

# Quick Sort

Murilo Dantas

# Quick Sort

- Funcionamento
  - ▶ O vetor é particionado em 2 de modo recursivo até que o vetor fique com apenas 1 elemento.
  - ▶ É semelhante ao Merge Sort.

# Quick Sort: exemplo

1ª execução do laço

Vetor de 0 a 9

Posição do pivô = parte inteira  $[(0+9)/2] = 4$

0	1	2	3	4	5	6	7	8	9
5	8	3	1	6	2	4	9	7	5

↑  
pivô

5 <= 6 ⇒ V ⇒ para  
↓ j = 9

0	1	2	3	4	5	6	7	8	9
5	8	3	1	6	2	4	9	7	5

↑  
pivô  
↓  
pivô

0	1	2	3	4	5	6	7	8	9
5	8	3	1	6	2	4	9	7	5

↑ i = 0  
5 >= 6 ⇒ F ⇒ continua

pivô  
↓

0	1	2	3	4	5	6	7	8	9
5	8	3	1	6	2	4	9	7	5

↑ i = 1  
8 >= 6 ⇒ V ⇒ para  
↓  
pivô

0	1	2	3	4	5	6	7	8	9
5	8	3	1	6	2	4	9	7	5

i < j ⇒ 1 < 9 ⇒ V ⇒ troca

0	1	2	3	4	5	6	7	8	9
5	5	3	1	6	2	4	9	7	8

7 <= 6 ⇒ F ⇒ continua  
↓ j = 8

0	1	2	3	4	5	6	7	8	9
5	5	3	1	6	2	4	9	7	8

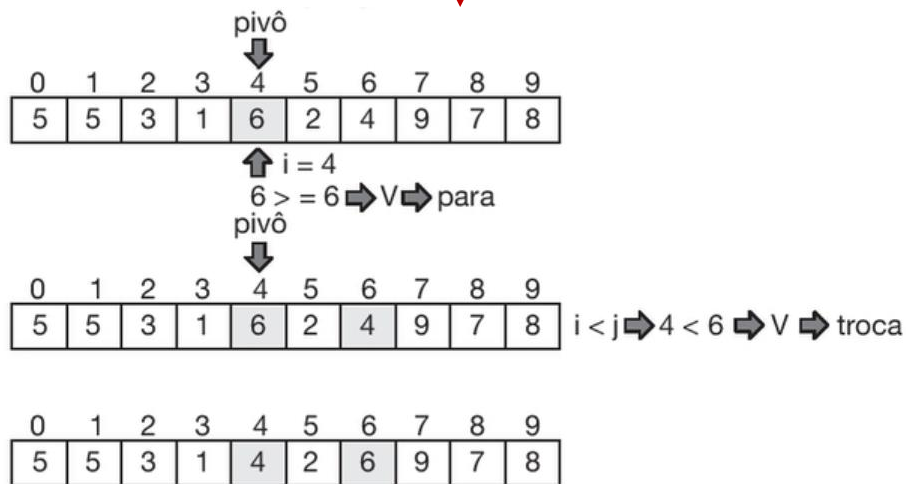
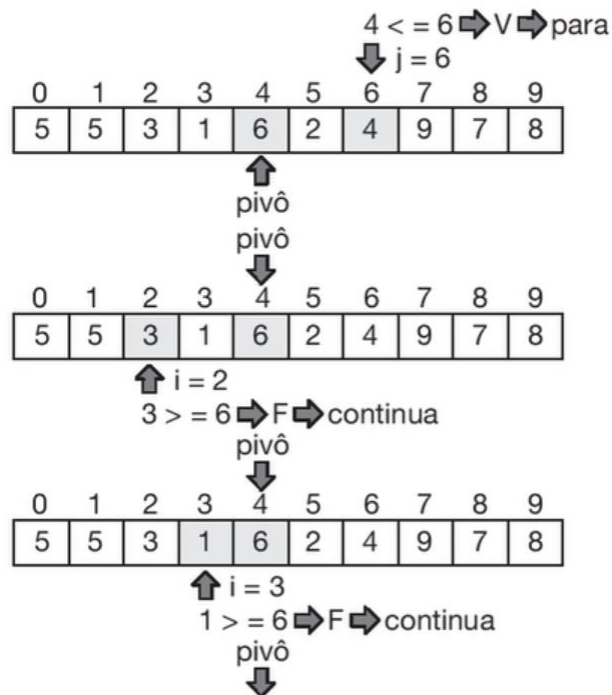
↑  
pivô

