

Algoritmi avansati Explicarea temei

Subsir crescator maximal

Ideea de N^2 este destul de straightforward. Vrem o dinamica $dp[i] =$ lungimea scmax care se termina pe pozitia i . Recurenta: $dp[i] = \max(dp[j] + 1)$ unde $j < i$, $A[j] < A[i]$.

Pentru a ajunge la complexitatea dorita de $N \cdot \log$, vom normaliza numerele (nu conteaza valorile, ci doar relatiile de ordine), si vom schimba recurenta sa fie: $dp[i] =$ lungimea scmax care se termina in valoarea i . Astfel, recurenta devine $dp[i] = \max(dp[1], dp[2], \dots, dp[i-1])$, acest maxim putand fi calculat cu un arbore de intervale.

Evaluare sursa: https://www.infoarena.ro/job_detail/3194561

Lastk

Vrem ca la fiecare pas sa retinem doar candidati pentru cele mai mari numere din sir, deoarece sunt exact k . Asta ne va da o complexitate de $N \log K$ in loc de $N \log N$ (ex. sortare). Pentru a face acest lucru, putem sa tinem candidatii intr-un heap. La fiecare pas, vom adauga elementul nou in heap si vom taia elementul minim.

Evaluare sursa: <https://www.pbinfo.ro/detalii-evaluare/48216913>.