



UFOP

BCC702 - Programação de Computadores II

Métodos Simples de Ordenação

Prof^a Valéria de Carvalho Santos



O problema de ordenação

- ❑ A ordenação de dados tem considerável importância para vários problemas
- ❑ Exemplo: localizar um número de telefone em uma agenda. A busca é simplificada se o nomes estão organizados em ordem alfabética

O problema de ordenação

- ❑ Ordenar: organizar uma sequência de elementos de modo que eles estabeleçam alguma relação de ordem

- ❑ Um conjunto de dados ordenados ocasionalmente facilita a busca/localização/recuperação de um elemento

O problema de ordenação

- ❑ Existem alguns métodos de ordenação propostos na literatura

- ❑ Dependendo do problema, um algoritmo apresenta vantagens e/ou desvantagens sobre outros

Métodos simples de ordenação

- ❑ Ordenação por Seleção (Selection Sort)
- ❑ Ordenação por Inserção (Insertion Sort)

Ordenação por Seleção

- ❑ Funcionamento:
 - ❑ Selecione o menor elemento do vetor;
 - ❑ Troque-o com o elemento da primeira posição do vetor;
 - ❑ Repita essas duas operações para os $n-1$ elementos restantes, depois para os $n-2$ elementos restantes, até que reste apenas um elemento.

Exemplo



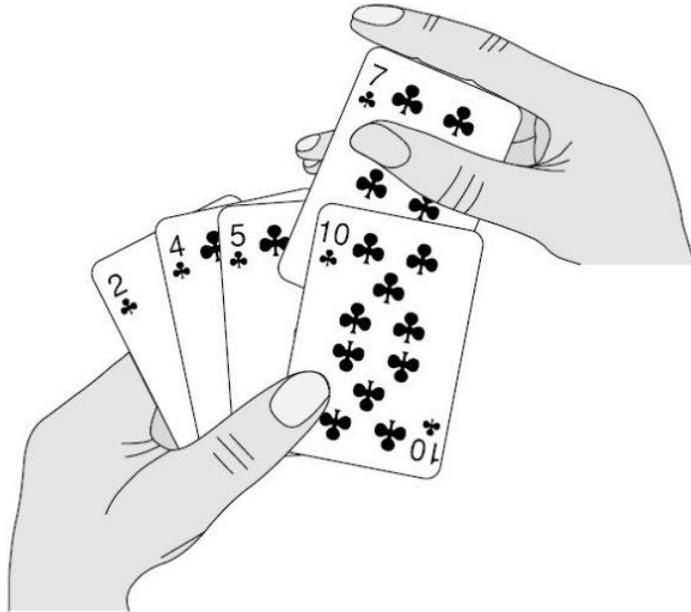
```
void ordena_selecao(int A[], int n){
    int min, aux;
    for(int i=0; i<n; i++){
        min = i;
        for(int j = i+1; j<n; j++){
            if(A[j] < A[min])
                min = j;
        }
        aux = A[i];
        A[i] = A[min];
        A[min] = aux;
    }
}
```

Ordenação por Seleção

- ❑ É um método simples e fácil de implementar
- ❑ É um método interessante para conjuntos pequenos de elementos
- ❑ O fato de o conjunto já estar ordenado não melhora a complexidade do algoritmo

Ordenação por Inserção

- ❑ É o método utilizado por jogadores de carta



Ordenação por Inserção

- ❑ Funcionamento:
 - ❑ Em cada passo, a partir da segunda posição, selecione o i -ésimo elemento do vetor
 - ❑ coloque-o no lugar apropriado na sequência destino, de acordo o critério de ordenação

Exemplo



```
void ordena_insercao(int A[], int n){
    int min, aux, j;
    for(int i=1; i<n; i++){
        aux = A[i];
        j = i-1;
        while(j >= 0 && A[j] > aux){
            A[j+1] = A[j];
            j = j-1;
        }
        A[j+1] = aux;
    }
}
```

Ordenação por Inserção

- ❑ É um bom método a ser utilizado quando:
 - ❑ o vetor está quase ordenado
 - ❑ possui um comportamento ruim quando o conjunto de elementos está bastante desorganizado