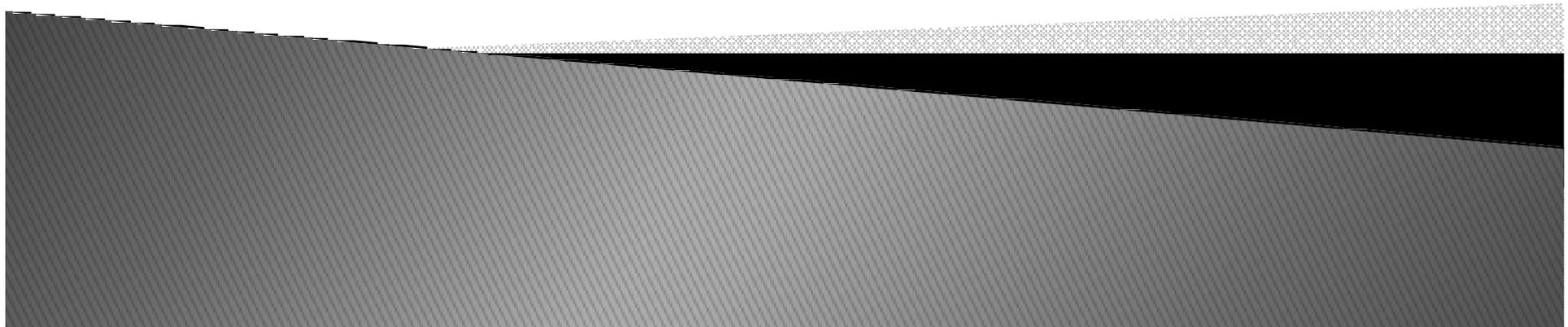


Métodos de Classificação em memória primária

Gustavo Callou
gcallou@gmail.com



Métodos de Classificação em Memória Primária

▶ Métodos Elementares

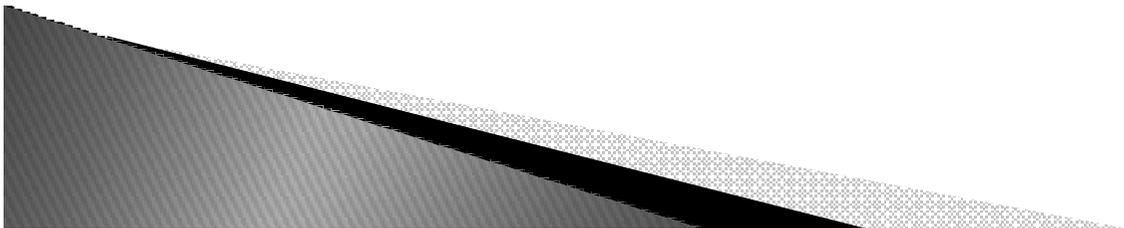
- Classificação por Trocas
 - Método da Bolha – Bubblesort
 - Método de Partição e Troca – Quicksort
- Classificação por Inserção
 - Método da Inserção Direta
 - Método dos Incrementos Decrescentes – Shellsort
- Classificação por Seleção
 - Método da Seleção Direta
- Classificação por Intercalação
 - Método da Intercalação Simples – MergeSort



Classificação por Trocas

▶ Definição

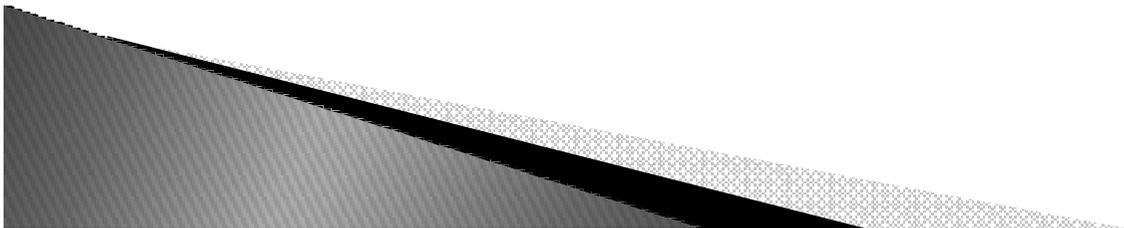
- Este processo de classificação consiste na comparação de pares de chaves de ordenação, trocando os elementos correspondentes caso estejam fora de ordem.