9 <= 6 ⇒ F ⇒ continua  
↓ j = 7

0	1	2	3	4	5	6	7	8	9
5	5	3	1	6	2	4	9	7	8

↑  
pivô

# Quick Sort: exemplo



# Quick Sort: exemplo

2ª execução do laço

Vetor de 0 a 5

Posição do pivô = parte inteira  $[(0+5)/2] = 2$

0	1	2	3	4	5
5	5	3	1	4	2

↑  
pivô

$2 \leq 3 \Rightarrow V \Rightarrow \text{para}$

↓  
 $j = 5$

0	1	2	3	4	5
5	5	3	1	4	2

↑  
pivô

0	1	2	3	4	5
5	5	3	1	4	2

↑  
 $i = 0$

$5 \geq 3 \Rightarrow V \Rightarrow \text{para}$

0	1	2	3	4	5
5	5	3	1	4	2

↑  
pivô

$i < j \Rightarrow 0 < 5 \Rightarrow V \Rightarrow \text{troca}$

0	1	2	3	4	5
2	5	3	1	4	5

↑  
pivô

$4 \leq 3 \Rightarrow F \Rightarrow \text{continua}$

↓  
 $j = 4$

0	1	2	3	4	5
2	5	3	1	4	5

↑  
pivô

$1 \leq 3 \Rightarrow V \Rightarrow \text{para}$

↓  
 $j = 3$

0	1	2	3	4	5
2	5	3	1	4	5

↑  
pivô

0	1	2	3	4	5
2	5	3	1	4	5

↑  
 $i = 1$

$5 \geq 3 \Rightarrow V \Rightarrow \text{para}$

0	1	2	3	4	5
2	5	3	1	4	2

$i < j \Rightarrow 1 < 3 \Rightarrow V \Rightarrow \text{troca}$

0	1	2	3	4	5
2	5	3	1	4	2

↑  
pivô

0	1	2	3	4	5
2	1	3	5	4	5

↑  
pivô

# Quick Sort: exemplo

## 3ª execução do laço

Vetor de 0 a 2

Posição do pivô = parte inteira  $[(0+2)/2] = 1$

0	1	2
2	1	3

↑  
pivô

$3 \leq 1 \Rightarrow F \Rightarrow \text{continua}$

↓  $j = 2$

0	1	2
2	1	3

↑  
pivô

$1 \leq 1 \Rightarrow V \Rightarrow \text{para}$

↓  $j = 1$

0	1	2
2	1	3

↑  
pivô

0	1	2
2	1	3

↑  
pivô

$i = 0$

$2 \geq 1 \Rightarrow V \Rightarrow \text{para}$

0	1	2
2	1	3

↑  
pivô

$i < j \Rightarrow 0 < 1 \Rightarrow V \Rightarrow \text{troca}$

0	1	2
1	2	3

↑  
pivô

# Quick Sort: exemplo

4ª execução do laço

Vetor de 1 a 2

Posição do pivô = parte inteira  $[(1+2)/2] = 1$

1	2
2	3

↑  
pivô

$3 < 2 \Rightarrow F \Rightarrow \text{continua}$

↓  $j = 2$

1	2
2	3

↑  
pivô

$2 < 2 \Rightarrow V \Rightarrow \text{para}$

↓  $j = 1$

1	2
2	3

↑  
pivô

1	2
2	3

↖  
↑  
pivô

$i = 1$

$2 \geq 2 \Rightarrow V \Rightarrow \text{para}$

1	2
2	3

↑  
pivô

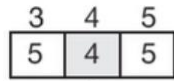
$i < j \Rightarrow 1 < 1 \Rightarrow F \Rightarrow \text{não troca}$

