

Clasa a 6-a, Problema 1

Enunțul problemei

Astăzi la ora de informatică, Nastia a aflat despre $CMMDC$ și $CMMMC$. Nastia este foarte perspicace, așa că a rezolvat toate sarcinile momentan și acum vă sugerează să rezolvați și voi una dintre ele. Definim o pereche de numere întregi (a, b) interesantă, dacă $CMMDC(a, b) = x$ și $CMMMC(a, b) = y$, unde $CMMDC(a, b)$ **cel mai mare divizor comun** al lui a și b , iar $CMMMC(a, b)$ **cel mai mic multiplu comun** al lui a și b .

Dându-se un interval închis $[st, dr]$ și două numere întregi x, y , voi trebuie să găsiți câte perechi interesante de numere întregi (a, b) se află în intervalul dat.

Date de intrare

Fișierul de intrare **nastia.in** conține patru numere naturale st, dr, x și y , separate printr-un spațiu, având semnificația din enunț.

Date de ieșire

Fișierul de ieșire **nastia.out** va conține un singur număr natural N , reprezentând numărul de perechi interesante din intervalul $[st, dr]$.

Restricții și precizări

- $1 \leq st \leq dr \leq 109$
- $1 \leq x \leq y \leq 109$
- Perechile (a, b) și (b, a) sunt considerate soluții distincte

Exemplu

nastia.in	nastia.out
1 2 1 2	2 Explicație: În primul exemplu, există două perechi interesante de numere întregi (a, b): (1, 2) și (2, 1).
1 12 1 12	4 Explicație: În cel de-al doilea exemplu, există patru perechi interesante de numere întregi (a, b): (1, 12), (12, 1), (3, 4) și (4, 3).
50 100 3 30	0 Explicație: În al treilea exemplu, există perechi interesante de numere întregi (de exemplu: 3, 30), dar niciuna dintre ele nu se află în intervalul $[st, dr]$.

Timp de execuție: **0.1 s**; Memorie disponibilă: **64 MB**; Stivă disponibilă: **1 MB**;

Rate

Concursul interjudetean Memorial „Stefan Dârtu”
decembrie 2018
Vatra Dornei

Clasa a 6-a, Problema 2

Enunțul problemei

Vânătorul șef al regelui Arthur a primit însărcinarea să vâneze primele rațe ce se întorc din țările calde. Regele fiind un tip cu idei fixe i-a cerut vânătorului să vâneze rațele albe cu săgeți albe, iar rațele negre cu săgeți negre.

Rațele vin în stoluri din ce în ce mai mari: mai întâi una, apoi două, trei, cinci, opt ș.a.m.d. Rațele fiind niște creaturi ordonate zboară în rânduri lungi, în care nu vei putea găsi două rațe de aceeași culoare alăturate, fiecare rând începând cu o rață albă. Vânătorul știe că dacă a început să doboare un rând de rațe trebuie să le doboare pe toate deoarece supraviețuitoarele vor alerta celelalte rațe și ele nu se vor mai întoarce niciodată, iar vânătorul nostru își va pierde slujba.

Știind că vânătorul a primit x săgeți albe și y săgeți negre, voi va trebui să determinați câte rânduri de rațe a doborât și câte săgeți de fiecare tip i-au rămas știind că el vrea să-și păstreze slujba.

Date de intrare

Fișierul de intrare **rate.in** conține două numere naturale x și y , separate printr-un spațiu, reprezentând numărul de săgeți albe, respectiv negre.

Date de ieșire

Fișierul de ieșire **rate.out** va conține trei numere naturale afișate pe linii separate, reprezentând numărul de rânduri doborâte, numărul de săgeți albe rămase și numărul de săgeți negre rămase.

Restricții și precizări

○ $0 \leq x, y \leq 2 * 10^9$

Exemplu

rate.in	rate.out
9 10	4 2 6

Explicație: Vânătorul are săgeți suficiente pentru a doborî complet primele 4 rânduri de rate (cele formate din 1, 2, 3, 5 rațe). După ce a doborât cele 4 rânduri, el rămâne cu 2 săgeți albe (câte una e folosită pentru primele două rânduri, două pentru al treilea și trei pentru al 4-lea) și 6 negre (0 la primul rând, câte una la rândurile 2, 3 și două la rândul 4).

Timpe de execuție: **0.05 s**; Memorie disponibilă: **64 MB**; Stivă disponibilă: **2 MB**;

Clasa a 7-a, Problema 1

Enunțul problemei

Ion joacă (din nou!) jocul său preferat cu împușcături. Personajul său are la brâu N puști, așezate în N huse speciale, numerotate de la 1 la N . Pușca din husa i are puterea pb_i ($1 \leq i \leq N$). În camera armelor a găsit M puști, așezate pe perete, în M locații, numerotate de la 1 la M . Pentru fiecare pușcă j ($1 \leq j \leq M$) este cunoscută puterea sa pc_j .

