

Classificação e Pesquisa de Dados

Aula 06

Classificação de dados por Troca: *Quicksort*

UFRGS

INF01124

Instituto de Informática - UFRGS

Classificação por Trocas

Classificação por comparação entre pares de chaves, trocando-as de posição caso estejam fora de ordem no par. Novamente iniciaremos com uma solução imediata e, posteriormente, desenvolveremos uma solução rápida e elegante.

- Principais Algoritmos

- *Bubblesort*
- *Quicksort*

Instituto de Informática - UFRGS

Quicksort - Método da Partição e Troca

- ◆ Comparado com os demais métodos, apresenta, em média, o **menor tempo de classificação**.
- ◆ Baseia-se numa técnica fundamental para solução de problemas conhecida como “**Dividir para Conquistar**”.
- ◆ O problema de ordenar um vetor de n elementos é subdividido em dois problemas **menores e independentes**, cujos resultados podem ser facilmente combinados.
- ◆ O processo de subdivisão/cominação é aplicado **recursivamente**.

Instituto de Informática - UFRGS

Quicksort - Princípio de Classificação

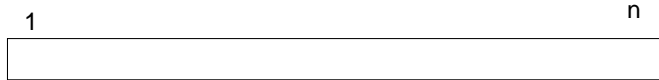
- Inicialmente, o vetor de chaves C é particionado em três segmentos S_1 , S_2 e S_3
- S_2 conterá apenas uma chave denominada **particionadora** ou **pivô**
- S_1 conterá todas as chaves cujos valores são **menores ou iguais ao pivô**. Esse segmento está posicionado à esquerda de S_2
- S_3 conterá todas as chaves cujos valores são **maiores do que o pivô**. Esse segmento está posicionado à direita de S_2

Instituto de Informática - UFRGS

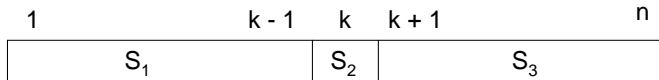
Quicksort

◆ Esquema conceitual do particionamento

Vetor Inicial : $C[1..n]$



Vetor Particionado



Onde: $C[i] \leq C[k]$, para $i = 1, \dots, k-1$
 $C[i] > C[k]$, para $i = k+1, \dots, n$

Instituto de Informática - UFRGS

Quicksort (Cont.)

- O particionamento é reaplicado aos segmentos S_1 e S_3 e a todos os segmentos correspondentes daí resultantes com comprimento ≥ 1
- Quando não restarem segmentos a serem particionados, o vetor estará ordenado
- Perguntas:
 - Qual é a chave particionadora ideal?
 - Como escolher essa chave ?

Instituto de Informática - UFRGS

Quicksort - Escolha do pivô

- ◆ A chave particionadora **ideal** é aquela que produz segmentos S_1 e S_3 com tamanhos (aproximadamente) iguais: **chave de valor mediano**
- ◆ A identificação do pivô ideal requer a varredura de todo o vetor (o benefício não justifica o custo)
- ◆ Deseja-se um critério de escolha simples e rápido
- ◆ Sem conhecimento prévio sobre a distribuição de valores das chaves, supõe-se que qualquer uma pode ser o pivô e arbitra-se a primeira chave
- ◆ Caso o vetor já se encontre parcialmente ordenado, pode-se utilizar o elemento médio

Instituto de Informática - UFRGS

Quicksort - Particionamento

- ◆ O algoritmo executa o particionamento em n passos (n = número de chaves)
- ◆ Nos primeiros $n-1$ passos:
 - ◆ Chaves menores ou iguais ao pivô são deslocadas para o lado esquerdo do vetor
 - ◆ Chaves maiores que o pivô são deslocadas para o lado direito do vetor
- ◆ No último passo o pivô é inserido em sua posição definitiva

Instituto de Informática - UFRGS

Quicksort - Exemplo

Vetor Original: [9 25 10 18 5 7 15 3]

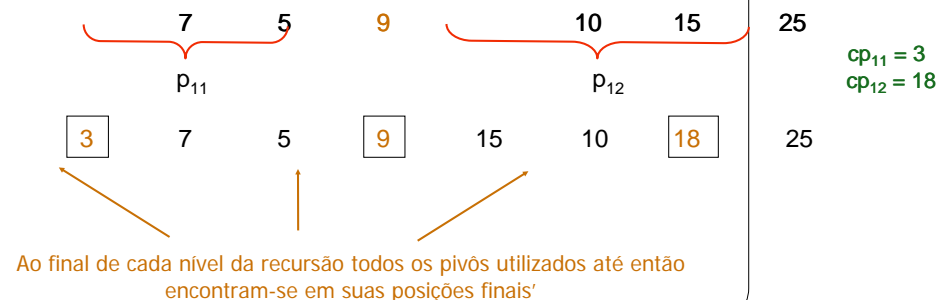
Particionamento no primeiro nível da recursão: **pivô = 9**

1.		25	10	18	5	7	15	3	esquerda
2.	i							f	direita
3.	3	25	10	18	5	7	15	f	esquerda
4.	3	i	10	18	5	7	15	f	esquerda
5.	3	i	10	18	5	7	15	f	esquerda
6.	3	7	10	18	5		15	f	direita
7.	3	7	i	18	5	10	15	f	esquerda
8.	3	7	5	18	f	10	15	f	direita
	3	7	5	9	18	10	15		25

Quicksort – Exemplo (Cont.)

Resultado do particionamento no primeiro nível da recursão:

Particionamento no segundo nível da recursão:

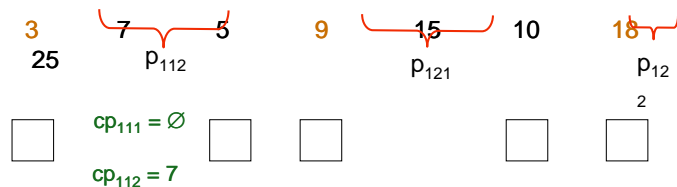


Quicksort – Exemplo (Cont.)

Resultado do particionamento no segundo nível da recursão:

3 7 5 9 15 10 18
25

Particionamento no terceiro nível da recursão:



Quando todas as sequências assumem tamanho 1, o vetor está ordenado!

3 5 7 9 10 15 18
25