei ordini singura indicație pe care a primit-o organizatorul a fost aceea că poate schimba locul unor persoane, dar nu poate muta mai mult de numărul de persoane indicat în prețuia conferinței.

Cerință

Cunoscând N numărul de persoane, rangul acestora și K numărul maxim de persoane care pot fi mutate determinați numărul variantelor de așezare a celor N persoane. Pentru situația în care $K=N$ afișați varianta de așezare a celor N persoane, minimă lexicografic.

Date de intrare

Fișierul **consiliu.in** conține pe prima linie 2 numere naturale N și K reprezentând numărul de persoane aflate pe primul rând respectiv numărul maxim de persoane care pot fi mutate. Pe următoarea linie, sunt cele N ranguri protocolare, separate prin spațiu, reprezentând așezarea impusă de protocol.

Date de ieșire

Dacă $K < N$ în fișierul **consiliu.out** se va scrie numărul posibilităților de așezare care respectă condiția impusă. Rezultatul va fi afișat modulo 666013.

Secțiunea 9-10 avansați

Dacă $K=N$ toate persoanele trebuie să fie mutate din poziția „oficială” și în fișierul „consiliu.out” se va scrie varianta cea mai mică din punct de vedere lexicografic de reșezare a persoanelor (ordinea lexicografică este dată de rangul persoanelor), rangurile fiind separate prin câte un spațiu.

Restricții și precizări

- $0 \leq K \leq N$
- $5 \leq N \leq 1000$
- Rangurile protocolare sunt numere naturale distincte < 5000 și sunt date în ordine crescătoare
- Pentru rezolvarea corectă a cazului $K=N$ se acorda 20 puncte

Exemple

consiliu.in	consiliu.out	Explicații
4 2 6 13 77 89	7	Sunt 7 posibilități de așezare în care să fie mutate maxim 2 persoane {6, 13, 77, 89}, {13, 6, 77, 89}, {77, 13, 6, 89}, {89, 13, 77, 6}, {6, 77, 13, 89}, {6, 89, 77, 13} și {6, 13, 89, 77}
3 3 6 13 77	13 77 6	Modalitățile de așezare sunt {6, 13, 77}, {6, 77, 13}, {13, 6, 77}, {13, 77, 6}, {77, 6, 13}, {77, 13, 6}. Dar numai 2 corespund cerinței ca toate persoanele să își schimbe poziția inițială (13, 77, 6), (77, 6, 13). Cea mai mică lexicografică este: (13, 77, 6)

Timp maxim de execuție: 1 secundă/test.

Memorie totală disponibilă 2 MB, din care 2 MB pentru stivă

Dimensiunea maximă a sursei: 5 Kb.