



# Quick Sort

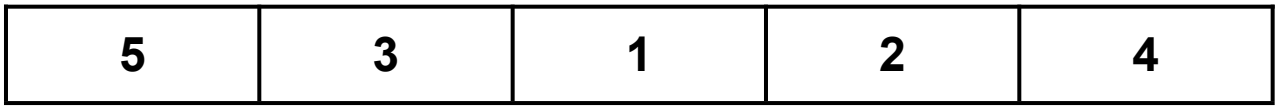
Diogo Oliveira
diogoespinharaoliveira@gmail.com


*A ideia envolve a escolha de um elemento, chamado pivô. Depois o pivô é colocado em sua correta posição na lista, isto é, elementos menores à esquerda e maiores à direita. O mesmo processo é realizado nas sub-listas esquerda e direita.*

# Exemplo

| 5 | 3 | 1 | 2 | 4 |
|---|---|---|---|---|

# Iteração 1




*i, j*

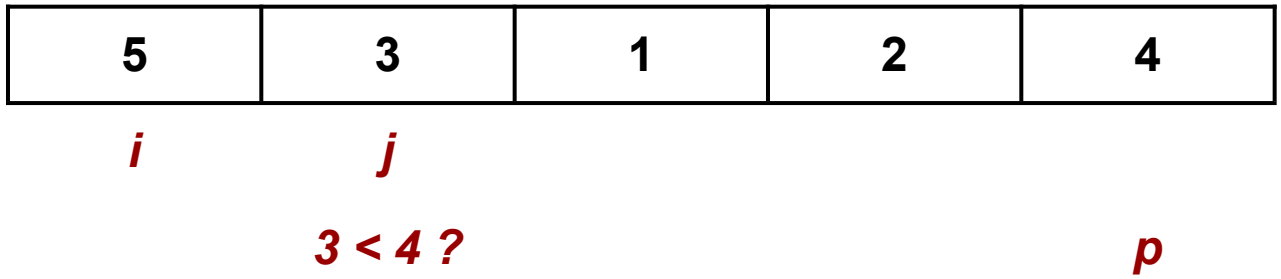
*5 < 4 ?*

*p*

---

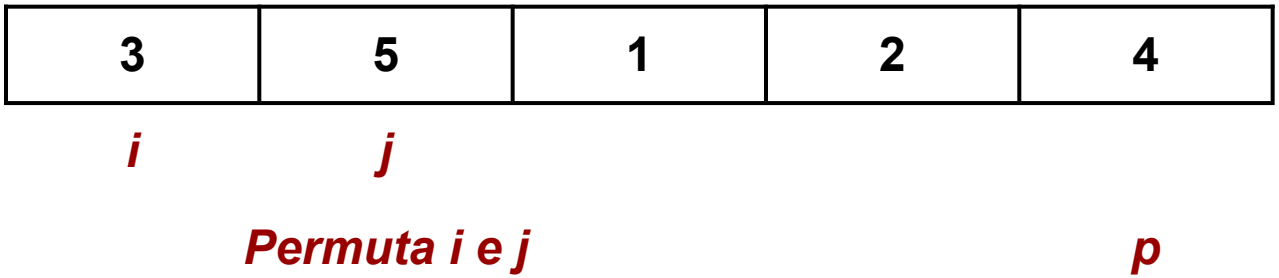
**Árvore**

# Iteração 1



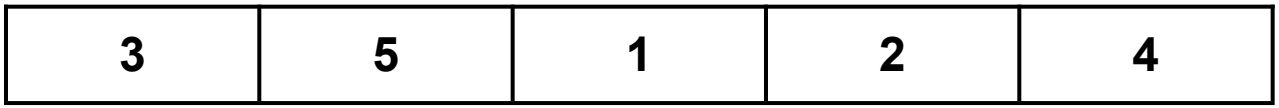

