

Prelucrarea secvențelor de valori

Prelucrarea secvențelor de valori

- 1. Prezentare generală.**
- 2. Sume, produse**
- 3. Algoritmi de numărare**
- 4. Maxime/minime**
- 5. Probleme din concursuri**

Prelucrarea secvențelor de valori

1. Prezentare generală

- în aceste probleme, primim de o serie de date și trebuie să asociem o valoare acestei serii.
- de exemplu, se citesc N numere naturale și trebuie să calculăm suma acestora.
- această valoare este dată de o funcție aplicată pe întreaga secvență
- această funcție poate fi descompusă într-o serie de funcții calculate pe perechi de valori (suma primilor 2 termeni, apoi suma obținută anterior adunată cu al treilea termen etc).

4	2	5	1	→	4	+	2	→	6	+	5	→	11	+	1
---	---	---	---	---	---	---	---	---	---	---	---	---	----	---	---

Prelucrarea secvențelor de valori

- astfel, toate operațiile pentru care matematica folosește o notație „uniformă” pot fi incluse în acest tip de sarcină: **sumă, produs, reunire, intersecție, operații logice, concatenare** etc
- fiecare operație poate fi descompusă până la o **operație cu doi operanzi** și se poate specifica un **element neutru**.
- dacă notăm operația cu **F** și cu **F₀** elementul neutru, putem avea
 - $f = +, *, \cap, \cup, \text{și}, \text{sau}, \text{concatenare}$
 - $F_0 = 0, 1, \text{mulțimea inițială}, \emptyset, A, F, ""$
- Se pornește cu elementul neutru **F₀**, care reprezintă valoarea funcției pentru 0 elemente prelucrate.
- dacă știm rezultatul pentru $i - 1$ elemente (notat cu **F_{i-1}**), se poate calcula **F_i = f(F_{i-1}, X_i)**.

Prelucrarea secvențelor de valori

$$F_i = F(X_1, X_2, \dots, X_i) = \begin{cases} F_0, & \text{daca } i = 0 \\ f(F_{i-1}, X_i), & \text{in caz contrar} \end{cases}$$

```
s ← F0
pentru i ← 1, nr_de_elemente executa
  citește x
  s ← f(s, x)
■
```

Prelucrarea secvențelor de valori

2. Sume, produse

Se citește un număr natural nenul n , apoi o serie de n elemente numere naturale. Să se calculeze suma valorilor citite.

- Numărul de elemente: n
- Termenii x_i se citesc
- $F_0 : 0$
- $f : +$

```
s ← 0
pentru i ← 1, n executa
    citește x
    s ← s + x
scrie s
```

Prelucrarea secvențelor de valori

Bomboane

Lucy lucrează n ore într-o fabrică de bomboane. Bomboanele trebuie împachetate și puse în cutii. La început, pe bandă, sosesc câte k bomboane în fiecare oră. Din p în p ore bomboanele sunt transportate astfel: dacă în cutii sunt un număr par de bomboane jumătate din ele vor fi transportate, iar dacă în cutii sunt un număr impar de bomboane Lucy primește cadou o bomboană, iar din ceea ce a rămas jumătate sunt transportate. După un transport numărul de bomboane care vin pe bandă crește cu 1. Calculați câte bomboane se găsesc în cutii după n ore.

Date de intrare: numere naturale n , k și p , cu semnificația din enunț. ($1 \leq p \leq n \leq 100\,000$, $1 \leq k \leq 2\,000$)

Date de ieșire: numărul de bomboane care se găsesc în cutii după n ore.

Exemplu:

Intrare	iesire
8 4 3	22

Prelucrarea secvențelor de valori

- În unele probleme termenii se determină pe baza unui calcul
- Numarul de elemente: n
- Termenii k_i = numarul de bomboane împachetate în ora i
- $F_0 : 0$
- $f : +$

Descrierea algoritmului:

- Vom simula procesul, timp de cele n ore
- inițial $s \leftarrow 0$ bomboane impachetate
- pentru fiecare ora $i \leftarrow 1, n$ adăugăm în cutie bomboanele împachetate în ora respectiva $s \leftarrow s + k$

Prelucrarea secvențelor de valori

- dacă ajungem la o ora în care are loc transportul (o ora divizibilă cu p) transportăm jumătate din bomboanele împachetate și crește numărul de bomboane care sosesc pe bandă: $s \leftarrow s/2$ și $k \leftarrow k + 1$

```
citește n, k, p
s ← 0 //bomboanele împachetate
pentru i ← 1, n executa //pentru fiecare oră din cele n
    s ← s + k
    dacă i % p = 0 atunci
        s ← s / 2, k ← k + 1
scrie s
```


Prelucrarea secvențelor de valori

Vizite

Butelian locuiește în casa cu numărul **1**, într-un oraș care are **n** case construite în jurul sensului giratoriu. Casele sunt numerotate de la **1** la **n** în sensul acelor de ceasornic. Traficul este cu sens unic în sensul acelor de ceasornic.



