

Range Minimum Query

Oprea Olivia Maria-Magdalena 323CA

Descrierea problemei rezolvate

Subiectul abordat este “Range Minimum Query”, care se axează pe găsirea eficientă a elementului minim de la indexul x la indexul y (inceputul respectiv finalul intervalului) dintr-un vector dat. O aplicație practică a acestui subiect este “Problema celui mai mic stramos comun” și “Problema celui mai lung prefix comun”.

Solutii alese

1. Square Root Decomposition

Folosim această metodă pentru a reduce spațiul cerut în felul următor: vom împărți vectorul în blocuri diferite de lungime aproximativ egale cu \sqrt{n} , iar apoi vom stoca poziția celui mai mic element din fiecare bloc.

Preprocesarea durează $O(n)$, iar spațiul alocat este $O(\sqrt{n})$.

2. Sparse Table Algorithm

Această metodă necesită $O(\sqrt{n})$ spațiu, dar timpul de interogare este $O(\sqrt{n})$. Tot această metodă acceptă timpul de interogare $O(1)$ cu un spațiu suplimentar de $O(n \log n)$.

Creem o matrice cu n linii și cel mai apropiat număr de $\log n + 1$ coloane. Matricea se completează astfel: pornim de la poziția i și căutăm elementul minim a 2^j elemente și stocăm poziția elementului minim. Dacă nu se pot compara cele 2^j elemente, poziția din matrice a elementului minim va rămâne goală.

Pentru a găsi elementul minim dintr-un interval se calculează lungimea acestuia și se compară primul element cu următoarele $2^{\log n}$ elemente, iar dacă diferența dintre n și următoarele elemente este mai mare decât 0 algoritmul se reia.

3. Segment Tree

Am ales această soluție deoarece nu avem voie să folosim soluția banală.

Timpul de preprocesare este $O(n)$, iar timpul pentru găsirea elementului minim este $O(\log n)$, spațiul necesar în plus este $O(n)$ pentru a stoca metoda.

Metoda se construiește de la un vector dat. Împărțim segmentul curent în 2 jumătăți (dacă nu este deja un segment de lungime 1), iar apoi apelăm aceeași procedură pe cele 2 jumătăți și pentru fiecare astfel de segment stocăm valoarea minimă într-un segment tree. Înafara de ultimul nivel, arborele este completat în totalitate și este un arbore binar.

Criterii de evaluare

Deoarece cautam elementul minim dintr-un interval vom avea nevoie de diferiti vectori pentru a valida algoritmul nostru si pentru a-i testa corectitudinea.

Vectorii alesi vor fi umpluti pe rand cu numere positive, negative, ambele, numere ordonate crescator, descrescator, piramida (crescator pana intr-un punct, apoi descrescator), partial crescatoare sau partial descrescatoare, elemente random.

Referinte

<https://www.geeksforgeeks.org/range-minimum-query-for-static-array/>

<https://www.geeksforgeeks.org/segment-tree-set-1-range-minimum-query/>

<https://www.geeksforgeeks.org/sqrt-square-root-decomposition-technique-set-1-introduction/>

https://en.wikipedia.org/wiki/Range_minimum_query

<https://cp-algorithms.com/sequences/rmq.html>

<https://web.stanford.edu/class/cs166/lectures/01/Small01.pdf>

<https://www.youtube.com/watch?v=ZBHKZF5w4YU>

https://www.youtube.com/watch?v=c5O7E_PDO4U

<https://www.youtube.com/watch?v=0CPBgO4Mg-E>