

PROBLEMA 1 Sah24**100 puncte**

În campionatul mondial de șah24 2021 există N participanți, fiecare șahist are un indice asociat i și un rating asociat $V[i]$. Campionatul este format din Q runde, iar fiecare rundă funcționează în modul următor.

- Sunt alese 2 valori, L și R
- Șahiștii cu indicele cuprins în intervalul $[L,R]$ participă în runda respectivă
- Dacă participă într-o rundă, un șahist va juca fix un meci împotriva fiecărui alt participant din acea rundă. (fiecare participant joacă $R-L$ meciuri)
- Când doi șahiști i și j joacă un meci, i va câștiga meciul doar dacă $V[i]$ divide $V[j]$ (iar j va câștiga meciul doar dacă $V[j]$ divide $V[i]$)
- Un șahist câștigă runda doar dacă a câștigat toate meciurile din runda respectivă

Cerință

Pentru fiecare din cele Q runde date, afișați numărul de câștigători ale runde respective.

Date de intrare

Pe prima linie a fișierului *sah24.in* se află numărul n . Pe a doua linie se află n numere, reprezentând valorile v_1, v_2, \dots, v_n cu semnificația din enunț.

Pe următoarea linie se află numărul Q de runde. Urmează Q linii care conțin fiecare câte 2 numere L și R .

Date de ieșire

Pe linia i din fișierul de ieșire *sah24.out* se va afla numărul de câștigători ale runde i .

Restricții și precizări

- $1 \leq N \leq 10^5$
- $1 \leq V[i] \leq 10^9$
- $1 \leq Q \leq 10^5$
- $1 \leq L < R \leq N$
- Un meci poate să aibă 0, 1 sau 2 câștigători
- În fiecare rundă pot exista mai mulți câștigători

Exemplu

sah24.in	sah24.out	Explicație
5	1	În prima rundă câștiga șahistul 1
1 3 2 4 2	0	În a doua rundă nu există câștigători
4	2	În a treia rundă câștiga șahiștii 3 și 5
1 5	1	În a patra rundă câștigă șahistul 5
2 5		
3 5		
4 5		

PROBLEMA 2 Ski**100 puncte**

Deși sunt foarte pricepuți la ski, Cătălin și David nu stau prea bine cu orientarea. Cei 2 s-au pierdut unul de altul pe părțile de ski.

Harta domeniului skiabil este sub forma unui arbore cu N noduri. Fiecare muchie este reprezentată de o pârtie și un telescaun. Astfel, de la nodul de sus la cel de jos se ajunge pe skiuri, iar urcarea se face cu telescaunul, motiv pentru care timpul de deplasare dintre 2 noduri adiacente diferă în funcție de sens.

Știind ca cei 2 sunt dispuși să facă oricâte urcări sau coborâri, găsiți timpul minim în care cei 2 se pot reîntâlni.

Din cauza lipsei lor de orientare, cei 2 ajung să se piardă de multiple ori (Q mai exact), de fiecare dată apelând la ajutorul vostru.

Cerință

Cunoscând harta părților, timpii de urcare și coborâre între nodurile adiacente și locurile unde ajung să se piardă băieții, determinați timpul minim după care se pot reîntâlni.

Date de intrare

Fișierul de intrare *ski.in* conține pe primul rând numărul de noduri N . Pe următoarele $(N-1)$ linii se găsesc câte 3 numere cu semnificația: **nodul tată** al nodului i unde i este numărul liniei din fișierul de intrare, **timpul de urcare** și **timpul de coborâre** în deplasarea dintre nodul i și tatăl acestuia.

Pe următoarea linie se află numărul Q , iar următoarele Q linii conțin câte o pereche de numere semnificând nodurile în care cei doi realizează că s-au pierdut.

Date de ieșire

Fișierul de ieșire *ski.out* conține Q linii, pe fiecare dintre acestea se va afișa timpul minim după care se pot reîntâlni băieții.

Restricții și precizări

- Pentru 30% din teste:
 - $5 \leq N \leq 1.500$
 - $1 \leq Q \leq 1.500$
- Pentru alte 40 % teste :
 - $5 \leq N \leq 100.000$
 - $1 \leq Q \leq 100.000$
- Pentru restul testelor:
 - $5 \leq N \leq 1.000.000$
 - $1 \leq Q \leq 1.000.000$
- Timpuri urcare și coborâre ≤ 1.000