**Árvore**

# Iteração 1




**Árvore**

# Iteração 1




*i, j*

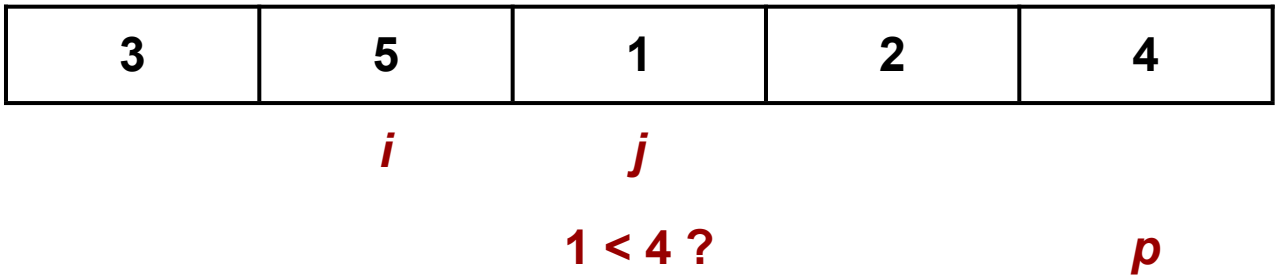
*Incrementa i*

*p*

---

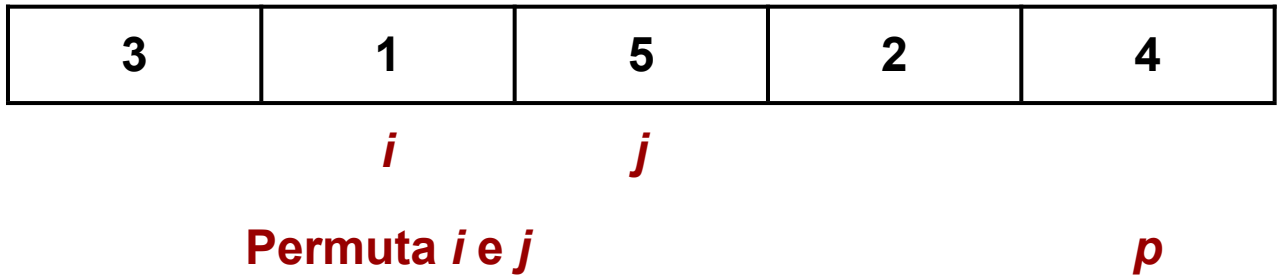
**Árvore**

# Iteração 1




**Árvore**

# Iteração 1




**Árvore**

# Iteração 1


*i, j*

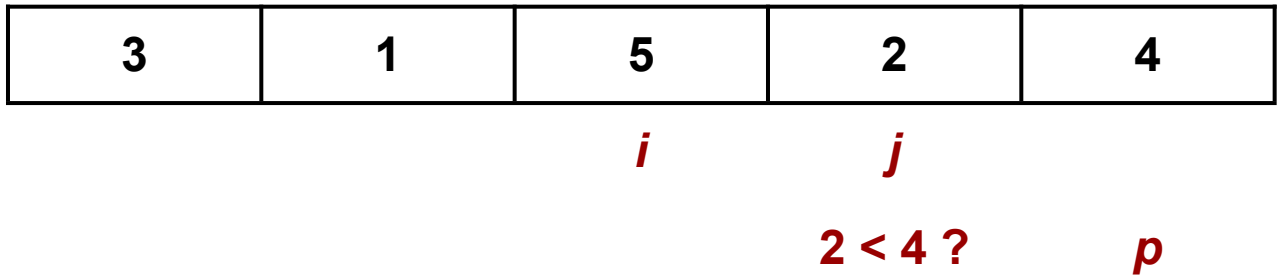
**Incrementa i**

*p*

---