Ion poate înlocui puștile pe care le are la brâu cu puști aflate pe perete în camera armelor. La o înlocuire el ia pușca de pe perete din locația j ($1 \leq j \leq M$) și o pune la brâu în husa i ($1 \leq i \leq N$), iar pușca din husa i o pune pe perete în locația j .

Scrieți un program care să determine suma maximă a puterilor puștilor pe care le va avea la brâu Ion după efectuarea înlocuirilor.

Date de intrare

Fișierul de intrare **joc.in** va conține pe prima linie două numere naturale N și M reprezentând numărul de puști pe care le are la brâu, respectiv numărul de puști aflate în camera armelor. Pe a doua linie se află N numere naturale $pb_1 pb_2 \dots pb_N$ reprezentând în ordine puterile puștilor pe care Ion le are la brâu. Pe a treia linie se află M numere naturale $pc_1 pc_2 \dots pc_M$ reprezentând în ordine puterile puștilor aflate în camera armelor. Numerele scrise pe aceeași linie sunt separate prin spațiu.

Date de ieșire

Fișierul de ieșire **joc.out** va conține o singură linie pe care va fi scrisă suma maximă a puterilor puștilor de la brâul lui Ion, după efectuarea înlocuirilor.

Restricții și precizări

- $1 \leq N, M < 1000$
- Puterile puștilor sunt numere naturale ≤ 10000 .

Exemplu

joc.in	joc.out
3 2	16
3 1 7	
4 5	

Timpe de execuție: 0.03 s; Memorie disponibilă: 64 MB; Stivă disponibilă: 2 MB;

Străzi

Concursul interjudetean Memorial „Stefan Dârtu”
decembrie 2018
Vatra Dornei

Clasa a 7-a, Problema 2

Enunțul problemei

Gigel primește o nouă provocare de la prietenul său Programatorul! Cunoscând înălțimile clădirilor aflate pe o anumită stradă, Gigel trebuie să răspundă rapid la întrebarea: “Care este gradul de vizibilitate al străzii?”

Gradul de vizibilitate al unei străzi este dat de raportul dintre numărul clădirilor vizibile din capătul stâng al străzii și numărul total al clădirilor de pe stradă, raport calculat cu trei zecimale. O clădire este considerată vizibilă dacă este mai înaltă decât toate clădirile aflate la stânga acesteia.

Pentru o intersecție de N străzi ajutați-l pe Gigel să determine gradul de vizibilitate al fiecărei străzi și să precizeze care este strada cu grad maxim de vizibilitate.

Date de intrare

Fișierul de intrare **strazi.in** va conține pe prima linie numărul de străzi N , urmat de N linii corespunzătoare datelor celor N străzi. Pe linia $K + 1$ ($K \geq 1$) din fișier sunt precizate: numărul de clădiri, apoi înălțimile acestor clădiri în ordinea parcurgerii străzii cu numărul K , de la stânga la dreapta.

Date de ieșire

Fișierul de ieșire **strazi.out** va conține pe primele N linii gradul de vizibilitate al străzilor, în ordinea crescătoare a indicilor de stradă (linia X va conține gradul de vizibilitate al străzii X , $1 \leq X \leq N$). Ultima linie din fișier va conține indicele străzii cu grad maxim de vizibilitate. Dacă sunt mai multe străzi cu grad maxim de vizibilitate atunci va fi afișată strada cu indicele de ordine cel mai mic.

Restricții și precizări

- $1 \leq N < 100$
- Numărul de clădiri de pe o stradă este mai mic decât 1000
- Înălțimea unei clădiri nu depășește 105 m.

Exemplu

strazi.in	strazi.out
4	0.600
5 5.1 7.2 2.0 6.9 8.3	0.333
3 6.5 4.2 3.1	1.000
2 3.4 5.0	1.000
4 1.2 2.3 3.4 4.5	3

Explicatie:

- Clădirile vizibile din partea stângă a primei străzi sunt cele cu înălțimile 5.1, 7.2 și 8.3. Gradul de vizibilitate este $3/5=0.600$
- Pentru strada 2 este vizibilă o singură clădire, prima. Gradul de vizibilitate este $1/3=0.333$
- Pentru strada 3 sunt vizibile cele 2 clădiri. Gradul de vizibilitate este $2/2=1.000$
- Pentru strada 4 sunt vizibile toate clădirile. Gradul de vizibilitate este $4/4=1.000$
- Străzile 3 și 4 are gradul de vizibilitate maxim dar va fi afișată strada cu indicele cel mai mic

Timp de execuție: **0.25 s**; Memorie disponibilă: **64 MB**; Stivă disponibilă: **2 MB**;

Relevant

Concursul interjudetean Memorial „Stefan Dârtu”
decembrie 2018
Vatra Dornei

Clasa a 8-a, Problema 1

Enunțul problemei

Considerând un șir de valori binare, numim secvență relevantă un set de elemente aflate pe poziții consecutive în șir care are proprietatea că numărul valorilor egale cu 1 este strict mai mare decât numărul valorilor de 0. De exemplu, în șirul 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 0, 0 o secvență relevantă este 0, 1, 1 și o alta, de lungime mai mare, este 0, 1, 1, 0, 1, 1, 1. Secvența relevantă maximală este secvența relevantă de lungime maximă. În șirul din exemplu secvența relevantă maximală este 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 0 (adică întreg șirul, fără ultimul zero).