Classificação por Trocas

▶ Método da Bollha – Bubblesort

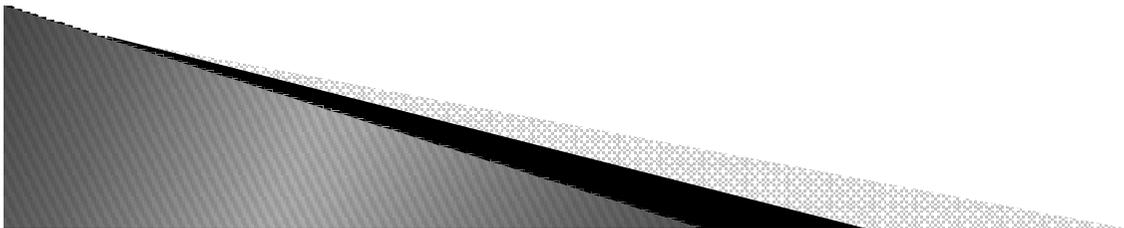
- Nesse método, o princípio geral da classificação por trocas é aplicado a todos os *pares consecutivos* de chaves não ordenados.
- Quando não restarem mais pares não ordenados, o vetor estará classificado.



Classificação por Trocas

▶ Bubblesort – Algoritmo

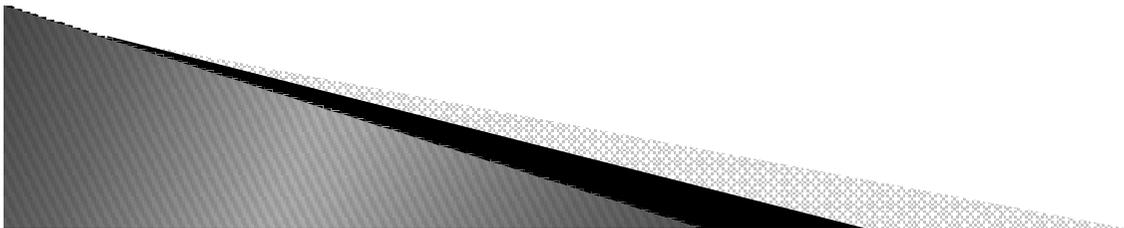
1. Em cada passo, cada elemento é comparado com seu próximo.
2. Se o elemento estiver fora de ordem a troca é realizada.
3. Realizam-se tantos passos quanto forem necessários até que não ocorram mais trocas.



Classificação por Trocas

▶ Bubblesort – Exemplo

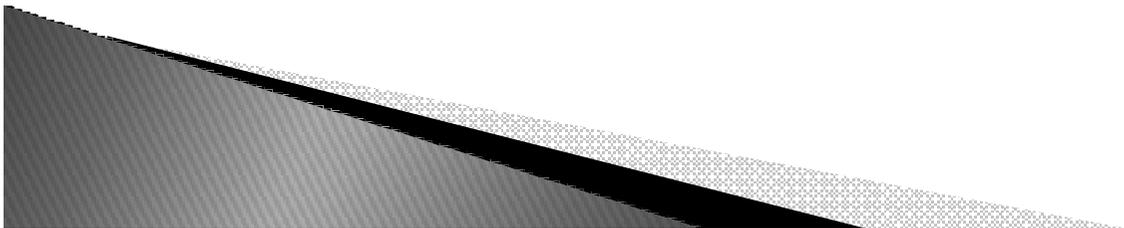
- Vetor inicial (28 26 30 24 25)
- Primeira Varredura:
 - (28 26 30 24 25) compara (28,26): troca.
 - (26 28 30 24 25) compara (28,30): não troca.
 - (26 28 30 24 25) compara (30,24): troca.
 - (26 28 24 30 25) compara (30,25): troca.
 - (26 28 24 25 30) fim da primeira varredura.