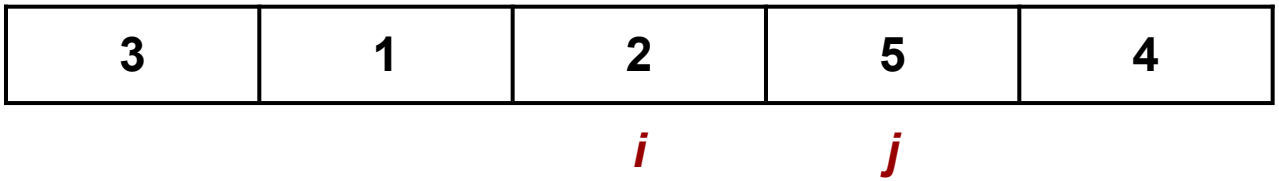
**Árvore**

# Iteração 1




**Árvore**

# Iteração 1



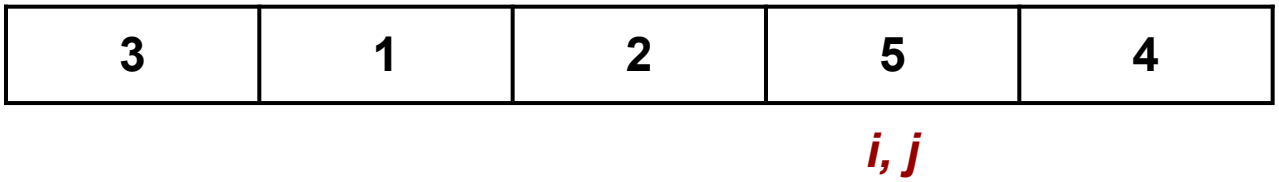

**Permuta *i* e *j***

***p***

---

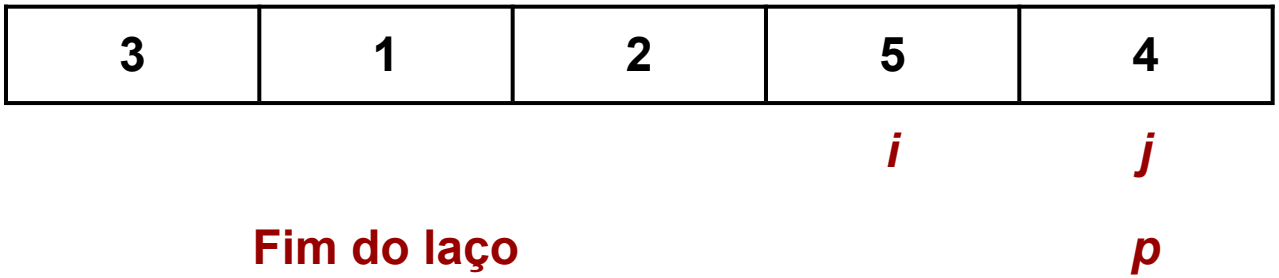
**Árvore**

# Iteração 1



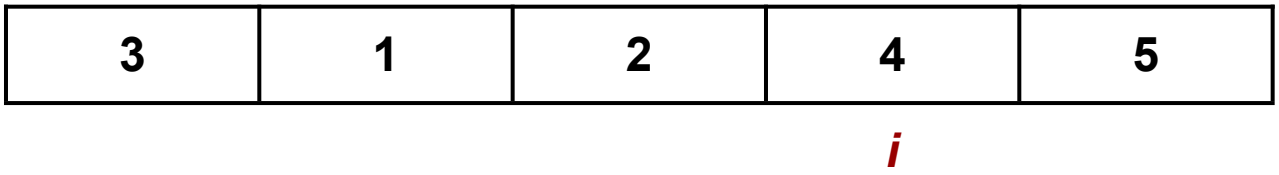

