

SELECTION SORT

Estrutura de dados

SELECTION SORT

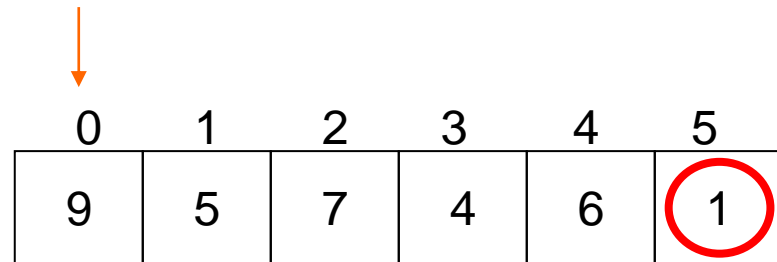
- É o algoritmo mais simples de implementar, mas é também o mais ineficiente de todos os aqui apresentados;
- Dado um vetor não ordenado, o método seleciona o menor item do vetor e faz a troca com o item da primeira posição.
- Essas duas operações são repetidas com os $n-1$ itens restantes, depois com os $n-2$ itens, até que reste apenas um elemento.

Estrutura de dados

SELECTION SORT

Chaves Iniciais:

$i=0$:

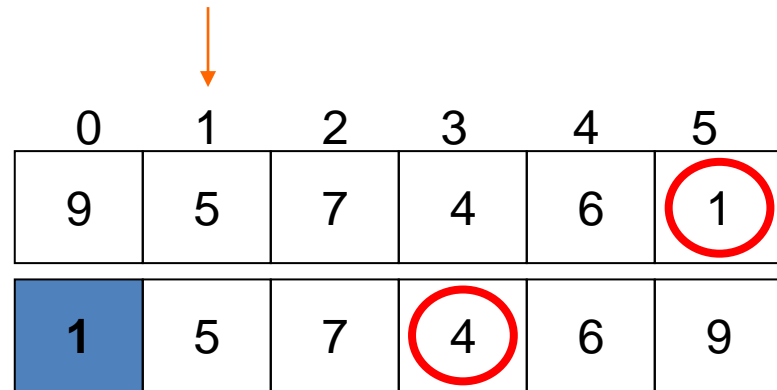


0	1	2	3	4	5
9	5	7	4	6	1

Estrutura de dados

SELECTION SORT

Chaves Iniciais:



The diagram illustrates the first two iterations of Selection Sort on an array of six numbers: 9, 5, 7, 4, 6, 1. The array is represented as a table with two rows. The first row, labeled 'Chaves Iniciais', shows the initial state where the element 1 at index 5 is circled in red, indicating it is the minimum element found. An orange arrow points down to index 1. The second row, labeled 'i=0:', shows the result after the first swap: the element 1 is now at index 0 (highlighted in blue), and the element 4 at index 3 is circled in red, indicating it is the minimum element in the remaining subarray. The third row, labeled 'i=1:', is empty.

0	1	2	3	4	5
9	5	7	4	6	1
1	5	7	4	6	9


i=0:

i=1:

Estrutura de dados

SELECTION SORT

Chaves Iniciais:



0	1	2	3	4	5
9	5	7	4	6	1

i=0:

1	5	7	4	6	9
---	---	---	---	---	---

i=1:


1	4	7	5	6	9
---	---	---	---	---	---

i=2:

Estrutura de dados

SELECTION SORT

Chaves Iniciais:



0	1	2	3	4	5
9	5	7	4	6	1

i=0:

1	5	7	4	6	9
---	---	---	---	---	---

i=1:

1	4	7	5	6	9
---	---	---	---	---	---

i=2:


1	4	5	7	6	9
---	---	---	---	---	---

i=3:

Estrutura de dados

SELECTION SORT

Chaves Iniciais:



0	1	2	3	4	5
9	5	7	4	6	1

i=0:

1	5	7	4	6	9
---	---	---	---	---	---

i=1:

1	4	7	5	6	9
---	---	---	---	---	---

i=2:

1	4	5	7	6	9
---	---	---	---	---	---

i=3:


1	4	5	6	7	9
---	---	---	---	---	---

i=4:

Estrutura de dados

SELECTION SORT

Chaves Iniciais:



0	1	2	3	4	5
9	5	7	4	6	1

i=0:

1	5	7	4	6	9
---	---	---	---	---	---

i=1:

1	4	7	5	6	9
---	---	---	---	---	---

i=2:

1	4	5	7	6	9
---	---	---	---	---	---

i=3:

1	4	5	6	7	9
---	---	---	---	---	---

i=4:

1	4	5	6	7	9
---	---	---	---	---	---

Estrutura de dados

SELECTION SORT

Chaves Iniciais:

0	1	2	3	4	5
9	5	7	4	6	1

i=0:

1	5	7	4	6	9
---	---	---	---	---	---

i=1:

1	4	7	5	6	9
---	---	---	---	---	---

i=2:

1	4	5	7	6	9
---	---	---	---	---	---

i=3:

1	4	5	6	7	9
---	---	---	---	---	---

i=4:

1	4	5	6	7	9
---	---	---	---	---	---

Estrutura de dados

SELECTION SORT

```
void selecaoDireta(int *vet, int n){  
    int i, j, min;  
    for(i=0; i< n-1; i++){  
        min=i;  
        for (j= i+1; j<n; j++){  
            if (vet[j]< vet[min])  
                min=j;  
        }  
        if (i!=min)  
            troca(vet, i, min);  
    }  
}
```

```
void troca(int *vet, int p1, int p2){  
    int aux;  
    aux= vet[p1];  
    vet[p1]= vet[p2];  
    vet[p2]=aux;  
}
```

Estrutura de dados

SELECTION SORT - EXERCÍCIO

Para o método de ordenação de seleção, apresente a ordenação do seguinte vetor. Mostre a sequência de todas as etapas executadas (comparações e trocas).

$$V = \{11; 9; 7; 5; 3; 1\}$$