Dat fiind un șir de valori binare, să se determine lungimea unei secvențe relevante maxime precum și numărul acestor secvențe.

Date de intrare

Fișierul de intrare **relevant.in** va conține pe prima linie un număr natural V , iar pe a doua, șirul de valori binare fără spații.

Date de ieșire

Fișierul de ieșire **relevant.out** va conține

- varianta 1: dacă $V = 1$, atunci pe prima linie a fișierului de ieșire va fi un singur număr natural reprezentând lungimea unei secvențe relevante maxime
- varianta 2: dacă $V = 2$, atunci pe prima linie a fișierului de ieșire va fi un singur număr natural reprezentând numărul secvențelor relevante maxime

Restricții și precizări

- V poate fi 1 sau 2
- Lungimea șirului de valori binare este de cel mult 300000
- Pentru toate testele șirul binar va conține cel puțin o valoare de 1
- Pentru 60% din punctaj, $V = 1$

Exemplu

relevant.in	relevant.out
1 100011011100	11 Explicația: Secvența relevantă maximală este 10001101110 și are lungimea 11.
2 100011011100	1 Explicația: Secvența relevantă maximală este 10001101110 și are lungimea 11. Este o singură secvență relevantă maximală.
1 0000110000111	9 Explicația: Secvența relevantă maximală are lungime 9; aceasta este 110000111.

Timp de execuție: **0.05 s**; Memorie disponibilă: **64 MB**; Stivă disponibilă: **2 MB**;

Clasa a 8-a, Problema 2

Enunțul problemei

Ferma lui Algo arată ca o gospodărie mare, în care sunt crescute multe animale și sunt cultivate pe suprafețe întinse legume, cereale și pomi fructiferi. În acest an, pomii au fost atacați de insecte care le distrug fructele. Algo a căutat o soluție pentru îndepărtarea insectelor, dar nu a găsit una eficientă. A observat însă că insectele sunt sensibile la fum. Așa că a construit un dispozitiv alcătuit din două țevi, cu care poate să tragă în același timp, pe aceeași direcție în ambele sensuri, două baloane speciale umplute cu fum. La fiecare acționare a dispozitivului sunt lansate cu aceeași viteză cele două baloane, care se sparg și împrăștie fumul la contactul cu copacul.

Deoarece baloanele speciale și tehnologia lui de a le umple cu fum sunt costisitoare, Algo își propune să alunge dăunătorii folosind cât mai eficient resursele. Astfel el vrea să folosească cât mai puține baloane și caută posibilitatea de a amplasa dispozitivul într-un punct din fermă care să îi permită trageri eficiente, adică să poată trage în toți pomii din fermă și la fiecare tragere să atingă doi pomi în același timp. Determinați dacă este posibil să găsească acest punct.

Date de intrare

Fișierul de intrare **insecte.in** va conține pe prima linie un număr natural N , reprezentând numărul de pomi de la fermă. Pe următoarele N linii se află coordonatele pomilor în forma abscisă ordonată, separate prin spațiu.

Date de ieșire

Fișierul de ieșire **insecte.out** va conține pe prima linie numărul 1 dacă există soluție sau 0 dacă nu există o soluție. În cazul în care există soluție se vor scrie în continuare, pe aceeași linie, separate prin spațiu, coordonatele punctului determinat, în ordinea abscisă ordonată, ca numere reale cu **3 zecimale exacte** .

Restricții și precizări

- $1 \leq N \leq 1000$
- pomii sunt considerați puncte în plan, fără să ținem cont de grosimea și înălțimea lor
- coordonatele punctelor sunt numere întregi din intervalul $[-10000, 10000]$
- dispozitivul nu poate fi amplasat într-un copac
- balonul explodează doar la contactul cu pomul

Exemplu

insecte.in	insecte.out
4 10 0 10 10 0 10 0 0	1 5.000 5.000 Explicația: Există un punct în care poate fi amplasat dispozitivul. Acesta se află la coordonatele (5, 5).

Timp de execuție: **0.05 s**; Memorie disponibilă: **64 MB**; Stivă disponibilă: **2 MB**;

Nrdiv

Concursul interjudetean Memorial „Stefan Dârtu”
decembrie 2018
Vatra Dornei

Clasa a 9-a, Problema 1

Enunțul problemei

Se știe că fiecare număr mai mare strict decât 1 are cel puțin 2 divizori (pe 1 și pe el însuși).

Se consideră N perechi de câte două numere naturale A_i și B_i . Pentru fiecare pereche i , să se afișeze suma numărului de divizori ale tuturor numerelor din intervalul $[A_i, B_i]$.

Date de intrare

Fisierul de intrare **nrdiv.in** conține pe prima linie numărul N . Pe următoarele N linii se află valorile celor N perechi de numere. Pe linia $i + 1$ se află cele două numere naturale ale seriei i , A_i și B_i separate prin câte un spațiu.

