

Selection Sort

Murilo Dantas

Selection Sort

- Funcionamento

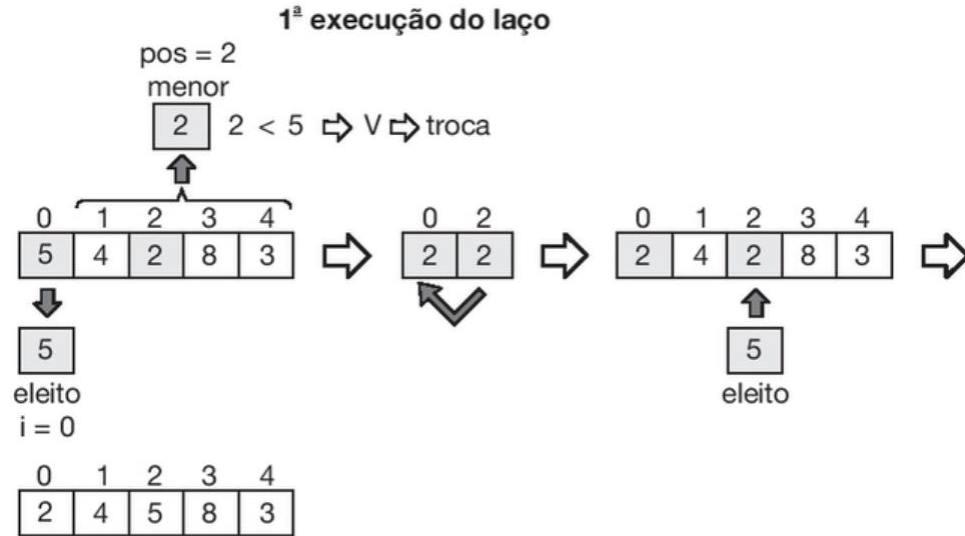
- ▶ Cada número do vetor, a partir do primeiro, será eleito e comparado com o menor ou maior, dentre os que estão à direita do eleito.
- ▶ Procura-se um número menor (crescente) ou maior (decrescente) que o eleito.

Selection Sort

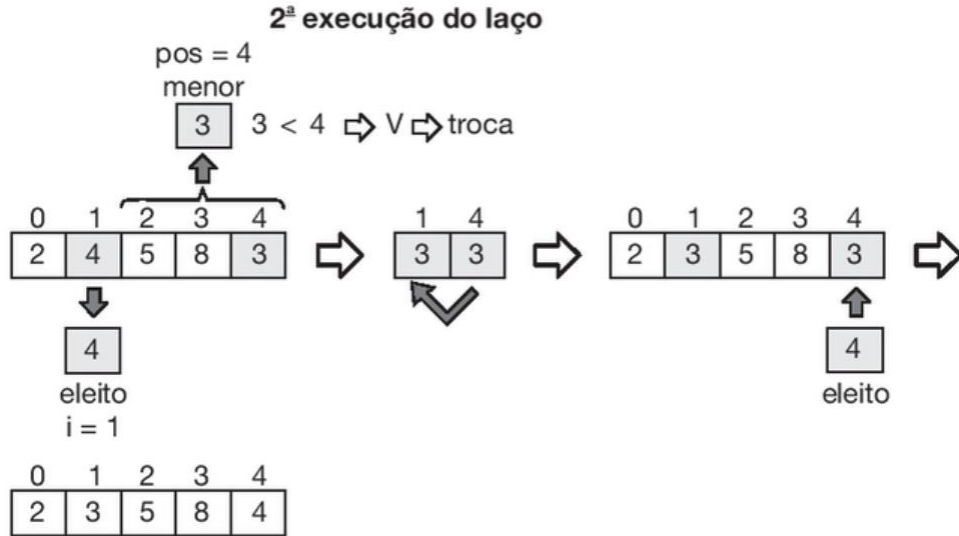
- Funcionamento (cont.)
 - ▶ Quando satisfaz as condições de ordenação, este trocará de posição com o eleito.
 - ▶ Todos à esquerda do eleito estarão ordenados.
 - ▶ Laços

```
for(i=0; i<n-1; i++)  
  for(j=i+2; j<=n-1; j++)
```
 - ▶ O número eleito é i e à direita são $i+1$ a $n-1$.