Classificação por Trocas

▶ Bubblesort – comentários

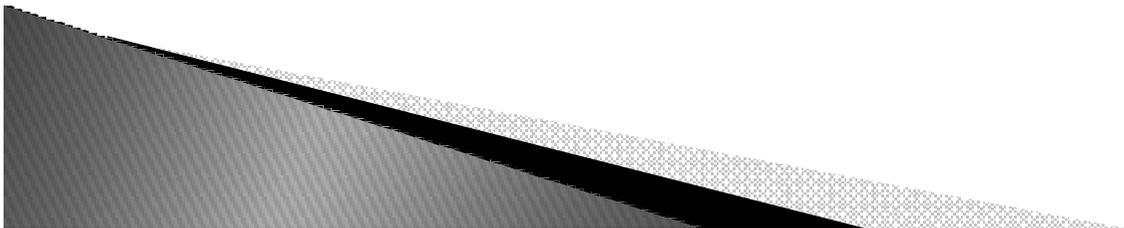
- O processo de comparação dos $n-1$ pares de chaves é denominado varredura.
- Cada varredura sempre irá posicionar a chave de maior valor em sua posição correta, definitiva (no final do vetor).
- Isso significa que a cada nova varredura podemos desconsiderar a última posição do vetor, que ficará reduzido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



Classificação por Trocas

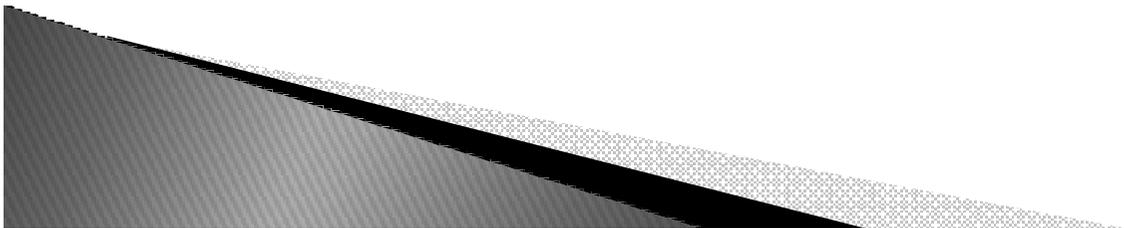
▶ Bubblesort – Exemplo(continuação)

- Vetor inicial (26 28 24 25 130)
- Segunda Varredura:
 - (26 28 24 25 130) compara (26,28): não troca.
 - (26 28 24 25 130) compara (28,24): troca.
 - (26 24 28 25 130) compara (28,25): troca.
 - (26 24 25 128 30) fim da segunda varredura.



Classificação por Trocas

- ▶ Bubblesort – Exemplo(continuação)
 - Vetor inicial (26 24 25 128 30)
 - Terceira Varredura:
 - (26 24 25 128 30) compara (26,24): troca.
 - (24 26 25 128 30) compara (26,25): troca.
 - (24 25 126 28 30) fim da terceira varredura.



Classificação por Trocas

Procedimento bubblesort(var v:vet, n:inteiro)

Variaveis i, fim, pos: inteiro

troca: logico

chave: <tipo_componente>

Inicio

troca=verdade

fim=n-1

pos=1

enquanto troca=verdade faça

troca=falso

para l de 1 ate fim faça

se v[i]>v[i+1] entao

chave = v[i]

v[i]=v[i+1]

