

Problema Dominew

Fișier de intrare `dominew.in`
Fișier de ieșire `dominew.out`

Pentru că se plictisește și este foarte inteligent, Radu l-a rugat pe prietenul lui, savantul Feder, să creeze o activitate care să-i pună mintea la încercare. Savantul Feder a adus N piese dreptunghiulare pe care sunt scrise numere naturale și le-a așezat pe masă în ordinea crescătoare a valorilor scrise pe ele, pe poziții consecutive, una lângă cealaltă. Apoi îi dă lui Radu, una câte una, alte M piese dreptunghiulare, pe care sunt scrise numere naturale, într-o ordine oarecare. Când Radu primește o piesă el trebuie să o așeze în șirul de pe masă pe **cea mai mică poziție posibilă**, astfel încât piesele din șir să rămână în ordine crescătoare. Evident, șirul de pe masă se modifică pe măsură ce Radu așază piesele în șir.

Cerință

Cunoscând șirul pieselor de pe masă, în ordinea în care sunt așezate, precum și cele M piese pe care le primește succesiv Radu, scrieți un program care să afișeze pentru fiecare dintre cele M piese poziția pe care aceasta este așezată în șir.

Date de intrare

Fișierul `dominew.in` conține pe prima linie numărul natural N . Pe a doua linie se află N numere naturale, în ordine crescătoare, reprezentând valorile pieselor din șirul aflat inițial pe masă. Pe a treia linie se află numărul natural M . Pe cea de-a patra linie sunt M numere naturale, reprezentând valorile pieselor pe care le primește Radu, în ordinea în care acesta le primește. Numerele scrise pe aceeași linie sunt separate prin câte un spațiu.

Date de ieșire

Fișierul de ieșire `dominew.out` va conține o singură linie pe care vor fi scrise M valori separate prin câte un spațiu, cea de a i -a valoare fiind poziția pe care este așezată în șirul de pe masă cea de a i -a piesă primită de Radu ($1 \leq i \leq M$).

Restricții

- $2 \leq N \leq 1\,000\,000$
- $1 \leq M \leq 8\,000$
- Valorile scrise pe piese sunt numere naturale $\leq 10^9$.
- Pozițiile elementelor din șirul de pe masă sunt numerotate începând cu 1.

#	Punctaj	Restricții
1	8	$M = 1$
2	14	$1 < N, M \leq 100$
3	8	$100 < N \leq 50\,000, 100 < M \leq 250$
4	24	$N \leq 100\,000, 250 < M \leq 1\,000$
5	46	$N \leq 1\,000\,000, 1000 < M \leq 8\,000$

Exemple

<code>dominew.in</code>	<code>dominew.out</code>
6 2 5 5 9 10 11 3 5 1 12	2 1 9
14 2 2 2 4 7 8 9 10 12 16 20 21 23 24 7 18 7 20 1 16 25 23	11 5 13 1 12 20 18

Explicație

Exemplul 1

Inițial pe masă se află $N = 6$ piese, în ordine crescătoare. 2 5 5 9 10 11 Radu primește $M = 3$ piese. Prima piesă are valoarea 5 și va fi așezată în șirul de pe masă pe poziția 2. Șirul va deveni 2 5 5 5 9 10 11. A doua piesă are valoarea 1, va fi așezată în șirul de pe masă pe poziția 1. Șirul va deveni 1 2 5 5 5 9 10 11. A treia piesă are valoarea 12 și va fi așezată pe poziția 9 în șirul de pe masă. Șirul va deveni 1 2 5 5 5 9 10 11 12.

Exemplul 2

După primele 3 inserări, șirul va fi 2 2 2 4 7 7 8 9 10 12 16 18 20 20 21 23 24, valoarea 20 ajungând pe poziția 13.

Problema Pix

Fișier de intrare `pix.in`
Fișier de ieșire `pix.out`

Robotul Vasile s-a angajat la un depozit de pixuri. Aici pixurile sunt ambalate în cutii. Există N tipuri de cutii; într-o cutie de tipul i ($1 \leq i \leq N$) sunt ambalate exact nr_i pixuri ($nr_1 \leq nr_2 \leq \dots \leq nr_N$). În depozit există un număr atât de mare de cutii de fiecare tip încât Vasile poate utiliza oricâte cutii dorește, de orice tip. Sarcina robotului Vasile este să livreze pixurile comandate de diferite firme de birotică. El nu știe câte pixuri va avea de livrat la următoarea comandă, dar știe că vor fi cel mult $Vmax$ pixuri. Ca urmare, pentru a fi eficient, robotul Vasile vrea să își pregătească în camera de livrare un număr minim de cutii de pixuri astfel încât să poată livra orice număr de pixuri cuprins între 1 și $Vmax$ folosind cutiile pregătite, evident, **fără a deschide cutiile**.

Cerință

Scrieți un program care citește valorile N , nr_1 , nr_2 , \dots , nr_N și $Vmax$ și determină numărul minim de cutii pe care robotul Vasile trebuie să le pregătească în camera de livrare astfel încât să poată livra orice număr de pixuri cuprins între 1 și $Vmax$, fără a deschide nicio cutie.

