Descrierea soluției - Problema Cetățuie, Lot de Informatica Seniori, Cluj 2022

Andrei-Robert Ion

February 2024

Cetățuie

Author: George-Alexandru Râpeanu, Ioan Popescu, Ștefan Dăscălescu

Observații despre sumă Dacă în urma unei operații valoarea de la poziția i crește cu o unitate ($V_i \to V_i + 1$), suma $|V_0 - V_1| + |V_1 - V_2| + \cdots + |V_{n-2} - V_{n-1}|$ scade în cazul în care $V_i < V_{i-1}, V_i > V_{i+1}$, crește în cazul în care $V_i > V_{i-1}, V_i > V_{i+1}$ și în orice alt caz nu își schimbă valoarea. Când $i \in \{0, n-1\}$, suma se schimbă cu o diferență de 1, și pentru orice $1 \leq i \leq n-2$ se schimbă cu o diferență de 2. De aceea, primele operații pe care le efectuăm sunt pe poziții în care suma scade cu 2 în urma unei singure mișcări, iar la epuizarea acestor posibilități efectuăm operații pe pozițiile în care suma scade cu 1.

Observații despre subsecvențe de numere egale Dacă la un anumit moment putem identifica o subsecvență formată din elementele $V_l = V_{l+1} = \cdots = V_r = \mathcal{V}$ cu $0 \leq l \leq r \leq n-1$, iar $V_{l-1} > \mathcal{V} < V_{r+1}$, atunci putem reduce suma care trebuie minimizată cu 2 folosind $r - l + 1$ operații, câte una pentru fiecare poziție $i, l \leq i \leq r$ (mai puțin in cazul când $l = 0$ sau $r = n-1$, când suma scade doar cu 1). Vom numi o subsecvență cu această proprietate un segment.

Identificarea segmentelor Pentru a putea identifica rapid subsecvențele de numere egale ce se pot forma, vom folosi metoda unei stive monotone pentru a reține la fiecare poziție i prima poziție din dreapta ei, fie aceasta j, care are $V_i < V_j$, alături de cea mai apropiată poziție din stânga, fie aceasta k cu aceeași proprietate $V_k > V_i$. Astfel, putem împărți șirul de numere în segmente cu $l = k + 1, r = j - 1$. În plus, segmentele ce vor apărea în viitor sunt calculate în același timp. Astfel, pentru o poziție i vom considera că ni se oferă de $\texttt{min}(V_{l-1}, V_{r+1}) - V_i$ ori opțiunea de a folosi $r - l + 1$ mișcări pentru a scădea 2 (sau 1, daca $l = 0$ sau $r = n - 1$) din sumă.

Problema rucsacului Putem transpune informația obținută despre segmente într-un rucsac. Deoarece numărul de elemente poate fi mare (până la 10^9), dar numărul de elemente distincte este mic (mai mic decât N), vom reține frecvența fiecărui element pentru a nu depăși limita de memorie. Evident, putem adăuga elementele care nu reprezintă operații valide la momentul curent de timp, deoarece profitul lor este mai mic sau egal cu profitul operațiilor care trebuie executate primele, dar lungimea segmentelor pe care le reprezintă este mai mare, deci au și costul (ca număr de mișcări asociat) mai mare. Astfel, trebuie să rezolvăm problema rucsacului când fiecare element are profitul $p \in \{1, 2\}$.

Rezolvarea rucsacului Problema rucsacului când profiturile sunt mici poate fi aproximată cu un grad mare de precizie folosind un algoritm de tip Greedy, în care sortăm elementele în funcția raportului de profit p la cost c în ordine descrescătoare. Astfel, prioritizăm elementele cu $r = \frac{p}{c}$ maxim. Când $p \leq 2$, soluția optimă se poate obține încercând să înlocuim elementul de profit $p = 1$ și cost maxim (dacă acesta există) cu cel mai ieftin element de profit $p = 2$ pe care nu l-am putut lua înainte.

Concluzie Problema se poate rezolva în $\mathcal{O}(N)$ timp și spațiu. Tehnica stivei monotone se încadrează în aceste limite de timp, iar sortarea elementelor în rucsac se poate face prin sortarea elementelor cu $p = 1$ și $p = 2$ separat folosind sortări prin numărare (după $c = r - l + 1$, lungimea intervalului $\leq N$), iar apoi prin interclasarea celor doi vectori. În final, soluția este egală cu suma inițială, din care scădem profitul maxim obținut din rucsac, calculat pentru K operații.

Indicii de implementare Putem borda șirul de numere $V_0, V_1, \dots, V_{n-1}$ cu elementele $V_{-1} = V_n = \frac{K}{N} + \texttt{max}_{i=0}^{n}(V_i) + 1$. De asemenea, rezultatul final și multe valori intermediare nu încap în tipuri de date întregi cu doar 32 de biți.