

PROBLEMA 1

100 puncte

FLORI

Maria vizitează o grădină botanică în care există o parcelă de teren în formă dreptunghiulară $n \times m$ împărțită în celule de dimensiune 1×1 . Unele celule sunt părăsite iar altele conțin flori codificate cu litere de la A..D. După ce a vizitat grădina cu atenție, Maria a constatat ca florile din fiecare celulă au fost plantate respectând regulile următoare:

- Florile din celulele cultivate sunt plantate în ordinea $(1,1) \square \dots (1,m) \square (2,1) \dots (2,m) \square \dots (n,1) \square (n,m)$ și celulele părăsite sunt ignorate.
- Florile sunt plantate în fiecare celulă una după alta ciclic.
- Floarea codificată cu A este plantată în prima celulă, floarea codificată cu B este plantată în celula doi și așa mai departe.

Cerință:

Ajutați-o pe Maria trebuie să răspundă la p întrebări de forma: *Ce floare există într-o celulă dată?*

Date de intrare:

Prima linie a fisierului *flori.in* conține 4 numere naturale n, m, k, p unde n reprezintă lungimea parcelei, m este lățimea parcelei, k numărul de celule părăsite și p numărul de întrebări. Următoarele k linii conțin câte două numere a, b care reprezintă coordonatele celulei părăsite. Pe următoarele p linii sunt două numere i, j care reprezintă o întrebare despre conținutul celulei (i,j) .

Date de ieșire:

Fisierul de ieșire *flori.out* conține pe linia x răspunsul la întrebarea $x, 1 \leq x \leq p$. Dacă celula este părăsită atunci răspunsul este codificat cu litera **P**

Restricții:

$$1 \leq n \leq 10^4, 1 \leq m \leq 10^4, 1 \leq k \leq 10^3, 1 \leq p \leq 10^3, 1 \leq a \leq n, 1 \leq b \leq m, 1 \leq i \leq n, 1 \leq j \leq m$$

Exemplu:

flori.in	flori.out	Explicații:																				
4 5 5 6 4 3 1 3 3 3 2 5 3 2 1 3 1 4 2 3 2 4 1 1 1 1	P C C D A A	<table border="1"> <tr> <td>A</td> <td>B</td> <td style="background-color: black;"></td> <td>C</td> <td>D</td> </tr> <tr> <td>A</td> <td>B</td> <td>C</td> <td>D</td> <td style="background-color: black;"></td> </tr> <tr> <td>A</td> <td style="background-color: black;"></td> <td style="background-color: black;"></td> <td>B</td> <td>C</td> </tr> <tr> <td>D</td> <td>A</td> <td style="background-color: black;"></td> <td>B</td> <td>C</td> </tr> </table>	A	B		C	D	A	B	C	D		A			B	C	D	A		B	C
A	B		C	D																		
A	B	C	D																			
A			B	C																		
D	A		B	C																		

Timp maxim de execuție: 0.06 secunde/test.

Memorie totală disponibilă 1 MB, din care 1 MB pentru stivă

Dimensiunea maximă a sursei: 5 KB

PROBLEMA 2

100 puncte

SIRUL

Un șir de numere naturale își poate mări lungimea inserând noi elemente între cele existente. Astfel, între orice două elemente vecine și **neegale** ale șirului se poate insera media lor aritmetică, numai dacă aceasta este la rândul ei un număr natural. Între orice două elemente vecine și neegale ale noului șir (obținut după una sau mai multe inserări) se poate insera din nou media lor aritmetică, dacă aceasta este un număr natural. Când nicio inserare nu mai este posibilă, spunem că șirul și-a atins lungimea maximă.

Cerință:

Pentru două valori N și M , un șir de N valori, V_1, V_2, \dots, V_N și un șir de M valori X_1, X_2, \dots, X_M se cere să se calculeze câte dintre numerele X_1, X_2, \dots, X_M vor aparține șirului V după ce acesta își atinge lungimea sa maximă.

Date de intrare:

Din fișierul *sirul.in* se citesc numerele naturale N și M de pe prima linie, separate printr-un spațiu, apoi N numere naturale V_1, V_2, \dots, V_N despărțite prin câte un spațiu, de pe linia a doua a fișierului și în final M numere naturale X_1, X_2, \dots, X_M despărțite prin câte un spațiu, de pe linia a treia a fișierului.

Date de ieșire:

În fișierul *sirul.out* se va scrie un număr K , reprezentând numărul de valori alese din șirul X care vor aparține șirului V după ce acesta își atinge lungimea sa maximă.

Restricții și precizări:

- $1 \leq N, M \leq 1000000$
- $0 \leq V_i \leq 1000000$, pentru orice i între 1 și N
- $0 \leq X_i \leq 1000000$, pentru orice i între 1 și M
- Pentru 50% din teste, toate valorile sunt mai mici sau egale cu 1000
- Elementele șirului V nu sunt neapărat distincte; la fel, elementele șirului X nu sunt neapărat distincte
- Dacă mai multe componente din șirul X au aceeași valoare, se vor număra toate, dacă valoarea respectivă apare în șirul V final măcar **o dată**.