Butelian primește sarcina să își viziteze vecinii din anumite case precizate a_i . Dar, pentru a nu supăra pe nimeni va vizita toate casele cu numere mai mici decât a_i . Mutarea dintr-o casă în una vecină de-a lungul șoselei de centură durează o unitate de timp.

Date de intrare: prima linie **n** și **m** ($2 \leq n, m \leq 10^5$), numărul de case, respectiv numărul de vecini care trebuie vizitați. A doua linie conține **m** numere întregi $a_1, a_2, \dots, a_m$ ($1 \leq a_i \leq n$).

Date de ieșire: timpul minim să viziteze vecinii indicați

Prelucrarea secvențelor de valori

Exemple:

Intrare	ieșire	Explicație
4 3	6	$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 1 \rightarrow 2 \rightarrow 3$
3 2 3		

Descrierea algoritmului:

- se aplica un algoritm de prelucrare a secvențelor de valori pentru calculul sumei
- termenii sumei vor fi timpii necesari deplasării între două case precizate în listă
- să presupunem că Butelian se află în casa numerotată **a** și următorul vecin vizitat se află în casa **b**. Dacă $b \geq a$ atunci timpul va fi $b - a$

Prelucrarea secvențelor de valori

secunde. În caz contrar, trebuie să facă traseul $a \rightarrow n \rightarrow 1 \rightarrow b$, în total $n - a + b$ secunde.

```
citește n, m
a ← 1
s ← 0 //timpul total
pentru i ← 1, m executa
    citește b
    dacă b ≥ a atunci
        s ← s + (b - a)
    altfel
        s ← s + (n - a + b)
    ■
a ← b //suntem în casa b
■
scrie s
```

Prelucrarea secvențelor de valori

3. Algoritmi de numărare

- În prelucrarea secvențelor este posibil ca mai multe elemente să aibă o proprietate cerută.
- ar putea fi interesant să examinăm câte astfel de elemente există.
- este echivalent cu algoritmul de sumă în care trebuie să adăugăm 1, dar în funcție de o anumită condiție.

```
s ← 0
pentru i ← 1, n executa
    citește x
    dacă x îndeplinește condiția atunci
        s ← s + 1
scrie s
```

Prelucrarea secvențelor de valori

TestIQ

Butelian se pregătește să treacă testul IQ. Cea mai frecventă sarcină în acest test este de a afla care dintre n numere date diferă de celelalte. Butelian a observat că un număr diferă de celelalte prin paritate.

Determinați poziția numărului care diferă de celelalte prin paritatea sa.

Date de intrare: prima linie n ($1 \leq n \leq 100$). A doua linie conține n numere întregi $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 100$). Este garantat că exact unul dintre aceste numere diferă de celelalte prin paritate.

Date de ieșire: poziția numărului care diferă de celelalte prin paritate. Numerele sunt numerotate de la 1 în ordinea introducerii.

Exemplu:

Intrare	ieșire
5	3
2 4 7 8 6	

Prelucrarea secvențelor de valori

- folosim 2 variabile **nrp** și **nri** care rețin numărul de elemente pare respectiv impare din serie.
- **pozp** și **pozi** ca index al ultimului element par/impar întâlnit.
- dacă apare doar un număr impar afișăm **pozi**; în caz contrar, apare un singur număr par, afișăm **pozp**.

```
nrp ← 0, nri ← 0
citeste n
pentru i ← 1, n executa
  citeste x
  daca x % 2 = 0 atunci
    nrp ← nrp + 1, pozp ← i
  altfel
    nri ← nri + 1, pozi ← i
  ■
■
```

Prelucrarea secvențelor de valori

```
daca nrp = 1 atunci  
    scrie pozp  
altfel  
    scrie pozi
```



Prelucrarea secvențelor de valori

Perle

Pe masă sunt n perle, fiecare dintre ele poate fi roșie, verde sau albastră. Determinați numărul minim de perle care trebuie luate de pe masă, astfel încât oricare două perle învecinate să aibă culori diferite.