v[i+1]=chave

pos=l

troca=verdade

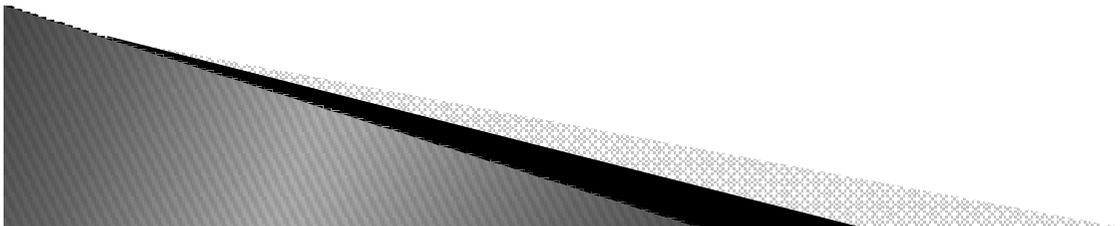
fim se

fim para

fim=pos-1

fim enquanto

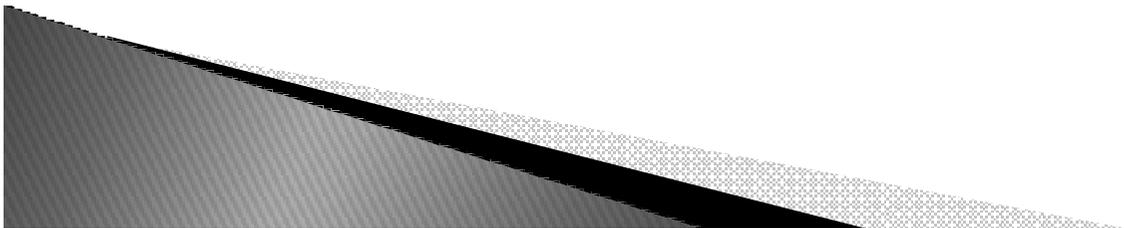
fim



Classificação por Trocas

▶ Bubblesort – Comentários

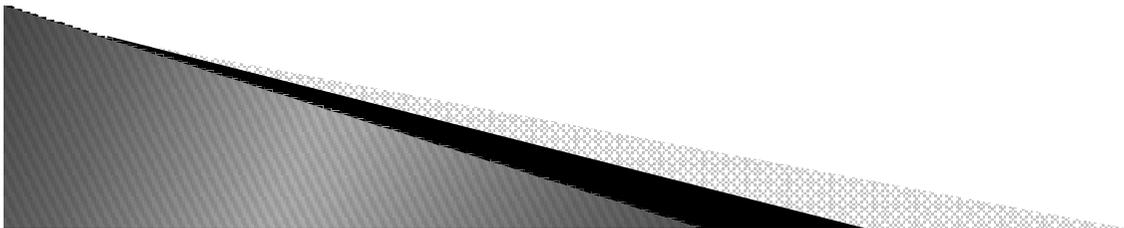
- A variável POS guarda a posição onde foi realizada a última troca da varredura.
- A partir dessa posição, os elementos já se encontram ordenados e podem ser ignorados na próxima varredura.



Classificação por Trocas

▶ Análise do Bubblesort

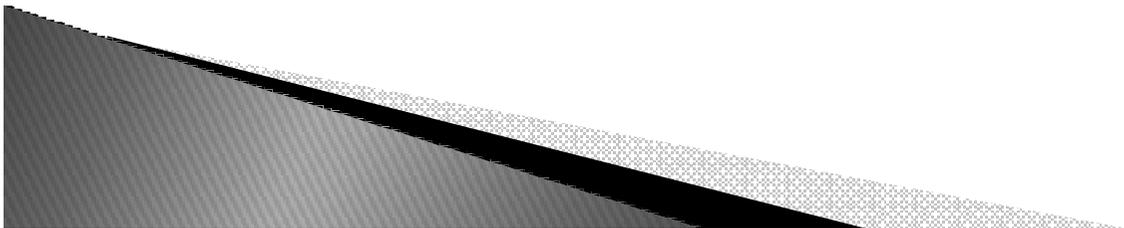
- Melhor caso (o vetor já está ordenado):
 - Ao final da primeira varredura, o método já terá descoberto que nenhuma troca foi realizada e, portanto, o vetor se encontra ordenado.
 - Esta primeira e única varredura demanda $n-1$ comparações



Classificação por Trocas

- ▶ **Pior caso (vetor inversamente ordenado)**
 - Nesse caso, a cada varredura, apenas uma chave será posicionada em seu local definitivo.
 - O total de comparações necessários para a ordenação do vetor, nesse caso, será a soma da seguinte progressão aritmética:

$$(n-1) + (n-2) + \dots + 2 + 1 = (n^2 - n) / 2$$



Classificação por Trocas

- ▶ Desempenho Médio do Bubblesort (casos normais):
 - Corresponde a média do desempenho nos casos extremos:
 - $((n-1) + (n^2-n)/2) / 2 = (n^2 + n - 2) / 4 = O(n^2)$
 - O desempenho médio é da ordem de n^2 , ou seja, é proporcional ao quadrado do número de elementos do vetor.
 - Este método não é indicado para vetores com muito elementos.