Exemplu

ski.in	ski.out	Explicație
8	5	2 – 3: Pentru ca cei 2 să se întâlnească pornind din nodurile 2 și 3, fie primul urcă în nodul 1 și apoi coboară în nodul 3 (timp = 4 + 1), fie urcă amândoi în nodul 1, timpul fiind de 5 minute pentru ambele variante.
1 4 2	9	
1 5 1	11	
2 7 3	5	4 – 7: Primul urcă în nodul 2 (t1 = 7). Al doilea urcă în nodul 3, apoi în nodul 1, apoi coboară în nodul 2 (t2 = 2 + 5 + 2 = 9). Deci, cei doi se reunesc după 9 minute.
2 6 2		
3 3 3		
3 2 8		5 – 8: Primul urcă în nodul 2, apoi urcă în nodul 1, apoi coboară în nodul 3 (t1 = 6 + 4 + 1 = 11). Al doilea urcă în nodul 6, apoi urcă în nodul 3 (t2 = 6 + 3 = 9). Cei doi se reunesc după 11 minute.
6 6 7		
4		
2 3		2 – 6: Primul urcă în nodul 1, apoi coboară în nodul 3 (t1 = 4 + 1 = 5). Al doilea urcă în nodul 3 (t2 = 3). Deci, se reunesc după 5 minute.
4 7		
5 8		
2 6		

PROBLEMA 3 Marflow

100 puncte

Marflow, pasionat de matematică și blocat în casă din cauza pandemiei, se gândește la o nouă problemă pe care să o rezolve. Din cauza propriei inteligențe, acesta își da seama că a creat o problemă mult prea dificilă pentru cunoștințele sale actuale și vă roagă să îl ajutați.

Cerință

Dându-se un număr natural k și un vector de k valori notat a ($a[i]$ reprezentând a i -a valoare) spunem că un număr natural n este o soluție a problemei dacă și numai dacă fracția $\frac{p}{q}$ este un număr

întreg unde pozitiv, unde $p = n!$ și $q = \prod_{i=1}^k a[i]!$ ($\prod = \text{produsul termenilor}$)

Marflow vă roagă să afișați soluția minimă a problemei.

Date de intrare

Pe prima linie a fișierului *marflow.in* se află numărul k . Pe a doua linie se află k numere, reprezentând valorile vectorului a .

Date de ieșire

Pe linia i din fișierul de ieșire *marflow.out* se va afla soluția minimă a problemei.

Restricții și precizări

- $1 \leq k \leq 10^6$
- $1 \leq a[i] \leq 10^7$
- Se garantează că soluția este $\leq 10^{13}$

Exemplu

marflow.in	marflow.out	Explicație
2 1000 1000	2000	2000 este cel mai mic număr pentru care fracția $(2000!) \div (1000! * 1000!)$ este un număr întreg.

PROBLEMA 4 Tryhard**100 puncte**

Nilugard a descoperit un nou joc video, care are N niveluri. De fiecare dată când se apuca de un joc nou, Nilugard este „try-hard”, adică vrea să termine jocul pe dificultatea maximă. Pentru a face acest lucru, nivelurile jocului trebuie trecute unul după altul.

În cazul în care pierde la un anumit nivel, atunci jocul trebuie reluat de la primul nivel.

Fiind „try-hard”, Nilugard și-a calculat șansele de reușită pentru fiecare nivel în parte și acum se întreabă care este valoarea așteptată a numărului de niveluri prin care trebuie să treacă pentru a termina jocul.

De exemplu, pentru un joc de 4 niveluri, s-ar putea întâmpla următoarele: trece de nivelul 1, pierde la nivelul 2, trece de nivelul 1, trece de nivelul 2, trece de nivelul 3 și trece de nivelul 4, terminând jocul trecând prin 6 niveluri.

Cerință

Cunoscând numărul de niveluri N și pentru fiecare nivel probabilitatea P_i de a trece nivelul respectiv, determinați valoarea medie a numărului de niveluri pe care trebuie să le joace Nilugard pentru a termina jocul la dificultatea maximă.

Probabilitățile sunt exprimate ca procente și sunt numere întregi (de exemplu $P_i = 25$ înseamnă 25% șanse de a trece de nivelul i la o încercare oarecare).

Pentru ca numărul poate fi unul foarte mare, acesta va fi afișat modulo $M=1.234.567.891$.

(Se poate demonstra că răspunsul poate fi exprimat ca o fracție ireductibilă $\frac{p}{q}$, unde p și q sunt numere întregi și $q \not\equiv 0 \pmod{M}$. Afișați numărul întreg egal cu $p * q^{-1} \pmod{M}$.)

Date de intrare

Fișierul de intrare *tryhard.in* conține pe primul rând numărul de niveluri N . Pe al doilea rând se găsesc N numere naturale.

Date de ieșire

Fișierul de ieșire *tryhard.out* conține o linie, pe care se află numărul cerut.

Restricții și precizări:

- $1 \leq N \leq 10^6$
- $1 \leq P_i \leq 100$

Exemple

tryhard.in	tryhard.out	Explicație
1 25	4	Fiind 25% șanse de reușită, în medie se va juca nivelul de 4 ori.
3 40 10 80	45	