

# **MÉTODOS DE ORDENAÇÃO DE DADOS EM MEMÓRIA PRIMÁRIA**

Material da Prof. Ana Eliza

# Métodos de Ordenação em Memória Primária

- Métodos Elementares
  - Classificação por Inserção
    - Método da Inserção Direta
  - Classificação por Seleção
    - Método da Seleção Direta
  - Classificação por Trocas
    - Método da Bolha - Bubblesort

# Ordenação por Inserção

- Definição

- Este método consiste em realizar a ordenação pela inserção de cada um dos elementos em sua posição correta, levando em consideração os elementos já ordenados.

# Ordenação por Inserção

- Método da Inserção Direta

- O vetor é dividido em dois segmentos: o primeiro contendo os valores já classificados e o segundo contendo os elementos ainda não ordenados.
- Inicialmente, o primeiro segmento contém apenas o primeiro elemento do vetor e o segundo contém todos os demais elementos do vetor.

# Ordenação por Inserção

- Método de Inserção Direta - Passos

1º) Retira-se o primeiro elemento do segmento não ordenado e coloca-se esse elemento no segmento ordenado na posição correta;

2º) Repete-se o processo até que todos os elementos do segmento não ordenado tenham passado para o segmento ordenado.

# Ordenação por Inserção

- Inserção Direta - Exemplo

Vetor inicial ( 21 | 27 12 20 37 19 17 15 )

Etapa 1 ( 21 27 | 12 20 37 19 17 15 )

Etapa 2 ( 12 21 27 | 20 37 19 17 15 )

Etapa 3 ( 12 20 21 27 | 37 19 17 15 )

Etapa 4 ( 12 20 21 27 37 | 19 17 15 )

Etapa 5 ( 12 19 20 21 27 37 | 17 15 )

Etapa 6 ( 12 17 19 20 21 27 37 | 15 )

Etapa Final ( 12 15 17 19 20 21 27 37 | )

# Ordenação por Inserção

- Inserção Direta - Exemplo

```
void insercaoDireta (float v [ ], int n) {  
    int i,j;  
    float chave;  
  
    for (i = 1; i <= n - 1; i++) {  
        chave = v [i];  
        j = i - 1;  
        while (j >= 0 && v[j] > chave) {  
            v[j+1] = v[j];  
            j = j - 1;  
        }  
        v[j+1] = chave;  
    }  
}
```

# Ordenação por Seleção

- Definição

- Este processo de classificação consiste em uma seleção sucessiva do menor ou do maior valor contido no vetor, dependendo se a classificação dos elementos será em ordem crescente ou decrescente.



# Ordenação por Seleção

- Método da Seleção Direta

- O vetor é dividido em dois segmentos: o primeiro contendo os valores já classificados e o segundo contendo os elementos ainda não selecionados.
- Inicialmente, o primeiro segmento está vazio e o segundo contém todos os elementos do vetor.

# Ordenação por Seleção

- Método da Seleção Direta - Passos

1º) É feita uma busca no segmento que contém os elementos ainda não selecionados, identificando o elemento de menor valor;

2º) É realizada a troca do elemento identificado no passo anterior com o 1º elemento do segmento não ordenado;

3º) O tamanho do segmento que contém os elementos ainda não selecionados é atualizado, ou seja, subtrai-se um de seu tamanho;

4º) O processo é repetido até que o segmento fique com apenas um elemento, que é o maior valor do vetor.

# Ordenação por Seleção

- Método da Seleção Direta - Exemplo

Vetor Inicial	(   21 27 <b>12</b> 20 37 19 17 15 )	TAM = 8
Etapa 1	( <b>12</b>   27 21 20 37 19 17 <b>15</b> )	TAM = 7
Etapa 2	( <b>12 15</b>   21 20 37 19 <b>17</b> 27 )	TAM = 6
Etapa 3	( <b>12 15 17</b>   20 37 <b>19</b> 21 27 )	TAM = 5
Etapa 4	( <b>12 15 17 19</b>   37 <b>20</b> 21 27 )	TAM = 4
Etapa 5	( <b>12 15 17 19 20</b>   37 <b>21</b> 27 )	TAM = 3
Etapa 6	( <b>12 15 17 19 20 21</b>   37 <b>27</b> )	TAM = 2
Etapa Final	( <b>12 15 17 19 20 21 27</b>   37 )	TAM = 1