Date de ieșire

Fisierul de ieșire **nrdiv.out** conține N linii. Pe linia i aflându-se un singur număr natural, reprezentând suma intervalului i .

Restricții și precizări

- $1 \leq N \leq 10^5$
- $2 \leq A < B \leq 10^6$
- Intervalul $[A, B]$ este închis, deci se vor lua în considerare și valorile A și B
- Pentru 20% din teste $N \leq 10$ și $B \leq 500$
- Pentru alte 30% din teste $B \leq 10^5$

Exemplu

nrdiv.in	nrdiv.out
2	4
2 3	13
3 7	

Explicația: În prima serie, 2 și 3 au câte 2 divizori fiecare. $2+2=4$. În a 2-a serie, 3, 5 și 7 au câte 2 divizori fiecare, 4 are 3 divizori, iar 6 are 4 divizori. $3*2+3+4=13$.

Timp de execuție: **0.2 s**; Memorie disponibilă: **64 MB**; Stivă disponibilă: **2 MB**;

Clasa a 9-a, Problema 2

Enunțul problemei

O proprietate interesantă a fracțiilor ireductibile este că orice fracție se poate obține după următoarele reguli:

- pe primul nivel se află fracția $1/1$
- pe al doilea nivel, în stânga fracției $1/1$ de pe primul nivel, plasăm fracția $1/2$ iar în dreapta ei fracția $2/1$
nivelul 1: $1/1$
nivelul 2: $1/2$ $2/1$
- pe fiecare nivel k se plasează sub fiecare fracție i/j de pe nivelul de deasupra, fracția $i/(i+j)$ în stânga, iar fracția $(i+j)/j$ în dreapta.
nivelul 1: $1/1$
nivelul 2: $1/2$ $2/1$
nivelul 3: $1/3$ $3/2$ $2/3$ $3/1$

Dându-se o fracție oarecare prin numărătorul și numitorul său, determinați numărul nivelului pe care se află fracția sau o fracție echivalentă (având aceeași valoare) cu aceasta.

Date de intrare

Fișierul de intrare **fr.in** conține pe prima linie două numere naturale nenule N și M , separate printr-un spațiu, reprezentând numărătorul și numitorul unei fracții.

Date de ieșire

Fișierul de ieșire **fr.out** va conține o singură linie, pe care va fi scris număr natural nenul, reprezentând numărul nivelului care corespunde fracției.

Restricții și precizări

- $1 \leq N, M \leq 2 * 10^9$

Exemplu

fr.in	fr.out
13 8	6
12 8	3

Timpe de execuție: **0.05 s**; Memorie disponibilă: **64 MB**; Stivă disponibilă: **2 MB**;

Pătrate

Concursul interjudetean Memorial „Stefan Dârtu”
decembrie 2018
Vatra Dornei

Clasa a 9-a, Problema 3

Enunțul problemei

În timpul lor liber, Bert și Rina se joacă împreună. Ei joacă pe un caroiaj de $N \times M$ jocul cu pătrate: la fiecare moment al jocului unul dintre ei așează un pătrat de dimensiune $P \times P$ într-o zonă neocupată a tablei de joc. Pentru că Rina se plictisește repede și vrea ca un joc să se termine cât mai repede, ea și Bert vor juca optim, astfel încât la sfârșitul jocului pe tablă vor fi cât mai puține piese.

Dându-se două numere naturale, N și M , ajutați-i pe Bert și Rina să afle care este dimensiunea B a celei mai mari și dimensiunea L a celei mai mici piese care se va afla pe tablă la sfârșitul jocului. De asemenea, pentru puncte bonus semnificative, ei vor să afle și câte piese de fiecare dimensiune P au nevoie pentru a umple întreaga tablă de joc. Rețineți că e esențial ca Rina să nu se plictisească de-a lungul jocului și că tabla trebuie umplută cu cât mai puține piese.

Date de intrare

Fisierul de intrare **patrate.in** conține pe prima linie numerele N , M și R reprezentând dimensiunile tablei de joc respectiv tipul cererii lui Bert și Rina:

- Dacă $R = 0$, Bert și Rina vor să știe doar care este dimensiunea B a celei mai mari și dimensiunea L a celei mai mici piese care se va afla pe tablă la sfârșitul jocului
- Dacă $R = 1$, Bert și Rina vor să știe câte piese de fiecare dimensiune P au nevoie pentru a umple întreaga tablă de joc

Date de ieșire

- Dacă $R = 0$, fisierul de intrare **patrate.out** va conține pe prima linie numerele B și L
- Dacă $R = 1$, fisierul de intrare **patrate.out** va conține perechi, câte una pe linie, de forma (*dimensiunepatrat, frecventa*), ordonate descrescător după primul câmp al perechii, "dimensiune patrat".

