



Motivação

- Ideias importantes para o projeto de algoritmos aparecem em algoritmos de ordenação: divisão e conquista e estruturas de dados
- Aplicações: um quarto dos ciclos de processamento dos mainframes é gasto com ordenação [Knut98]
 - Busca
 - Pares próximos
 - Unicidade
 - Distribuição de frequência
- Problema mais estudado em Ciência da Computação: dezenas de algoritmos, cada um dos quais importante em diferentes situações

2

Aplicação: Envoltória Convexa

● Fase 1: ordenar pontos com respeito a x
● Fase 2: percorrer pontos incluindo e removendo elementos da EC

3

Questões Práticas Gerais

- Ordenar chaves ou registros inteiros ?
- Chaves repetidas: a maior parte dos algoritmos não se importa de trocar de lugar elementos com chave igual – algoritmos que preservam a posição de elementos com chaves iguais são ditos *estáveis*

4

O Método da Bolha

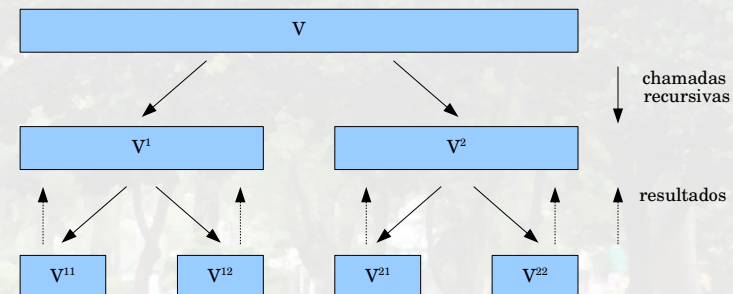
```

função bSort( V[1...n] )
1. trocou = não
2. repita
3.     para  $i = 1$  até  $n-1$  faça
4.         se  $V[i] > V[i+1]$ 
5.             troque  $V[i]$  e  $V[i+1]$ 
6.             trocou = sim
7.         fim se
8.     fim para
9. enquanto trocou
fim

```

até n^2 operações !

Ordenação - Divisão e Conquista



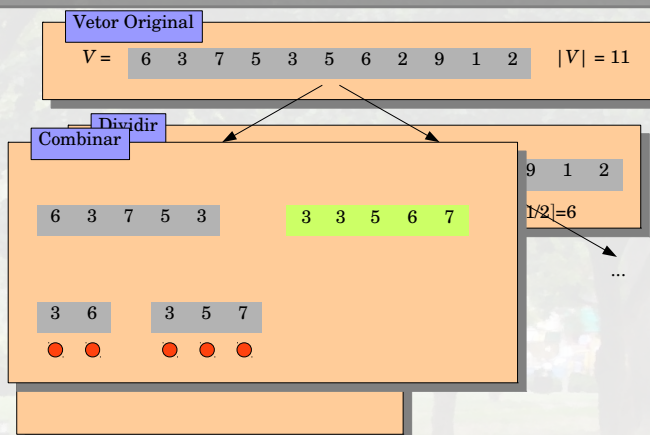
Divisão e Conquista

```

função divConq( $x$ )
1. se  $x$  é suficientemente pequeno então
2.   retorne resolver( $x$ )
3. senão
4.   decomponha  $x$  em  $k$  conjuntos menores:  $x^1, x^2, \dots, x^k$ 
5.   para  $i$  de 1 até  $k$ 
6.      $y_i = \text{divConq}(x_i)$ 
7.   fim
8.   combine os resultados de  $y_1, y_2, \dots, y_k$  em  $y$ 
9.   retorne  $y$ 
fim

```

MergeSort



MergeSort

Abordagem Top-Down

MergeSort($V[1, n]$)
Merge(**MergeSort**($V[1, \lfloor n/2 \rfloor]$), **MergeSort**($V[\lfloor n/2 \rfloor + 1, n]$))

Caso base: vetor com 1 elemento

9

Custo Computacional - Subproblemas

n elementos								0
$n/2$		$n/2$						1
$n/4$	$n/4$	$n/4$	$n/4$	$n/4$	$n/4$	$n/4$	$n/4$	2
$n/8$	$n/8$	$n/8$	$n/8$	$n/8$	$n/8$	$n/8$	$n/8$	3
$n/(2^k)$	$n/(2^k)$	$n/(2^k)$	$n/(2^k)$	$n/(2^k)$	$n/(2^k)$	$n/(2^k)$	$n/(2^k)$	k

Qual o maior valor de k tal que $2^k \leq n$?

10

Outros Algoritmos de Ordenação

- Insertion Sort
- HeapSort
- QuickSort
- Radix Sort
- IntroSort (híbrido)

Escolhendo o melhor para sua aplicação:

Considere:

- requerimento adicional de memória
- número médio e máximo de elementos envolvidos
- dados randômicos ou *quase* ordenados
- recursivo/não recursivo
- facilidade de paralelização
- localidade de memória
- hardware

Experimentos !!!

11

Exercícios

- Qual a profundidade máxima da recursão no MergeSort (o valor máximo de k) ?
- Qual o número de operações requerido para o merge de dois vetores de tamanho n ?
- Mostre um caso onde o MergeSort não oferece uma ordenação estável.
- A implementação anterior é no estilo *Top-Down*, pois começa da solução completa e progressivamente vai gerando subproblemas usando recursão. Escreva o pseudocódigo de uma implementação *Bottom-Up*, onde o trabalho inicie nos menores subproblemas e progressivamente crie problemas maiores, sem utilizar recursão.

12