# Quick Sort: exemplo

5ª execução do laço

Vetor de 3 a 5

Posição do pivô = parte inteira  $[(3+5)/2] = 4$



↑  
pivô

$5 \leq 4 \Rightarrow F \Rightarrow \text{continua}$

↓  $j = 5$



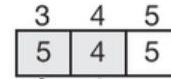
↑  
pivô

$4 \leq 4 \Rightarrow V \Rightarrow \text{para}$

↓  $j = 4$



↑  
pivô



↑  
pivô

$i = 3$

$5 \geq 4 \Rightarrow V \Rightarrow \text{para}$



↑  
pivô

$i < j \Rightarrow 3 < 4 \Rightarrow V \Rightarrow \text{troca}$



↑  
pivô

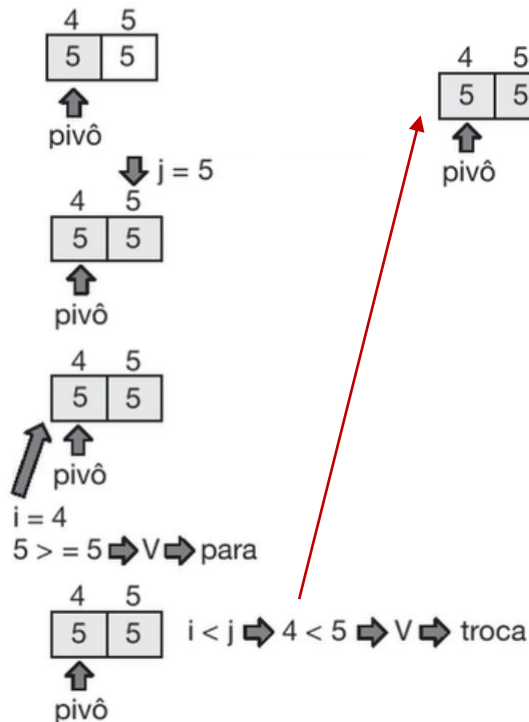


# Quick Sort: exemplo

## 6ª execução do laço

Vetor de 4 a 5

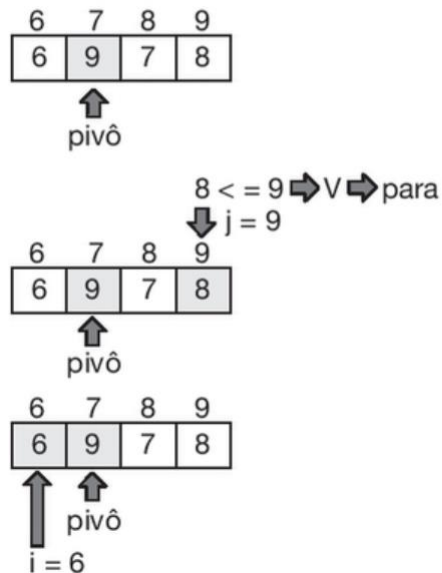
Posição do pivô = parte inteira  $[(4+5)/2] = 4$



## 7ª execução do laço

Vetor de 6 a 9

Posição do pivô = parte inteira  $[(6+9)/2] = 7$



# Quick Sort: exemplo

$6 \geq 9 \Rightarrow F \Rightarrow$  continua

6	7	8	9
6	9	7	8



$i = 7$

$9 \geq 9 \Rightarrow V \Rightarrow$  para

6	7	8	9
6	9	7	8

$i < j \Rightarrow 7 < 9 \Rightarrow V \Rightarrow$  troca

6	7	8	9
6	9	7	8

pivô

6	7	8	9
6	8	7	9

pivô

8ª execução do laço

Vetor de 6 a 8

Posição do pivô = parte inteira  $[(6+8)/2] = 7$

6	7	8
6	8	7

pivô

$7 \leq 8 \Rightarrow V \Rightarrow$  para

$j = 8$

6	7	8
6	8	7

pivô

6	7	8
6	8	7

pivô  
 $i = 6$

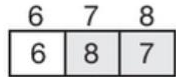
# Quick Sort: exemplo

$6 \geq 8 \Rightarrow F \Rightarrow \text{continua}$

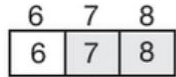


$i = 7$

$8 \geq 8 \Rightarrow V \Rightarrow \text{para}$



$i < j \Rightarrow 7 < 8 \Rightarrow V \Rightarrow \text{troca}$



pivô

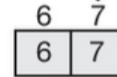
9ª execução do laço  
Vetor de 6 a 7  
Posição do pivô = parte inteira $[(6+7)/2] = 7$



pivô

$7 \leq 6 \Rightarrow F \Rightarrow \text{continua}$

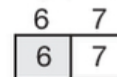
$j = 7$



pivô

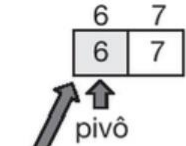
$6 \leq 6 \Rightarrow V \Rightarrow \text{para}$