Restricții și precizări

- $1 \leq N, M \leq 10^9$
- $R \in \{0, 1\}$
- $1 \leq P \leq \min(N, M)$
- Se garantează pentru orice N și M că există o soluție și că aceasta e unică
- Pentru 20% din test, $R = 0$

Exemplu

patrate.in	patrate.out
2 3 0	2 1
2 3 1	2 1 1 2

Timp de execuție: **0.1 s**; Memorie disponibilă: **64 MB**; Stivă disponibilă: **2 MB**;

Clasa a 10-a, Problema 1

Enunțul problemei

Se consideră un șir ce conține numere întregi. Șirul poate fi organizat într-un tablou bidimensional parcurs în spirală doar dacă elementele sale formează o progresie geometrică. Tipic pentru progresiile geometrice este faptul ca primul termen este dat, iar raportul dintre oricare doi termeni consecutivi este constant; acest raport se numește rația progresiei.

$b_k = b_{k-1} * q = b_1 * q^{k-1}$, $k > 2$. Exemplu : 3, 30, 300, 3000, 30000, ... este o progresie geometrică cu primul termen $b_1 = 3$ și rația $q = 10$.

Să se stabilească dacă șirul dat formează sau nu o progresie geometrică. În cazul în care răspunsul este afirmativ, să se afișeze matricea în spirală, în caz contrar să se afișeze valorile primelor două elemente din șir ale căror valori nu respectă condiția dată.

Date de intrare

Fișierul de intrare **spg.in** conține pe prima linie două numere naturale N și M reprezentând numărul liniilor și al coloanelor. Pe următoarea linie se află elementele șirului despărțite printr-un spațiu.

Date de ieșire

Fișierul de ieșire **spg.out** va conține:

- o matricea parcursa în spirală (ce conține N linii și M coloane) dacă șirul formează o progresie geometrică
- o două numere, reprezentând primele două numere din tablou care nu îndeplinesc proprietatea de progresie geometrică dacă șirul nu formează o progresie geometrică

Restricții și precizări

- o $2 \leq N, M \leq 20$
- o Elementele potențialei progresii geometrice sunt numere naturale cuprinse între 0 și 10^9

Exemplu

spg.in	spg.out
3 3 2 4 8 16 32 64 128 256 512	2 4 8 256 512 16 128 64 32
3 2 5 25 125 625 1255 20000	625 1255

Timpe de execuție: 0.05 s; Memorie disponibilă: 64 MB; Stivă disponibilă: 2 MB;

Portal

Concursul interjudetean Memorial „Stefan Dârtu”
decembrie 2018
Vatra Dornei

Clasa a 10-a, Problema 2

Enunțul problemei

John se află într-un labirint 2D de dimensiune $N \times N$. Celulele labirintului pot fi:

- celula de intrare în labirint, de unde începe jocul
- celule libere, care pot fi traversate cu cost 1
- celule zid, pe unde nu se poate trece
- celule portal, prin care se poate teleporta din celula A în celula B cu cost 1; dintr-o celulă portal se poate trece în altă celulă portal doar dacă acestea sunt perechi
- celula de ieșire din labirint, unde se încheie jocul

O celulă din labirint poate să aibă un singur rol dintre cele de mai sus. În momentul când John ajunge pe o celulă de tip portal e teleportat instant pe partea cealaltă a portalului. Ajuns acolo, John e nevoit să părăsească acea celulă, ulterior putând să facă și drumul înapoi prin portal dacă e nevoie. John se întreabă care e cel mai scurt drum prin labirint de la poziția lui inițială până la ieșire.

Date de intrare

Fișierul de intrare **portal.in** conține pe prima linie trei numere naturale N , Z și P reprezentând dimensiunea labirintului, numărul de ziduri și respectiv numărul de perechi de portaluri. Pe următoarea linie vor fi două numere reprezentând coordonatele de un John începe. A treia linie conține două numere reprezentând coordonatele punctului de final. Următoarele Z linii vor conține perechi de câte două numere reprezentând coordonatele celulelor blocate. Ultimele P linii, conțin câte 4 numere naturale, reprezentând coordonatele unei perechi de portaluri.

Date de ieșire

Fișierul de ieșire **portal.out** va conține lungimea celui mai scurt drum al lui John până la celula de ieșire. Dacă un astfel de drum nu e posibil, se va afișa -1.

Restricții și precizări

- $2 \leq N \leq 100$
- $0 \leq Z \leq N^2 - 2$
- $0 \leq P \leq N^2 - 2$
- John poate trece prin celule doar în direcțiile nord, sud, est, vest
- John nu poate părăsi labirintul în timpul jocului

Exemplu

portal.in	portal.out	Explicatie
7 5 2	5	J - 1 x - - -
1 1		x x x x - - -
6 6		- - - - - - -
1 4		- - - - - - -
2 1		- - - - - 2
2 2		1 2 - - - E -
2 3		- - - - - - -
2 4		
1 3 6 1		
5 7 6 2		

Timp de execuție: 0.05 s; Memorie disponibilă: 64 MB; Stivă disponibilă: 2 MB;

Clasa a 10-a, Problema 3

Enunțul problemei

Mai sunt cateva saptamani si vine Craciunul. Ajuns intr-un magazin de jucarii Robert il roaga pe tatal sau sa-i cumpere cea mai frumoasa masina cu telecomanda. Tatal sau ii spune ca nu a fost cuminte in timpul anului si nu merita aceasta jucarie, insa dupa dispute intense acesta hotaraste sa-i mai acorde o sansa doar daca va rezolva urmatoarea problema: Avand un string S , putem sa obtinem un palindrom din acest sir stergand un singur caracter ? Robert nu se prea descurca la algoritmica asa ca va roaga foarte mult sa-i rezolvati problema pentru a se putea juca cu masina cu telecomanda. In schimbul acestei masini el va va recompensa cu 100 puncte.

