Range Minimum Query

Brebu Costin-Bogdan

Facultatea de automatică și calculatoare - 321 CD

1. Introducere

În stiința computerelor, problema elementului minim dintr-un interval se ocupă cu gasirea valorii minime dintr-un subvector al unui vector cu elemente comparabile.

Există multe aplicatii in legătură cu aceasta problem, dar cele mai cunoscute:

- The lowest common ancestor problem
  Un exemplu relevant este analiza ADN pentru a gasi stramosul comun a două specii.
- The longest common prefix problem.
  Gasirea celui mai lung prefix comun a doua stringuri.

2. Soluții posibile (1)

Există multe metode de a rezolva problema elementului minim cum ar fi:

- Sqrt-decomposition
  - complexitate preprocesare $O(N)$
  - complexitate per interval $O(\sqrt{\mathrm{N}})$
- Segment tree
  - complexitate preprocesare $O(N)$
  - complexitate per interval $O(\log\mathrm{N})$
- Fenwick tree
  - complexitate preprocesare $O(N\log N)$
  - complexitate per interval $O(\log\mathrm{N})$
- Sparse Table
  - complexitate preprocesare $O(N\log N)$
  - complexitate per interval $O(1)$

- Sqrt Tree
  - complexitate preprocesare $O(Nlog\log N)$
  - complexitate per interval $O(1)$
- Disjoint Set Union / Arpa's Trick
  - complexitate preprocesare $O(N)$
  - complexitate per interval $O(1)$
- Cartesian Tree and Farach-Colton and Bender algorithm
  - complexitate preprocesare $O(N)$
  - complexitate per interval $O(1)$

Metode alese:

- Sparse Table **(2)**

E un algoritm care are complexitate de $O(1)$ pentru fiecare interval in parte si o complexitate $O(N\log\log N)$ pentru preprocesare.

Acesta are o eficiență bună in aflarea minimului dintr-un interval. Singurul dezavantaj al acestei metode este ca vectorul nu poate fi modificat fara a relua preprocesarea si intreg procesul.

Un sparse table este o matrice a cărei elemente respecta regula:

$$st_{ij} = query(i, i + 2^j - 1)$$

Astfel minimul dintr-un interval dat se va extrage din matrice.

- Segment tree **(3)**

Acest algoritm este flexibil si accepta modificarea vectorului cu un element sau cu un intreg segment.

Un “segment tree” este în general o structură flexibilă, cu care se pot rezolva o mulțime de probleme. De exemplu, cu un “segment tree” bidimensional putem afla suma elementeor unei submatrice a unei matrice date.

Avantaj: complexitate bună, admite actualizări pe vector

Dezavantaj: implementare dificilă

- Sqrt-decomposition **(4)**

In preprocesare se imparte vectorul in bucati de lungime aproximativa $\sqrt{n}$, iar pentru fiecare subvector se reține minimul într-un alt vector creat de noi.

Aproape toate bucațile au lungime $\sqrt{n}$ cu excepția ultimei. Ultima bucata are mai putine elemente daca n nu se imparte exact la $\sqrt{n}$.

Această metodă găseste minimul dintr-un interval in $O(\sqrt{\mathrm{n}})$ oprtatii, fiind mult mai rapidă comparativ cu solutia banală care executa $O(n)$ operatii.

3. Criteriile de evaluare pentru soluția propusă

Mi-am propus pentru testarea algoritmilor sa mă gândesc la o serie de teste variate.
Să am teste in care se dau multe intervale si teste in care se dau mai putine și să incerc modificarea vectorului in timpul verificării intervalelor pentr a vedea ce se întâmplă cu fiecare algoritm in parte.
O sa creez cate 3 serii de teste pentru fiecare din cazurile:

- Numărul de interogări e mult mai mic decât numărul de elemente
- Numărul de interogări e proportional cu numărul de elemente
- Numărul de interogări e mult mai mare decât numărul de elemente

4. Referinte

(1) https://cp-algorithms.com/sequences/rmq.html
(2) https://cp-algorithms.com/data_structures/sparse-table.html
(3) https://cp-algorithms.com/data_structures/segment_tree.html
(4) https://cp-algorithms.com/data_structures/sparse-table.html