$j = 6$



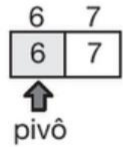
pivô

# Quick Sort: exemplo



$i = 6$

$6 \geq 6 \Rightarrow V \Rightarrow \text{para}$



$i < j \Rightarrow 6 < 6 \Rightarrow F \Rightarrow \text{não troca}$

**Vetor ordenado**

0	1	2	3	4	5	6	7	8	9
1	2	3	4	5	5	6	7	8	9

# Quick Sort: código

```
#include <iostream.h>
```


```
#include <conio.h>
```

```
void troca(int X[], int i, int j)
```

```
{
    int aux;
    aux = X[i];
    X[i] = X[j];
    X[j] = aux;
}
```

```
int particao(int X[],int p,int r)
```

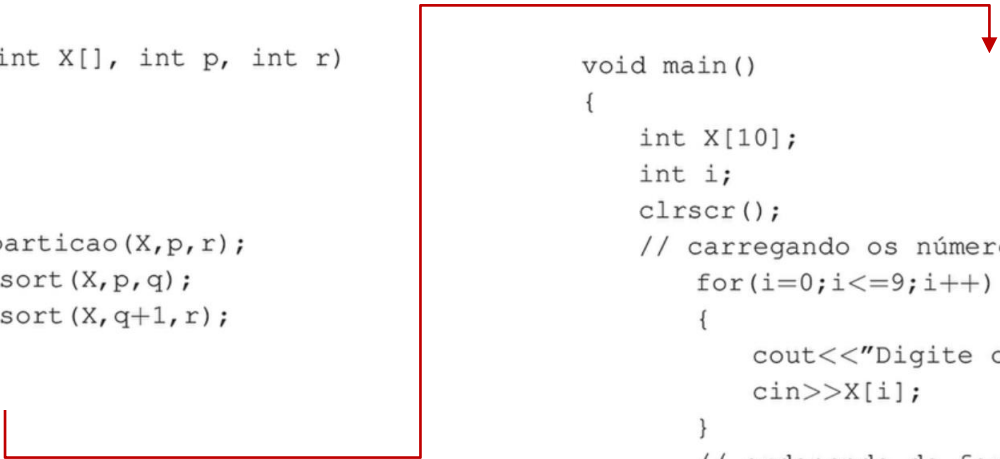
```
{
    int pivo, i, j;
    pivo = X[(p+r)/2];
    i = p-1;
    j = r+1;
    while (i < j)
    {
        do
        {
            j = j - 1;
        }
        while (X[j] > pivo);
    }
}
```



```
do
{
    i = i + 1;
}
while (X[i] < pivo);
if (i < j) troca(X,i,j);
}
return j;
}
```

# Quick Sort: código

```
void quicksort(int X[], int p, int r)
{
    int q;
    if (p < r)
    {
        q = particao(X,p,r);
        quicksort(X,p,q);
        quicksort(X,q+1,r);
    }
}
```



```
void main()
{
    int X[10];
    int i;
    clrscr();
    // carregando os números no vetor
    for(i=0;i<=9;i++)
    {
        cout<<"Digite o "<<i+1<<"º número: ";
        cin>>X[i];
    }
    // ordenando de forma crescente
    quicksort(X,0,9);
    // mostrando o vetor ordenado
    cout<<"Vetor Ordenado";
    for (i=0;i<=9;i++)
    {
        cout<<" "<<X[i];
    }
    getch();
}
```

**Perguntas?**

# Bibliografia da aula

- ASCENCIO, A. F. G.; ARAÚJO, G. S. Estrutura de dados. Algoritmos, análise da complexidade e implementação em Java e C/C++. Pearson. 2010.