Find dat un string S , se poate obtine un palindrom din sirul initial stergand doar un singur caracter?

Date de intrare

Fișierul de intrare **constructpalindrom.in** conține pe prima linie o valoare T reprezentand numarul de teste. Pe urmatoarele T linii vom avea cate un string reprezentand intrebarea adresata lui Robert de catre tatal sau.

Date de ieșire

Fișierul de ieșire **constructpalindrom.out** va contine T linii cu raspunsul „YES” daca se poate obtine un palindrom stergand un singur caracter si „NO” daca nu se poate obtine.

Restricții și precizări

- $T \leq 100$
- Dimensiunea stringului ≤ 100000
- Pentru 10% din punctaj dimensiunea stringului ≤ 1000
- String-ul contine caractere de la ,a' la ,z'.
- Dimensiunea string-ului dupa stergerea unui caracter va fi mai mica decat a fost inainte

Exemplu

constructpalindrom.in	constructpalindrom.out
4	YES
aaa	NO
abc	YES
abdbca	YES
abba	

Explicatie:

Pentru I exemplu(aaa): Putem sterge orice 'a', string-ul rezultat este 'aa' care este palindrom.

Pentru al II-lea exemplu(abc): Nu este posibil sa eliminam exact un singur caracter si sa obtinem un

palindrom. Pentru al III-lea exemplu(abdbca): Stergem caracterul 'c',string-ul rezultat este 'abdba' care este palindrom. Pentru exemplul IV(abba): Stergem 'b', obtinem 'aba' care este palindrom.

Timp de execuție: **0.2 s**; Memorie disponibilă: **64 MB**; Stivă disponibilă: **2 MB**;

Split

Concursul interjudetean Memorialul "Stefan Dartu"

Decembrie 2018, Vatra Dornei

Clasa a 11-a Problema 1

Enuntul problemei

O subsecventa a unui sir reprezinta o submultime de litere din sirul respectiv aflate pe pozitii consecutive.

Se dau doua siruri $S1$ si $S2$ formate din litere mici ale alfabetului englez. Se mai dau N operatii de forma : **start, count** cu semnificatia ca din sirul **S1** se sterg **count** elemente incepand cu pozitia **start**. Se garanteaza ca plecand de la pozitia **start** un numar de **count** elemente nu se va depasi sfarsitul sirului. Dupa fiecare operatie se extrag intru-un vector **V** pozitiile de inceput in care sirul $S2$ apare ca subsecventa a $S1$. Vi se cere sa determinati lungimea maxima a unei secvente din **V** pentru care diferenta intre oricare 2 elemente consecutive este cel mult egala cu lungimea sirului **S2**.

Date de intrare

Pe prima linie a fisierului **split.in** se afla sirul **S1**, iar pe a doua linie sirul **S2**. Pe a 3-a linie a fisierului se afla numarul n de operatii care efectueaza. Pe urmatoarele n linii se vor afla n perechi de forma **start, count** cu semnificatia din enunt.

Date de iesire

Fisierul **split.out** va contine n linii reprezentand raspunsul cerut dupa fiecare operatie.

Restrictii si precizari

- $1 \leq s1 \leq 100000$
- $1 \leq s2 \leq s1$
- $1 \leq n \leq 10$
- $1 \leq start \leq s1$
- indicii din sirul $s1$ si $s2$ sunt numerotati de la 0
- se garanteaza ca prin stergere sirul $s1$ nu ramane gol
- operatiile se efectueaza asupra sirului initial, nu asupra celui de la pasul anterior

Exemplu

split.in	split.out
babababaabb	3
bab	3
2	
10 1	
9 2	

Timpe de executie : **3 s** ; Memorie disponibila : **64 MB**; Stiva disponibila : **2 MB**;

Taxiuri

Concursul interjudetean Memorial „Stefan Dârtu”
decembrie 2018
Vatra Dornei

Clasa a 11-a, Problema 2

Enunțul problemei

În orașul Dorna Călimanilor, care are M străzi bidirecționale și N intersecții, au sosit o multitudine de turiști grăbiți să meargă în localurile lor favorite, neavând răbdare să se întâlnească cu concurenții lor de mâine. Asadar, ei comandă Q taxiuri însă și taximetristii sunt agitați, pentru că ei nu știu care drum să îl aleagă în așa fel încât să ajungă într-un timp cât mai scurt la destinație. Misiunea voastră este să-i ajutați să găsească aceste drumuri optime!