Date de intrare

Fișierul de intrare `pix.in` conține pe prima linie numărul natural N reprezentând numărul de tipuri de cutii. Pe a doua linie se află N numere naturale în ordine crescătoare, separate prin câte un spațiu, nr_1 , nr_2 , \dots , nr_N reprezentând numărul pixuri ambalate în fiecare tip de cutie. Pe a treia linie se află numărul natural $Vmax$ cu semnificația din enunț.

Date de ieșire

Fișierul de ieșire `pix.out` va conține o singură linie pe care va fi scris un număr natural reprezentând numărul minim de cutii pe care robotul Vasile trebuie să le pregătească în camera de livrare astfel încât să poată livra orice număr de pixuri cuprins între 1 și $Vmax$.

Restricții

- $1 \leq N \leq 100\,000$
- $1 \leq Vmax, nr_i \leq 10^{12}$, pentru $1 \leq i \leq N$
- Se garantează că pentru toate datele de test există soluție.

#	Punctaj	Restricții
1	20	$1 \leq N < 15$
2	10	$15 \leq N < 600$
3	40	$Vmax \leq 100\,000$

Exemple

pix.in	pix.out	Explicații
4 1 2 3 5 15	5	<p>Numărul minim de cutii pe care trebuie să le pregătească Vasile este 5 (o cutie de tipul 1, două de tipul 2 și două de tipul 4, numărul de pixuri din aceste cutii fiind 1, 2, 2, 5, 5)</p> <p>El poate astfel livra orice număr de pixuri între 1 și 15, o modalitate posibilă fiind:</p> <ol style="list-style-type: none">1: cutia de tip 12: o cutie de tip 23: o cutie de tip 1 și o cutie de tip 24: două cutii de tip 25: o cutie de tip 46: o cutie de tip 1 și o cutie de tip 47: o cutie de tip 2 și o cutie de tip 48: câte o cutie de tipurile 1, 2, 49: o cutie de tip 4 și două cutii de tip 210: două cutii de tip 411: două cutii de tip 4 și o cutie de tip 112: două cutii de tip 4 și o cutie de tip 213: două cutii de tip 4, o cutie de tip 2 și o cutie de tip 114: două cutii de tip 4 și două cutii de tip 215: toate cutiile <p>Aceasta nu este singura posibilitate de a alege un număr minim de cutii pentru a obține toate valorile de la 1 la 15.</p>

Problema Secvmin

Fișier de intrare `secvmin.in`
Fișier de ieșire `secvmin.out`

Fie un șir de n numere naturale v_1, v_2, \dots, v_n , unde v_i reprezintă al i -lea număr din șir. O subsecvență $[x, y]$ a șirului v (cu $1 \leq x \leq y \leq n$) conține toate elementele $v_x, v_{x+1}, \dots, v_{y-1}, v_y$.

Cerință

Fiind date două numere naturale n și k și un șir v de n numere naturale, scrieți un program care să răspundă la următoarea întrebare: câte subsecvențe conțin simultan cele mai mici k valori distincte din șir?

Date de intrare

Fișierul de intrare `secvmin.in` conține pe prima linie numerele n și k , iar pe cea de a doua linie se vor afla numerele naturale v_1, v_2, \dots, v_n , cu semnificația din enunț. Numerele scrise pe aceeași linie sunt separate prin câte un spațiu.

Date de ieșire

Fișierul de ieșire `secvmin.out` va conține o singură linie pe care va fi scris răspunsul la cerința problemei.

Restricții

- $1 \leq n \leq 1\,000\,000$
- $1 \leq k \leq 5$
- $1 \leq v_i \leq 1\,000\,000\,000$, pentru $1 \leq i \leq n$
- Se garantează că șirul va conține cel puțin k valori distincte.

#	Punctaj	Restricții
1	23	$k = 1$
2	32	$k = 2$
3	45	$3 \leq k \leq 5$

Exemple

<code>secvmin.in</code>	<code>secvmin.out</code>	Explicații
7 1 1 3 2 2 1 3 2	19	$k = 1$. Cea mai mică valoare din șir este egală cu 1. Subsecvențele care conțin valoarea 1 sunt: [1 1], [1 2], [1 3], [1 4], [1 5], [1 6], [1 7], [2 5], [2 6], [2 7], [3 5], [3 6], [3 7], [4 5], [4 6], [4 7], [5 5], [5 6], [5 7].
7 2 1 5 2 2 1 5 2	15	$k = 2$. Cea mai mică valoare din șir este egală cu 1, iar a doua cea mai mică valoare din șir este egală cu 2. Subsecvențele care conțin valorile 1 și 2 sunt: [1 3], [1 4], [1 5], [1 6], [1 7], [2 5], [2 6], [2 7], [3 5], [3 6], [3 7], [4 5], [4 6], [4 7], [5 7].