

Projeto e Análise de Algoritmos

José Ricardo Mello Viana

Aula 03
Conceitos Básicos

2010.2

- 1 Introdução
- 2 Ordenação por inserção
 - Loops invariantes
- 3 Convenções de pseudocódigo

- Familiarização com as estruturas usadas no restante do livro
- Resolveremos o problema de ordenação: **por inserção**
- Apresentaremos o **pseudocódigo** usado
- Analisaremos o tempo de execução
 - Modo como o tempo aumenta com o número de itens
- Dividir para conquistar: **ordenação por intercalação**

- Entrada: sequência de n números (a_1, a_2, \dots, a_n)
- Saída: permutação da entrada $(a'_1, a'_2, \dots, a'_n)$, tal que $a'_1 \leq a'_2 \leq \dots \leq a'_n$
- **Chaves:** números a ordenar

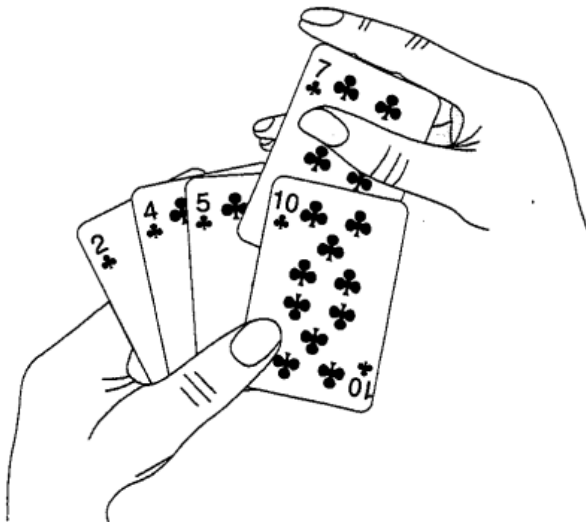


Figure: Ordenando cartas por inserção

- Começamos com a mão vazia
- Tiramos uma carta de cada vez da mesa
- Inserimos na posição correta
 - Comparamos com as cartas que já estão na mão

insertion_sort(A):

for $j \leftarrow 2$ to *comprimento*[A]

do *chave* $\leftarrow A[j]$

//Inserir $A[j]$ na sequência ordenada $A[1..j - 1]$

$i \leftarrow j - 1$

while $i > 0$ e $A[i] > \textit{chave}$

do $A[i + 1] \leftarrow A[i]$

$i \leftarrow i - 1$

$A[i + 1] \leftarrow \textit{chave}$

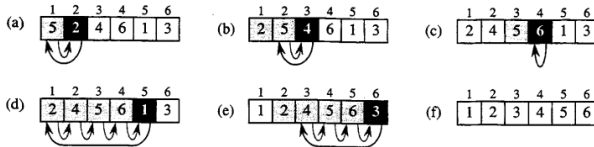


Figure: Funcionamento do insertion sort

- j indica cada carta sendo inserida
- $A[1..j - 1]$: mão atualmente ordenada (**loop invariante**)
- $A[j + 1..n]$: pilha de cartas na mesa

- Usamos um loop invariante para ajudar a provar a corretude
- **Inicialização**: é verdadeiro antes da primeira iteração
- **Manutenção**: se for verdadeira antes de uma iteração, permanecerá na próxima
- **Término**: propriedade de correção do algoritmo

- Nível de indentação indica blocos
- while, for e repeat
- if, then e else
- Comentários: //
- Atribuição múltipla: $a \leftarrow b \leftarrow 10$
- Variáveis sempre locais. Globais apenas com indicação específica
- Arranjos acessados pela posição: $A[j]$
- Atributos de objetos: $comprimento[A]$
- Parâmetros são passados por valor

- Implemente o algoritmo de ordenação por inserção demonstrado nesta aula
Use qualquer linguagem (preferência para C/C++)