Date de intrare: prima linie n ($1 \leq n \leq 100$). A doua linie conține n caractere, separate prin câte un spațiu, din mulțimea **R, B, G**

Date de ieșire: numărul minim de perle care trebuie luate de pe masă, astfel încât oricare două perle învecinate să aibă culori diferite.

Exemplu:

Intrare	ieșire
3	1
R R G	

Prelucrarea secvențelor de valori

- soluția este numărul de perechi de litere egale situate pe poziții consecutive.

```
citeste n
nr ← 0
citeste a //prima perla
pentru i ← 2, n executa
    citeste b
    daca a = b atunci
        nr ← nr + 1
    a ← b
scrie nr
```

Prelucrarea secvențelor de valori

Poliție

Departamentul de poliție tocmai și-a început activitatea. Inițial, nu au agenți, așa că, au început să angajeze noi ofițeri.

Între timp, infracțiunile continuă să se producă în oraș. Un membru al poliției poate investiga o singură infracțiune în timpul carierei sale. Dacă nu există niciun ofițer de poliție liber (nu este ocupat cu cercetarea) în timpul producerii unei infracțiuni, aceasta va rămâne necercetată.

Cerință cunoscând ordinea cronologică a crimelor și a angajărilor, găsiți numărul de infracțiuni care nu vor fi cercetate.

Date de intrare Prima linie de intrare va conține un număr întreg n ($1 \leq n \leq 10^5$), numărul de evenimente. Următoarea linie va conține n numere întregi separate prin spațiu. Dacă numărul întreg este **-1** înseamnă că a avut loc o crimă. În caz contrar, numărul întreg va fi

Prelucrarea secvențelor de valori

pozitiv, numărul de ofițeri recrutați împreună la acel moment. Nu vor fi recrutați mai mult de 10 ofițeri simultan.

Date de ieșire: un număr întreg, numărul de infracțiuni care nu vor fi cercetate.

Exemple

intrare

8
1 -1 1 -1 -1 1 1 1

ieșire

1

intrare

11
-1 -1 2 -1 -1 -1 -1 -1 -1 -1 -1

ieșire

8

Prelucrarea secvențelor de valori

Date și structuri de date folosite:

- nro – numărul de ofițeri disponibili
- nr – nr de infracțiuni necercetate
- x – operațiunea curentă (-1 = infracțiune, altfel angajare)

```
citeste n
nro ← 0, nr ← 0
pentru i ← 1, n executa
    citeste x //evenimentul
    daca x = -1 atunci //daca este infractiune
        daca nro = 0 atunci //daca nu avem ofiteri disponibili
            nr ← nr + 1 //infractiune necercetata
            altfel nro ← nro - 1 //un ofiter cerceteaza
        altfel nro ← nro + x //angajare x ofiteri
scrie nr
```

Prelucrarea secvențelor de valori

4. Maxime/minime

- în unele probleme se cere **determinarea valorii minime/maxime** dintr-o serie de valori.

```
citeste n
max ← -∞, min ← ∞
pentru i ← 1, n executa
    citeste x
    daca x > max atunci
        max ← x
    ■
    daca x < min atunci
        min ← x
    ■
■
```

Prelucrarea secvențelor de valori

Zero

Butelian are un șir X de n numere naturale. În cadrul unei operații el poate să incrementeze sau să decrementeze cu **1** valoarea oricărui element. Determinați numărul minim de operații necesare pentru ca produsul $X_1 * X_2 * \dots * X_n = 0$.

Date de intrare: de pe prima linie se citește n ($1 \leq n \leq 100000$), iar de pe a doua linie un șir de n numere naturale ($-1000 \leq X_i \leq 1000$)

Date de ieșire: numărul minim de operații pentru ca produsul să devină nul.

Exemple

Intrare	ieșire
---------	--------

3	2
---	---

2 -6 5	
--------	--

Soluție: pentru ca produsul tuturor elementelor să fie 0, cel puțin un element trebuie să fie 0.