**Árvore**

# Iteração 1



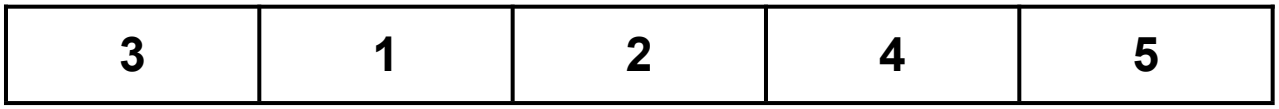

**Árvore**

# Iteração 1




**Árvore**

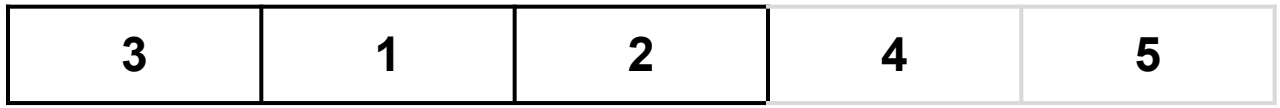
# Iteração 1




p

**Árvore**

# Iteração 2



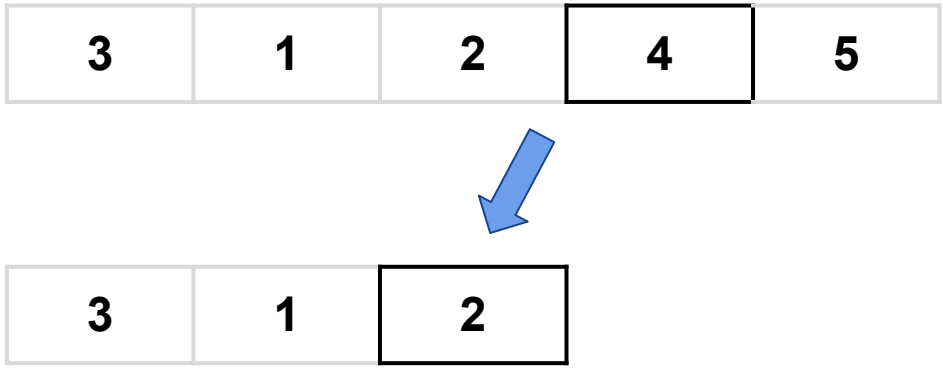

*i, j*

*3 < 2 ?*

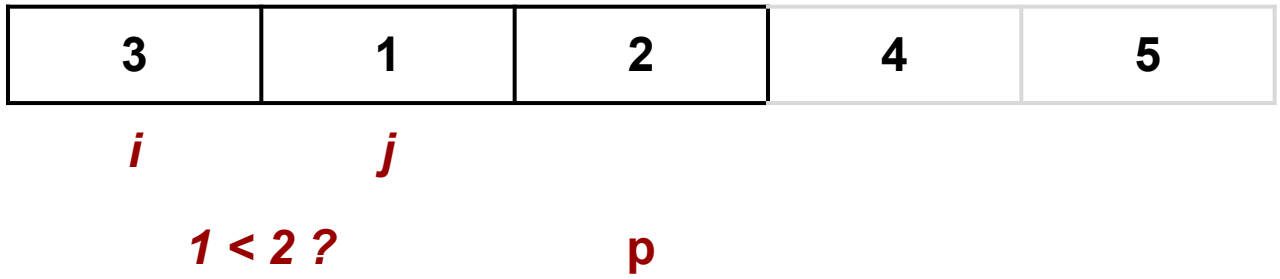
p

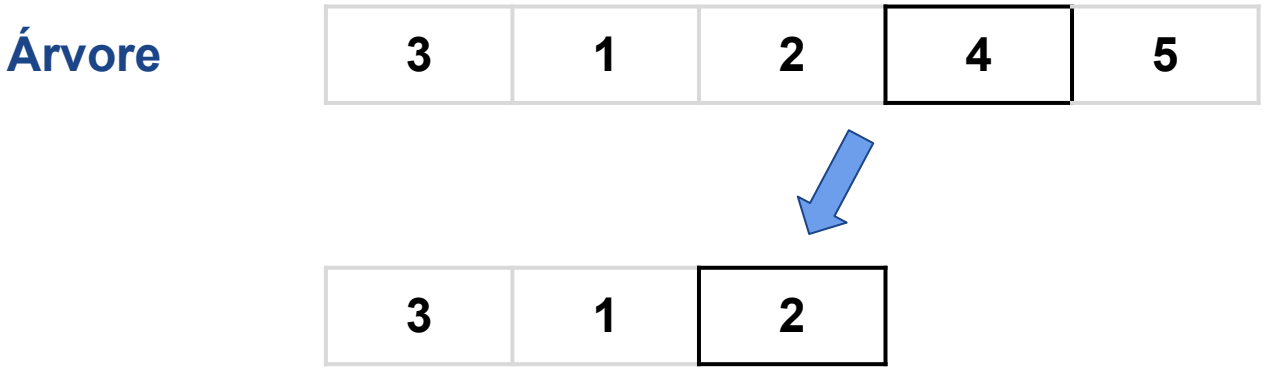