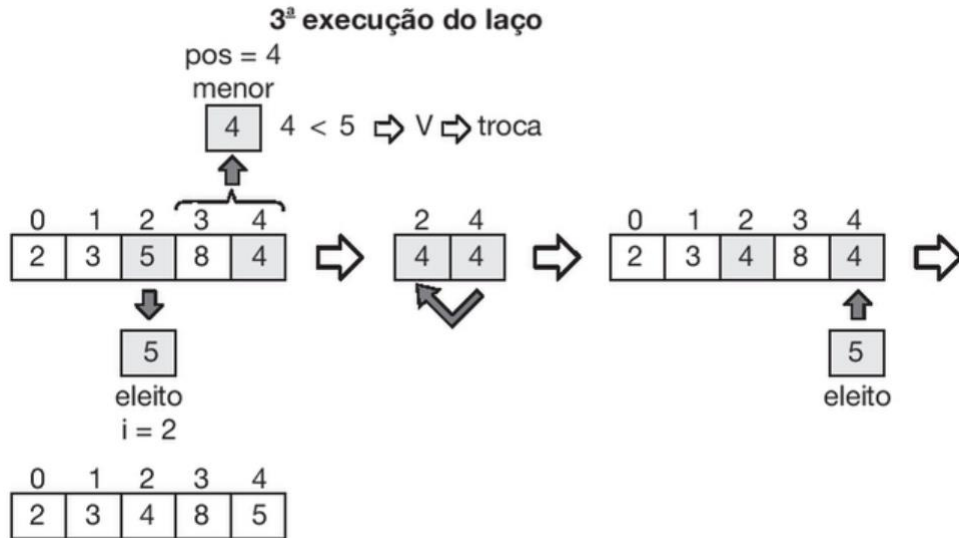
Selection Sort: exemplo



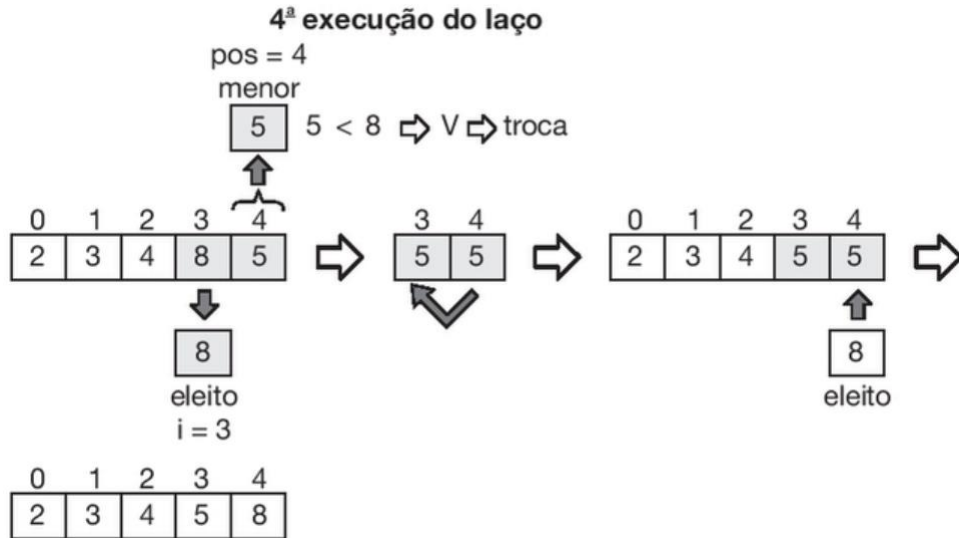
Selection Sort: exemplo



Selection Sort: exemplo



Selection Sort: exemplo



Selection Sort: código

```
#include <iostream.h>
#include <conio.h>

void main ()
{
    int X[5];
    int i, j, eleito, menor, pos;
    clrscr();
    // carregando os números no vetor
    for(i=0;i<=4;i++)
    {
        cout<<"Digite o "<<i+1<<"º número: ";
        cin>>X[i];
    }
    // ordenando de forma crescente
    // laço que percorre da 1ª posição
    // à penúltima posição do vetor
    // elegendo um número para ser comparado
    for(i=0;i<=3;i++)
    {
        eleito = X[i];
        // encontrando o menor número à direita do eleito
        // com sua respectiva posição
```

```
        // posição do eleito = i
        // primeiro número à direita do
        // eleito na posição = i + 1
        menor = X[i+1];
        pos = i + 1;
        // laço que percorre os elementos
        // que estão à direita do
        // número eleito, retornando o
        // menor número à direita
        // e sua posição
        for (j=i+1;j<=4;j++)
        {
            if (X[j] < menor)
            {
                menor = X[j];
                pos = j;
            }
        }
        // troca do número eleito com o número da posição pos
        // o número da posição pos é o menor número à direita
        // do número eleito
```


Selection Sort: código (cont.)

```
        if (menor < eleito)
        {
            X[i] = X[pos];
            X[pos] = eleito;
        }
    }
    // mostrando o vetor ordenado
    for(i=0; i<=4; i++)
    {
        cout<<"\n"<<i+1<<"º número: "<<X[i];
    }
    getch();
}
```

Perguntas?

Bibliografia da aula

- ASCENCIO, A. F. G.; ARAÚJO, G. S. Estrutura de dados. Algoritmos, análise da complexidade e implementação em Java e C/C++. Pearson. 2010.