- pentru fiecare X_i , numărul minim de operații pentru a-l transforma în 0 este $|X_i|$.
- prin urmare, numărul minim de operații pentru a transforma cel puțin un element în 0 este valoarea minimă a lui $|X_i|$ din toate elementele.

Prelucrarea secvențelor de valori

- pot exista cazuri când trebuie găsită **poziția primei sau a ultimei valori maxime** sau minime în cadrul seriei
- exemplul de mai jos determină poziția primului și ultimului element maxim

```
citeste n
max ← -∞
prim ← -1, ultim ← -1
pentru i ← 1, n executa
    citeste x
    daca x > max atunci
        prim ← i, max ← x
    ■
    daca x ≥ max atunci
        ultim ← i, max ← x
    ■
■
```

Prelucrarea secvențelor de valori

Tramvai

Sunt n stații de tramvai numerotate de la **1** la n . În fiecare stație coboară a_i persoane și urcă b_i persoane. Tramvaiul este gol înainte de a ajunge la prima stație, iar când tramvaiul ajunge la ultima oprire, toți pasagerii coboară. Să se determine capacitatea minimă a tramvaiului astfel încât niciun pasager să nu rămână pe dinafară și numărul stației la care tramvaiul ajunge cu un număr maxim de pasageri.

Date de intrare: prima linie n ($1 \leq n \leq 100000$), apoi n perechi de numere naturale a_i și b_i ($0 \leq a_i, b_i \leq 1000$).

Date de ieșire: capacitatea minimă a tramvaiului și numărul stației la care tramvaiul ajunge cu un număr maxim de pasageri (dacă există mai multe se afișează prima stație).

Exemplu

4	iesire
0 3	6 3
2 5	
4 2	
4 0	

Prelucrarea secvențelor de valori

- unele probleme pot solicita **calculul numărului de apariții** ale valorii maxime sau minime.

```
citeste n
max ← -∞, nrap ← 0
pentru i ← 1, n executa
    citeste x
    daca x > max atunci
        max ← x, nrap ← 1
    altfel
        daca x = max atunci
            nrap ← nrap + 1
```

Prelucrarea secvențelor de valori

Cearta

Butelian și Flaconel s-au certat, astfel că ei vor să plece în două localități între care diferența de altitudine este maximă. Determinați câte astfel de perechi de localități există.

Date de intrare: de pe prima linie se citește n ($1 \leq n \leq 100000$), iar de pe a doua linie un șir de n numere naturale reprezentând altitudinea la care se găsesc localitățile ($0 \leq X_i \leq 10000$)

Date de ieșire: numărul de perechi de localități cu diferența maximă de altitudine.

Exemple

Intrare

ieșire

5

2

6 2 3 8 1

Prelucrarea secvențelor de valori

- se poate folosi un algoritm de maxim/minim și pentru determinarea celei de a doua valori maxime sau minime

```
citeste n
max1 ← -∞, max2 ← -∞ // cea mai mare, resp a 2-a cea mai mare
pentru i ← 1, n executa
  citeste x
  daca x > max1 atunci
    max2 ← max1, max1 ← x
  altfel
    daca x > max2 atunci
      max2 ← max2
```

Prelucrarea secvențelor de valori

Licitație

Într-o licitație la prețul doi, la fel ca în licitația obișnuită, ofertanții plasează o ofertă care este prețul pe care ofertantul este gata să-l plătească. Câștigătorul licitației este participantul care a oferit cel mai mare preț. Cu toate acestea, el nu plătește prețul pe care îl oferă, ci prețul cel mai mare dintre ofertele celorlalți participanți (de unde și numele: licitația cu prețul doi).

Determinați indicele câștigătorului și prețul pe care îl va plăti.

Date de intrare: Prima linie conține n ($2 \leq n \leq 1000$) — numărul de ofertanți. A doua linie conține n numere întregi distincte două câte două $p_1, p_2, \dots, p_n$ ($1 \leq p_i \leq 10000$).

Date de ieșire: două numere întregi indicele câștigătorului și prețul pe care îl va plăti.

Exemplu:

Intrare	ieșire
6	4 5
4 2 5 7 1 3	

Prelucrarea secvențelor de valori

5. Probleme din concursuri

<https://www.pbinfo.ro/probleme/3736/sir15>

<https://www.pbinfo.ro/probleme/2979/cartele1>

<https://www.pbinfo.ro/probleme/2007/numere17>

<https://www.pbinfo.ro/probleme/1051/bete1>