---

**Árvore**

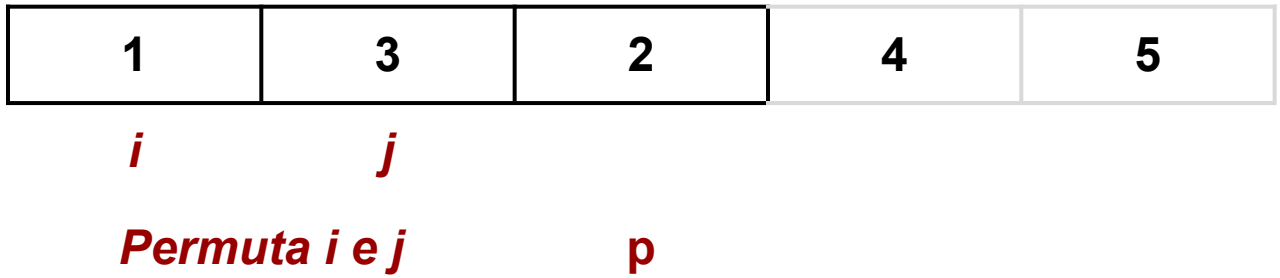


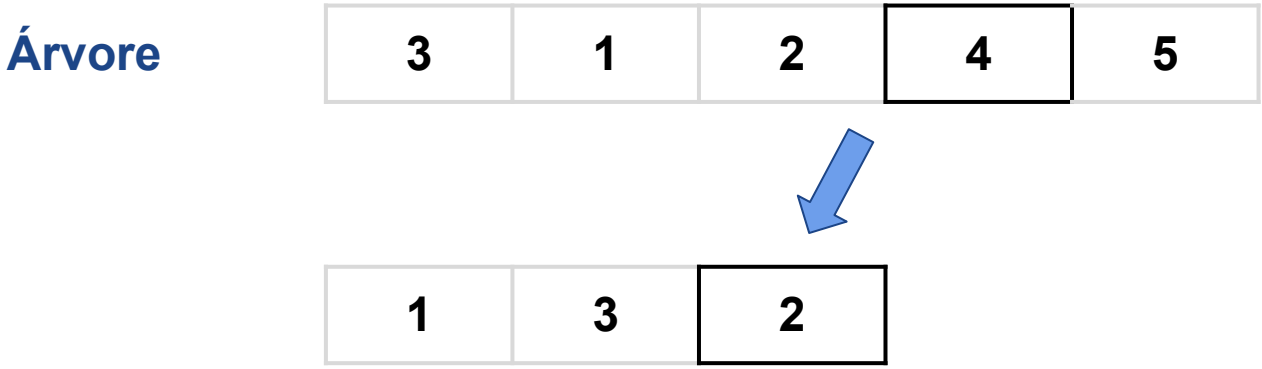
# Iteração 2



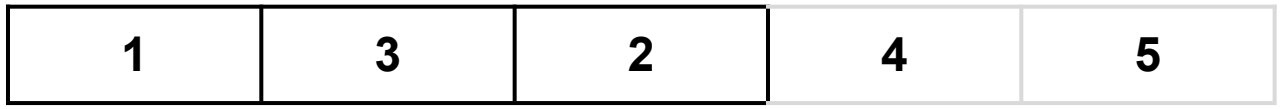



# Iteração 2



# Iteração 2



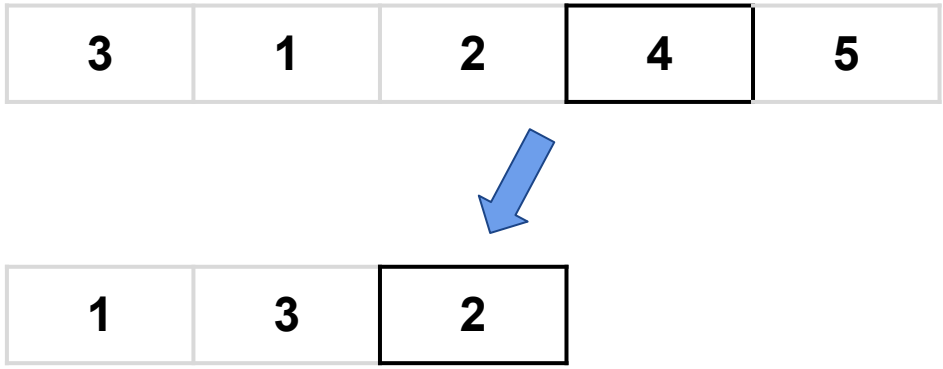

*i, j*

*Incrementa i*

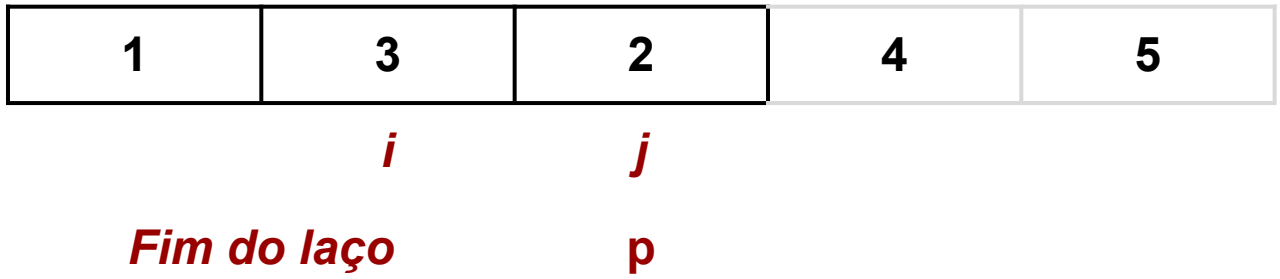
p