Date de intrare

Pe prima linie a fișierului **taxiuri.in** se vor afla numerele N și M , care reprezintă numărul de intersecții, respectiv numărul de străzi ale orașului. Pe următoarele M linii se vor afla triplete de numere x , y și t , reprezentând capetele fiecărei străzi (câte o intersecție), precum și timpul parcurgerii acesteia. Pe linia $M+1$ se va afla Q , numărul de interogări (query-uri), urmând ca pe liniile $M+2$, $M+3$, ..., $M+Q+1$ să existe perechile de numere x_k și y_k , ($1 \leq k \leq Q$), reprezentând capetele fiecărui drum de durată minimă ce trebuie aflat.

Date de ieșire

Pe fiecare dintre cele Q linii din fișierul **taxiuri.out** va fi afișat drumul de durată minimă dintre capetele x_k și y_k . În caz că nu există drum de la x_k la y_k , se va afișa mesajul „nu exista” (fără ghilimele).

Restricții și precizări

- $1 \leq N \leq 300$
- $1 \leq M \leq N * (N - 1) / 2$
- $0 \leq Q \leq N * (N - 1) / 2$
- $1 \leq x, y, x_k, y_k \leq N$
- $1 \leq t \leq 3 * 10^6$
- Orice drum optim este valid.

Exemplu

taxiuri.in	taxiuri.out	explicatii
6 8	1 6	Pentru al patrulea query, mergând pe drumurile (6,1), (1,5) și (5,2), obținem un timp total de parcurgere de $477 + 82 + 443 = 1002$, ceea ce este mai mic decât dacă am merge pe drumul (6,2), durând 2699 unități de timp.
4 2 1515	3 1	
5 1 82	5 2	
6 1 477	6 1 5 2	
6 2 2699	3 1 5 2	
4 6 1649		
5 4 193		
5 2 443		
3 1 223		
5		
1 6		
3 1		
5 2		
6 2		
3 2		

Timp de execuție: **0.25 s**; Memorie disponibilă: **64 MB**; Stivă disponibilă: **2 MB**;

Ciuperci

Concursul interjudetean Memorial „Stefan Dârtu”
decembrie 2018
Vatra Dornei

Clasa a 11-a, Problema 3

Enunțul problemei

Pentru a-si putea pune ciuperci la murat pentru iarnă, Ștefan s-a gândit să meargă la pădure. El găsește o multime de N ciuperci așezate în linie, fiecare având o greutate g_i ($1 \leq i \leq N$), așezate la o distanță egală de 1 metru. Un lucru de menționat este că unele ciuperci sunt otrăvite, iar Ștefan nu le va culege pe acestea. Așadar, el începe cu a culege o ciupercă oarecare k (care nu e otrăvită) și continuă prin a le alege pe următoarele după o regulă specială: următoarea ciupercă culeasă după cea curentă trebuie să fie strict mai grea. Totuși, Ștefan este o persoană leneșă, el alegând să meargă cel mult L metri între oricare două ciuperci culese consecutiv.

Numărați câte ciuperci se pot strânge maxim conform regulilor descrise mai sus și în câte moduri distincte acestea pot fi culese.

Date de intrare

Fișierul de intrare **ciuperci.in** conține pe prima linie două numere naturale N și L , urmate de N linii ce conțin două valori: g_i și ot_i . G_i reprezintă greutatea ciupercii i , iar ot_i indică dacă ciupercă i e otrăvită sau nu (1 dacă e otrăvită, 0 dacă e comestibilă).

Date de ieșire

Fișierul de ieșire **ciuperci.out** va conține două numere separate printr-un spațiu reprezentând numărul maxim de ciuperci ce pot fi culese și în câte moduri distincte se poate face aceasta. În cazul în care Ștefan nu poate culege nici o ciupercă, se va afișa mesajul "nu există".

Restricții și precizări

- $1 \leq N \leq 5000$
- $1 \leq L \leq N$
- $1 \leq g_i \leq 10^9$
- Se acordă 20% din punctaj pe fiecare test pentru afișarea primului număr, și 80% din punctaj pentru celălalt.

Exemplu

ciuperci.in	ciuperci.out
7 2	3 2
4 0	
5 0	Explicatie: Ștefan poate să culeagă prima
6 0	ciupercă, cu $g_1=4$, precum și pe a doua și a
1 0	treia, dar de la a treia nu poate ajunge la a
2 1	șaptea, deoarece ar trebui să meargă 4 metri
3 0	înainte, iar el nu poate decât maxim 2. Și chiar
9 0	dacă ar pleca de la a patra ciupercă (valoare 1),
	nu o poate lua pe următoarea, pentru că este
	otrăvită, așa că sare peste ea.

