# Quicksort

## Análise de Algoritmos – Ciência da Computação

Prof. Daniel Saad Nogueira Nunes

IFB – Instituto Federal de Brasília, Campus Taguatinga

INSTITUTO FEDERAL
Brasília

# Sumário

# Sumário

# Quicksort

## Quicksort

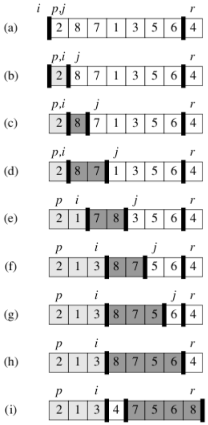
O Quicksort se baseia na escolha de um pivô. Após escolhido este pivô, a sequência original é particionada em três partes:

1. Elementos menores que o pivô;
2. Pivô;
3. Elementos maiores que o pivô;

O procedimento é aplicado recursivamente na primeira e última partes.

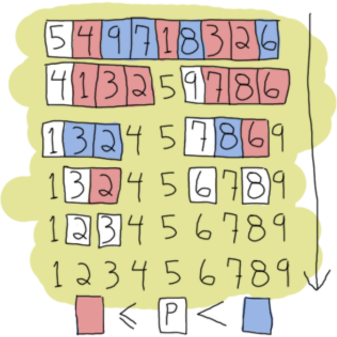
# Quicksort

## Exemplo



# Quicksort

## Exemplo



# Quicksort

**Function** Quicksort

**Input:** $V[0, n-1], i, j$

**Output:** $V, \quad V[i] < V[i+1], 0 \leq i < n-2$

```
if( i < j )
    p ← PARTITION(V, i, j)
    QUICKSORT(V, i, p − 1)
    QUICKSORT(V, p + 1, j)
```

# Quicksort

**Function** Partition

**Input:** $V[0, n-1], i, j$

**Output:** $k$, a posição do pivô após particionar $V$

$k \leftarrow i$; `// Marca posição do término da menor partição`

$l \leftarrow i$

$pivot \leftarrow j$

**while** $l < j$ **do**

  **if**( $V[l] \leq V[pivot]$ )

    $\textsc{Swap}(V[k], V[l])$

    $k++$

  $l++$

$\textsc{Swap}(A[k], A[pivot])$

**return** $k$

# Sumário

# Quicksort: Análise

## Análise

A relação de recorrência do Mergesort, no pior caso, corresponde à:

$$T(n) = T(n-1) + O(n) \in \Theta(n^2)$$

Contanto, no caso médio, o Quicksort divide as partições de modo em que a primeira e a última partição tenham tamanhos similares, o que leva a uma relação de recorrência que se aproxima de:

$$T(n) = 2 \cdot T(n/2) + O(n) \in \Theta(n \lg n)$$

| In-place | Estável |
|---|---|
| ✗ | ✗ |

# Quicksort

## Pior Caso vs Melhor Caso