

Skaidymas

Temuras ir jo padėjėjas Ulugbekas ruošiasi magiškam triukui, kurį rodys per šou *Uzbekistan Got Talent*. Triuko metu Temuras išspręs pateiktą **skaidymo uždavinį**: suskaidys pateiktus sveikuosius skaičius į K netuščių grupių taip, kad kiekvienos grupės elementų suma būtų tokia pati. Kitaip sakant, kiekvienas rinkinio sveikasis skaičius turi būti priskirtas į vieną iš K grupių ir kiekvienos grupės elementų suma turi būti vienoda. Pavyzdžiui, jei gautas rinkinys yra $[2, 1, 6, 4, 5]$ ir $K = 3$, galimas skaidymas būtų $[2, 4]$, $[1, 5]$ ir $[6]$. Šiuo atveju kiekvienos grupės elementų suma yra lygiai 6.

Triukas atliekamas šitaip:

- Ulugbekas ir Temuras įeina į atskiras kabinas ir negali bendrauti tarpusavyje.
- Vertinimo komisijos narys pateikia Ulugbekui masyvą A , sudarytą iš N teigiamų sveikųjų skaičių $A[0], A[1], \dots, A[N - 1]$, kur kiekviena reikšmė yra nuo 1 and M imtinai. Komisijos narys Ulugbekui taip pat pateikia K reikšmę.
- Ulugbekas pasirenka daugiausiai $K - 1$ (nebūtinai skirtingų) sveikųjų skaičių tam, kad juos **pridėtų kaip naujus elementus** į masyvą A . Kiekvienas pridedamas sveikasis skaičius taip pat privalo būti nuo 1 iki M imtinai.
- Vertinimo komisija Ulugbeko pasirinktus skaičius įdeda į pradinį masyvą. Šis naujas masyvas tuomet **išrikiuojamas nemažėjimo tvarka** ir duodamas Temurui kartu su K reikšme.
- Tada Temuras privalo išspręsti skaidymo uždavinį šiam naujam, išrikiuotam masyvui.

Sugakvokite ir realizuokite Temuro ir Ulugbeko strategiją. Galima įrodyti, kad duotiems ribojimams egzistuoja tokia strategija abiems, kad jie galėtų sėkmingai išspręsti skaidymo uždavinį, kad ir kokį masyvą A pateiktų vertinimo komisija.

Realizacija

Parašykite dvi procedūras.

Ulugbekui turite parašyti šią procedūrą:

```
std::vector<int> add_numbers(std::vector<int> A, int K, int M)
```

- A : masyvas, kurio ilgis N ir kuris atitinka pradinį Ulugbekui pateikiamą masyvą.
- K : grupių skaičius; pastebėkite, kad Ulugbekas gali pridėti iki $K - 1$ skaičių į masyvą.
- M : didžiausia įmanoma reikšmė pradiniam ir naujai pridedamiems skaičiams.
- Ši procedūra iškviečiama lygiai vieną kartą kiekvienam testui.

Procedūra turi grąžinti masyvą C , sudarytą iš Ulugbeko norimų pridėti į pradinį masyvą skaičių. Tegu S yra masyvo C ilgis.

- S privalo būti daugiausiai $K - 1$.
- Kiekvienas masyvo C elementas privalo būti nuo 1 ir M imtinai.

Temurui turite parašyti šią procedūrą:

```
std::vector<int> find_partition(std::vector<int> B, int K)
```

- B : masyvas, kurio ilgis $N + S$ ir kuriame yra tiek pradiniai skaičiai iš masyvo A , tiek Ulugbeko pridėti skaičiai. Masyvo B elementai išrikiuoti nemažėjimo tvarka.
- K : grupių skaičius.
- Ši procedūra kviečiama lygiai vieną kartą kiekvienam testui.

Procedūra turi grąžinti masyvą P , kuris aprašo masyvo B skaidymą į K grupių, neturinčių bendrų elementų.

- Masyvo P ilgis privalo būti $N + S$.
- Kiekvienam j , kur $0 \leq j < N + S$, elementas $P[j]$ nusako, kuriai grupei $B[j]$ priklauso.
- Grupės turi būti numeruojamos nuo 0 iki $K - 1$. Tai yra, kiekvienam $0 \leq j < N + S$ privalo galioti $0 \leq P[j] < K$.
- Kiekvienam i nuo 0 ir $K - 1$ imtinai, privalo būti bent vienas elementas j ($0 \leq j < N + S$) toks, kad $P[j] = i$.
- Elementų, priskirtų kiekvienai K grupių, suma turi būti vienoda.

Oficialaus vertinimo metu programa, kviečianti aukščiau aprašytas procedūras, vykdoma **lygiai du kartus**.

- Pirmu programos paleidimu:
 - `add_numbers` iškviečiamas lygiai vieną kartą.
 - Grąžintas masyvas apdorojamas vertinimo sistemos ir sudaromas masyvas B .
- Antru programos paleidimu:
 - `find_partition` iškviečiamas lygiai vieną kartą.

Ribojimai

- $3 \leq N \leq 100\,000$
- $2 \leq K \leq 100\,000$
- $K \leq N$
- $1 \leq M \leq 10^9$
- $1 \leq A[i] \leq M$ kiekvienam i , kur $0 \leq i < N$.

Dalinės užduotys

Dalinė užduotis	Taškai	Papildomi ribojimai
1	5	$N = 3$
2	4	$M = 1$
3	7	$M \leq 2$
4	10	$N \leq 10$
5	7	$K = 2$
6	12	$K \leq 3$
7	17	$K \leq 10$
8	20	$K \leq 100$
9	18	Papildomų ribojimų nėra.

Pavyzdys

Panagrinėkime pateiktą iškvietimą:

```
add_numbers([8, 2, 9, 6, 1, 5, 5], 3, 9)
```

Šiame pavyzdyje $A = [8, 2, 9, 6, 1, 5, 5]$. Norime suskaidyti skaičius į $K = 3$ grupių. Ulugbekas gali pridėti skaičius tarp 1 ir $M = 9$.

Ši procedūra gali grąžinti $[5, 4]$, kas reikštų, kad Ulugbekas nusprendžia pridėti $S = 2$ naujus skaičius: 5 ir 4. Šie skaičiai galimi, kadangi abu yra tarp 1 ir $M = 9$.

Naujas masyvas tuomet išrikiuojamas ir pateikiamas Temurui tokiu iškvietimu:

```
find_partition([1, 2, 4, 5, 5, 5, 6, 8, 9], 3)
```

Galime suskaidyti šio masyvo skaičius į šias tris grupes: $[1, 5, 9]$, $[2, 5, 8]$ ir $[4, 5, 6]$. Pastebėkite, kad kiekvienos grupės suma lygi 15. Kad pateiktumėte šį skaidymą, procedūra turi grąžinti masyvą $[0, 1, 2, 0, 1, 2, 2, 1, 0]$.

Pavyzdinė vertinimo programa

Įvesties formatas:

```
N K M
A[0] A[1] ... A[N-1]
```

Rezultatų formatas:

Kai `add_numbers` įvykdoma, pavyzdinė vertinimo programa:

- Sudaro išrikiuotą masyvą B .
- Išveda masyvus C ir B šiuo formatu, **taip pat prideda tuščią eilutę pabaigoje**:

```
S
C[0] C[1] ... C[S-1]
B[0] B[1] ... B[N+S-1]
```

Kai `find_partition` įvykdoma, vertinimo programa išveda:

```
L
P[0] P[1] ... P[L-1]
```

Čia skaičius L yra masyvo P , grąžinto `find_partition` iškvietimo, ilgis.