Timp de execuție: **0.1 s**; Memorie disponibilă: **64 MB**; Stivă disponibilă: **2 MB**;

K-numere

Concursul interjudetean Memorial „Stefan Dârtu”
decembrie 2018
Vatra Dornei

Clasa a 12-a, Problema 1

Enunțul problemei

Andrei este unul dintre cei mai gălăgioși elevi din clasa lui când vine vorba de informatică, iar din acest motiv, într-o zi, profesoara i-a dat următoarea temă: având un șir de N numere naturale și o valoare K , Andrei trebuie să găsească valoarea din șir ce apare o singură dată, știind că orice altă valoare din șir apare exact de K ori. Pentru că nu a fost atent la orele de informatică și șirul primit poate avea foarte multe valori, Andrei vă roagă să scrieți un program care să îi rezolve problema.

Dându-se numerele N și K , și cele N numere, ajutați-l pe Andrei să găsească valoarea care apare o singură dată în șirul de numere.

Date de intrare

Fisierul de intrare **knumere.in** conține pe prima linie numerele N și K cu specificațiile din enunț. Pe a doua linie se află N numere ce reprezintă șirul de numere primit de Andrei.

Date de ieșire

Fisierul de ieșire **knumere.out** va conține o linie cu numărul ce apare în șirul de numere o singură dată.

Restricții și precizări

- $200000 \leq N \leq 300000$
- $2 \leq K \leq 15$
- Pentru 20% din teste $K = 2$

Exemplu

knumere.in	knumere.out
10 3 1 3 5 7 5 1 3 1 5 3	7

Timp de execuție: **0.6 s**; Memorie disponibilă: **0.5 MB**; Stivă disponibilă: **0.5 MB**;

Bitcoin

Concursul interjudetean Memorial „Stefan Dârtu”
decembrie 2018
Vatra Dornei

Clasa a 12-a, Problema 2

Enunțul problemei

John are pentru voi o problema foarte simpla. Avand o lista de monede bitcoin, John va cere doua chestii:

- sa se adauge bitcoinului de pe pozitia i o valoare x
- sa se calculeze suma bitcoinurilor dintre doua pozitii.

Date de intrare

Pe prima linie a fisierului **bitcoin.in** se afla un numar N reprezentand numarul de monede si M numarul de operatii descris de John. Urmatoarele M linii descriu operatiile descrise de John dupa cum urmeaza:

- 1 a b - valorii bitcoinului de pe pozitia a se adauga o valoare b
- 2 x y - se determina suma bitcoinilor din intervalul $[x,y]$

Date de ieșire

Fisierul de iesire **bitcoin.out** va contine X valori reprezentand raspunsul la a doua intrebare a lui John.

Restricții și precizări

- $1 \leq N \leq 10^5$
- $1 \leq a \leq b \leq N$
- $1 \leq A_i \leq 10000$, pentru orice i , $1 \leq i \leq N$

Exemplu

bitcoin.in	bitcoin.out
10 5	90
10 20 30 40 50 13 15 19 21 62	130
1 1 10	
1 2 20	
2 1 3	
2 1 4	
1 4 100	

Timp de execuție: **0.16 s**; Memorie disponibilă: **64 MB**; Stivă disponibilă: **2 MB**;

Sugestie

Concursul interjudetean Memorial „Stefan Dârtu”
decembrie 2018
Vatra Dornei

Clasa a 12-a, Problema 3

Enunțul problemei

John e leneș, scrie multe mesaje și are un dicționar cu perechi (cuvânt, popularitate).

Dându-se Q secvențe de litere Q_i , John se întreabă pentru fiecare care e cel mai popular cuvânt care începe cu Q_i .

Date de intrare

Pe prima linie a fișierului **sugestie.in**:

* N , dimensiunea dicționarului

* Q , numărul de interogări

Pe următoarele N linii:

* N elemente din dicționar (cuvânt, popularitate), sir de caractere, respectiv număr întreg

Pe următoarele Q linii:

* Q interogări, secvențe de litere, siruri de caractere

Date de ieșire

Fisierul de ieșire **sugestie.out** va conține Q siruri de caractere, câte unul pe linie. Dacă pentru o secvență de litere Q_i nu există un cuvânt în dicționar cu un prefix comun de cel puțin un caracter, se va afișa -1 pe linia corespunzătoare lui Q_i .

Restricții și precizări

- $1 \leq N \leq 10^4$
- $1 \leq Q \leq 200$
- Sirurile de caractere din fișierul de intrare conțin doar caractere mici ale alfabetului englez (26 de caractere, între 'a' și 'z')
- Pentru 70% din teste $N = 10^4$ și John folosește un dicționar sofisticat: în medie, pentru fiecare cuvânt din dicționar există în dicționar cel puțin un alt cuvânt cu un sufix S mai lung decât 5, cu prefix de lungime maximală (de exemplu: "cine", "cinematograf", "cititor"); John vă recomandă să fiți atenți la limita de memorie.
- Pentru orice secvență de litere Q_i interogată se garantează existența unui cuvânt în dicționar care să conțină la începutul lui Q_i

Exemplu

sugestie.in	sugestie.out
4 3	banana
banana 7	bandana
bandana 3	-1
ana 6	
banda 2	
b	
band	
qana	

Timp de execuție: **2 s**; Memorie disponibilă: **4 MB**; Stivă disponibilă: **2 MB**;