# Ordenação por Seleção

- Método da Seleção Direta - Exemplo

```
void selecaoDireta (float v [ ], int n) {  
    int i, j, menor;  
    float aux;  
  
    for (i = 0; i < n - 1; i++) {  
        menor = i;  
        for (j = i + 1; j <= n - 1 ; j++) {  
            if (v[j] < v[menor]) {  
                menor = j;  
            }  
        }  
        aux = v [i];  
        v [i] = v [menor];  
        v [menor] = aux;  
    }  
}
```

# Ordenação por Trocas

- Definição

- Este processo de classificação consiste na comparação de pares de chaves de ordenação, trocando os elementos correspondentes caso estejam fora de ordem.

# Ordenação por Trocas

- Método da Bolha - Bubblesort

– Neste método, o princípio geral da classificação por trocas é aplicado a todos os ***pares consecutivos*** de chaves não ordenados. Quando não restarem mais pares não ordenados, o vetor estará classificado.

# Ordenação por Trocas

- Método da Bolha - Passos

1º) Em cada passo, cada elemento é comparado com seu próximo;

2º) Se o elemento estiver fora de ordem a troca é realizada;

3º) Realizam-se tantos passos quanto forem necessários até que não ocorram mais trocas.

# Ordenação por Trocas

- Método da Bolha - Exemplo

Vetor inicial (28 26 30 24 25)

Primeira Varredura:

(28 26 30 24 25) compara (28,26): troca.

(26 28 30 24 25) compara (28,30): não troca.

(26 28 30 24 25) compara (30,24): troca.

(26 28 24 30 25) compara (30,25): troca.

(26 28 24 25 **30**) fim da primeira varredura.



# Ordenação por Trocas

- Método da Bolha - Comentários

- O processo de comparação dos  $n - 1$  pares de chaves é denominado *varredura*.
- Cada varredura sempre irá posicionar a chave de maior valor em sua posição correta definitiva (no final do vetor).
- Isso significa que a cada nova varredura podemos desconsiderar a última posição do vetor, que fica reduzido de um elemento.

# Ordenação por Trocas

- Método da Bolha - Exemplo (cont.)

Vetor inicial (26 28 24 25 | 30)

Segunda Varredura:

(26 28 24 25 | 30) compara (26,28): não troca.

(26 28 24 25 | 30) compara (28,24): troca.

(26 24 28 25 | 30) compara (28,25): troca.

(26 24 25 28 | 30) fim da segunda varredura.

# Ordenação por Trocas

- Método da Bolha - Exemplo (cont.)

Vetor inicial (26 24 25 | **28 30**)

Terceira Varredura:

(26 24 25 | **28 30**) compara (26,24): troca.

(24 26 25 | **28 30**) compara (26,25): troca.

(24 25 **26** | **28 30**) fim da terceira varredura.

# Ordenação por Trocas

- Método da Bolha - Exemplo (cont.)

Vetor inicial (24 25 | **26 28 30**)

Quarta Varredura:

(24 25 | **26 28 30**) compara (24,25): não troca.

(24 **25** | **26 28 30**) fim da quarta varredura.

# Ordenação por Trocas

- Método da Bolha - Algoritmo

```
void bubblesort (float v [ ], int n) {  
    int i, fim, pos, troca;  
    float chave;  
  
    troca = 1; fim = n - 2; pos = 0;  
    while (troca == 1) {  
        troca = 0;  
        for (i = 0; i <= fim; i++) {  
            if (v[i] > v[i+1]) {  
                chave = v[i];  v[i] = v[i+1];  v[i+1] = chave;  
                pos = i; troca = 1;  
            }  
        }  
        fim = pos-1;  
    }  
}
```

# Ordenação por Trocas

- Método da Bolha - Comentários

- A variável POS guarda a posição onde foi realizada a última troca da varredura. A partir dessa posição todos os elementos já estão ordenados e podem ser ignorados na próxima varredura.