

Fenwick Tree (BIT)

Inversões

Laboratório de Programação Competitiva - 2020

Pedro Henrique Paiola

Contando inversões

- Uma possível aplicação para a BIT é a contagem de inversões em um vetor.
- Uma inversão consiste em um par (i,j) tal que $v[i] > v[j]$

	0	1	2	3	4	5	6	7	8
v	1	3	3	5	4	6	7	9	10

- No vetor acima, por exemplo, temos uma inversão em $(3,4)$

Contando inversões

- Para resolver este problema, iremos construir uma BIT que conta quantas vezes um certo número apareceu.
- Ou seja, $\text{query}(\text{BIT}, i) = \text{quantidade de números menor ou igual a } i$
- Percorremos o vetor da direita para a esquerda, consultando a BIT para saber a quantidade de inversões (quantos números menor do que $a[i]$ aparecerem DEPOIS DE i) e a atualizamos em seguida.

Contando inversões

```
contar_inversoes(v, n):  
    max_elem = maior v[i]  
    BIT[max_elem+1]  
    cont_inv = 0  
    para i = n - 1 até 0  
        cont_inv += query(BIT, v[i] - 1);  
        updateAdd(BIT, v[i], 1);  
    retorna cont_inv
```

Contando inversões

	0	1	2	3
v	8	4	2	1

	0	1	2	3	4	5	6	7	8
P	0	0	0	0	0	0	0	0	0

Inversões: 0

Contando inversões

	0	1	2	3
v	8	4	2	1

	0	1	2	3	4	5	6	7	8
P	0	0	0	0	0	0	0	0	0

Inversões: 0

Contando inversões

	0	1	2	3
v	8	4	2	1

	0	1	2	3	4	5	6	7	8
P	0	1	1	1	1	1	1	1	1

Inversões: 0

Contando inversões

	0	1	2	3
v	8	4	2	1

	0	1	2	3	4	5	6	7	8
P	0	1	1	1	1	1	1	1	1

Inversões: $0 + 1$

Contando inversões

	0	1	2	3
v	8	4	2	1

	0	1	2	3	4	5	6	7	8
P	0	1	2	2	2	2	2	2	2

Inversões: 1

Contando inversões

	0	1	2	3
v	8	4	2	1

	0	1	2	3	4	5	6	7	8
P	0	1	2	2	2	2	2	2	2

Inversões: $1 + 2$

Contando inversões

	0	1	2	3
v	8	4	2	1

	0	1	2	3	4	5	6	7	8
P	0	1	2	2	3	3	3	3	3

Inversões: 3

Contando inversões

	0	1	2	3
v	8	4	2	1

	0	1	2	3	4	5	6	7	8
P	0	1	2	2	3	3	3	3	3

Inversões: $3 + 3$

Contando inversões

	0	1	2	3
v	8	4	2	1

	0	1	2	3	4	5	6	7	8
P	0	1	2	2	3	3	3	3	4

Inversões: 6

Contando inversões

- Em muitos casos temos problemas em construir uma BIT tendo uma posição para cada elemento, pois o intervalo de dados pode ser muito grande, e incluir números negativos.
- Porém, se nosso vetor tem tamanho n , temos no máximo n diferentes valores.
- Podemos remapear seus valores, compactando o intervalo para ele possuir comprimento n .
- Basicamente, essa conversão é feita ordenando o vetor, e então substituindo cada elemento pela posição que ele assumiria no vetor ordenado.
- Mais detalhes no link:

<https://www.geeksforgeeks.org/count-inversions-array-set-3-using-bit/>

Contando inversões

	0	1	2	3
v	16	-3	18	5

	0	1	2	3
aux				

	0	1	2	3
v2				

Contando inversões

	0	1	2	3
v	16	-3	18	5

	0	1	2	3
aux	-3	5	16	18

	0	1	2	3
v2				

Contando inversões

	0	1	2	3
v	16	-3	18	5

	0	1	2	3
aux	-3	5	16	18

	0	1	2	3
v2	2			

Contando inversões

	0	1	2	3
v	16	-3	18	5

	0	1	2	3
aux	-3	5	16	18

	0	1	2	3
v2	2	0		

Contando inversões

	0	1	2	3
v	16	-3	18	5

	0	1	2	3
aux	-3	5	16	18

	0	1	2	3
v2	2	0	3	

Contando inversões

	0	1	2	3
v	16	-3	18	5

	0	1	2	3
aux	-3	5	16	18

	0	1	2	3
v2	2	0	3	1

Contando inversões

	0	1	2	3
v	16	-3	18	5

	0	1	2	3
aux	-3	5	16	18

	0	1	2	3
v2	2	0	3	1