Exemple:

sirul.in	sirul.out	Explicații
3 4 10 28 20 22 16 28 22	3	Șirul 10 28 20 devine după două inserări 10 19 28 24 20 . Între 10 și 19 nu se mai poate insera nimic și la fel între 19 și 28. Dar între 28 și 24 se poate insera 26, iar între 24 și 20 se poate insera 22. Se obține astfel șirul 10 19 28 26 24 22 20 . Se pot face patru noi inserări obținând șirul 10 19 28 27 26 25 24 23 22 21 20 . În noul șir nu se mai poate insera niciun element, deci șirul și-a atins lungimea maximă. Sunt 3 numere în șirul X (22, 28 și 22) care se găsesc în șirul 10 19 28 27 26 25 24 23 22 21 20

șirul.in	șirul.out	Explicații
5 2 7 7 28 16 31 10 20	0	După o inserare șirul 7 7 28 16 31 devine 7 7 28 22 16 31 Se pot face două noi inserări și obținem șirul 7 7 28 25 22 19 16 31 Nu mai sunt posibile inserări, deci șirul și-a atins lungimea maximă și niciuna dintre valorile 10, 20 nu se află în șirul final.

Timp maxim de executare/test: 0.5 sec
Memorie totală: 4MB din care 2MB stiva
Dimensiunea maximă a sursei: 5KB

PROBLEMA 3

100 puncte

GAMES

La magazinul *SuperGames* sunt n copii la coadă. Copilul 1 se află la capătul cozii și copilul al n -lea se află la începutul cozii. Copilul i este nemulțumit dacă există un alt copil mai tânăr în fața lui. Nemulțumirea copilului i este dată de numărul de copii dintre el și cel mai îndepărtat copil situat în fața lui, care este mai tânăr decât copilul i .

Cerințe:

Ajutați-l pe administratorul magazinului SuperGames

1. să determine nemulțumirea fiecărui copil aflat la coadă
2. să calculeze câți copii sunt mulțumiți
3. să determine care sunt copiii a căror nemulțumire scade după eliminarea copilului q

Date de intrare:

În fișierul *games.in* se află pe prima linie numărul n reprezentând numărul de copii aflați la coadă, pe a doua linie n numere separate între ele prin câte un spațiu reprezentând vârstele copiilor iar pe a treia linie numărul q reprezentând copilul care se va elimina.

Date de ieșire:

Fișierul *games.out* va conține pe prima linie n numere reprezentând mulțumirea fiecărui copil. Dacă un copil este mulțumit se va scrie -1

Linia a doua a fișierului va conține un număr reprezentând numărul de copii mulțumiți.

Linia a treia conține numerele copiilor care indeplinesc cerința 3. Dacă nu există un astfel de copil se afișează -1

Restricții și precizări:

- $n \leq 10^5$
- vârsta unui copil este mai mică decât 100
- există copii cu aceeași vârstă

Exemplu:

games.in	games.out	Explicații
7 10 8 5 3 50 45 46 4	2 1 0 -1 1 -1 -1 3 1 2 3	Copilul 1 are nemulțumirea 2 deoarece copilul 4 este cel mai departe de el, situat în fața lui (între 10 și 3 sunt două valori). Copilul 4 este mulțumit deoarece în fața

		lui nu mai sunt copii mai tineri decât el. Dacă se elimină copilul 4 atunci copilul unu are nemulțumirea 1, copilul doi are nemulțumirea 0, copilul trei este mulțumit, ceilalți copii au aceeași nemulțumire.
--	--	--

Timp maxim de execuție: 0.1 secunde/test.

Memorie totală disponibilă 1 MB, din care 1 MB pentru stivă

Dimensiunea maximă a sursei: 5 KB

PROBLEMA 4

100 puncte

EMMA

Emma nu a fost la toate orele de matematică. Prin urmare, aceasta nu știe să numere ca ceilalți, ci doar în numere prime (2, 3, 5, etc). Astăzi, la școală, ea întâmpină un obstacol. Profesorul îi pune pe elevi să scrie T numere ca sumă de 2 numere. Cum aceasta nu știe toate numerele, vă cere ajutorul.

Cerința:

Pentru fiecare din numerele date, să se afișeze "DA" (fara ghilimele) dacă acesta se poate scrie ca suma de 2 numere prime, și "NU" altfel.

Date de intrare:

Pe prima linie a fișierului *emma.in* se află un număr T, reprezentând numărul de numere. Pe fiecare din următoarele T linii se află câte un număr N.

Date de ieșire:

Pe fiecare linie i din fișierul de ieșire *emma.out* se află "DA" sau "NU", corespunzător cu numărul de pe linia i+1 din fișierul de intrare.

Restricții și precizări:

- $T \leq 1000$
- $10 < N \leq 1.000.000$

Pentru 30% din teste, $T \leq 10$, $N \leq 100.000$

Pentru 70% din teste, $T \leq 1000$, $N \leq 100.000$

Exemplu:

emma.in	emma.out	Explicații
2	DA	$15 = 2 + 13$
15	NU	11 nu poate fi scris conform cerinței
11		

Timp maxim de execuție: 1,7 secunde/test.

Memorie totală disponibilă 3 MB, din care 2 MB pentru stivă

Dimensiunea maximă a sursei: 5 KB.