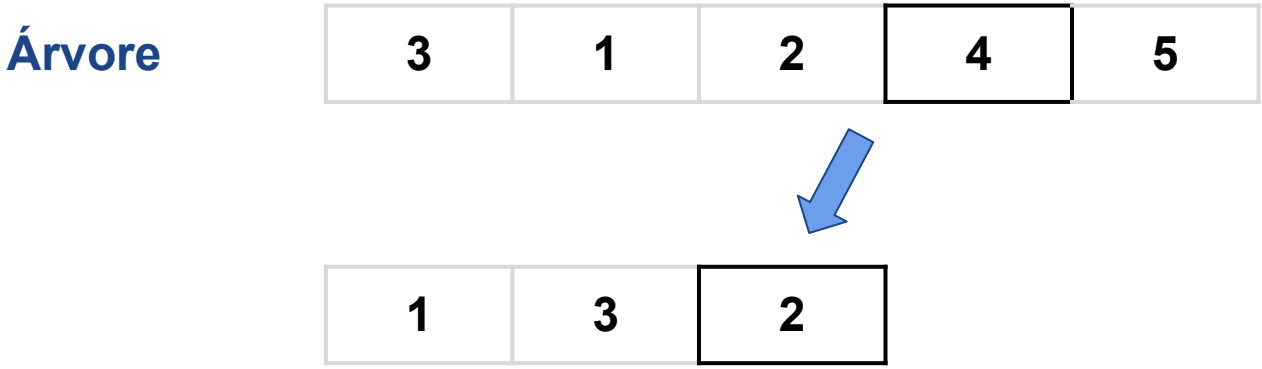
---

**Árvore**



# Iteração 2



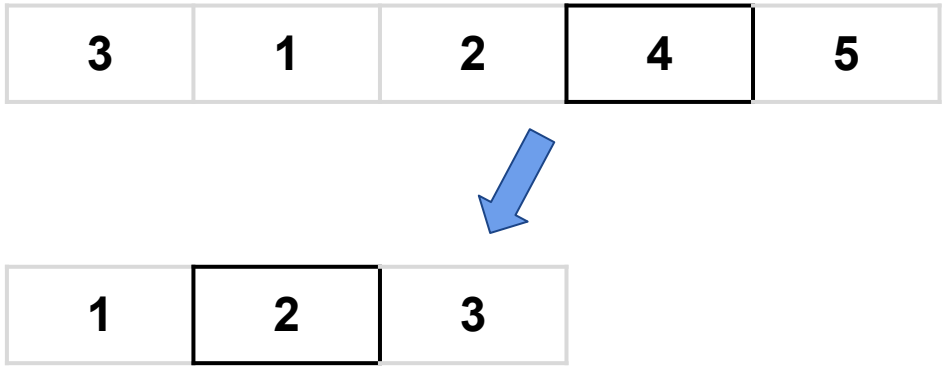



# Iteração 2


***Permuta i e p*** **p**

---

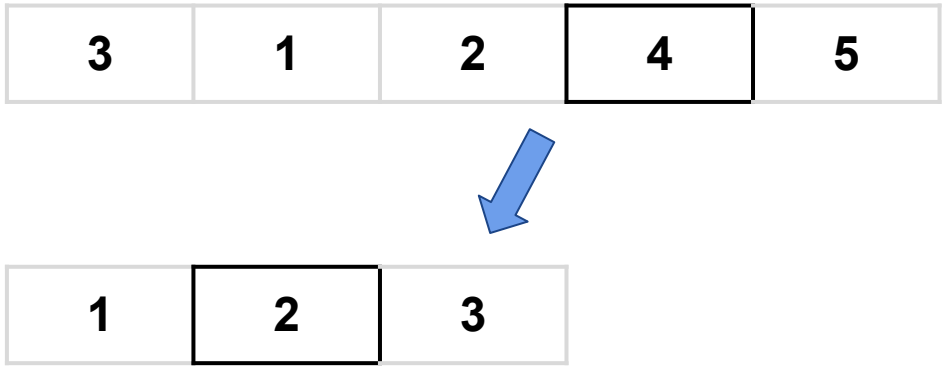
**Árvore**



# Iteração 2


p

**Árvore**

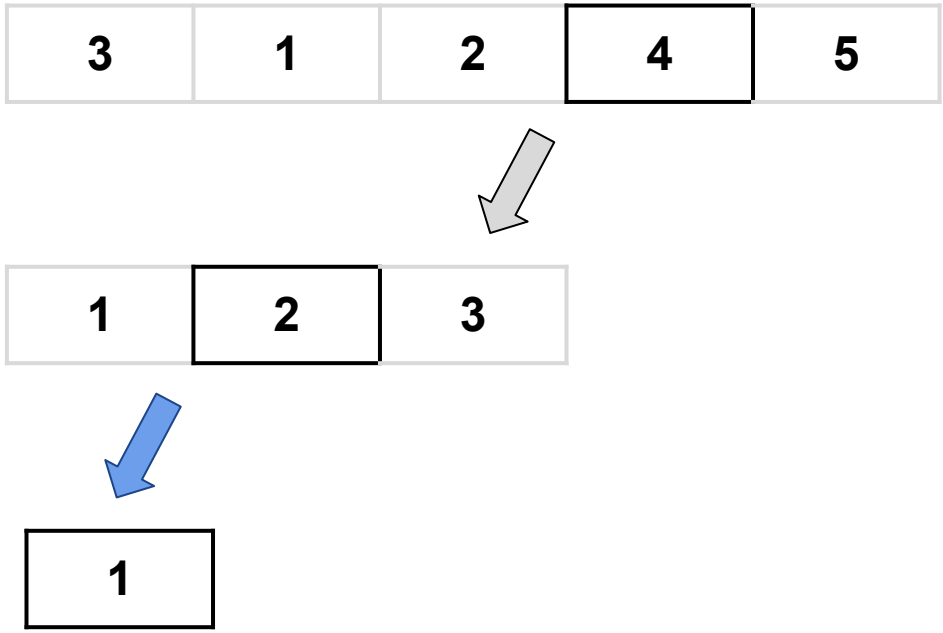


# Iteração 3


p

---

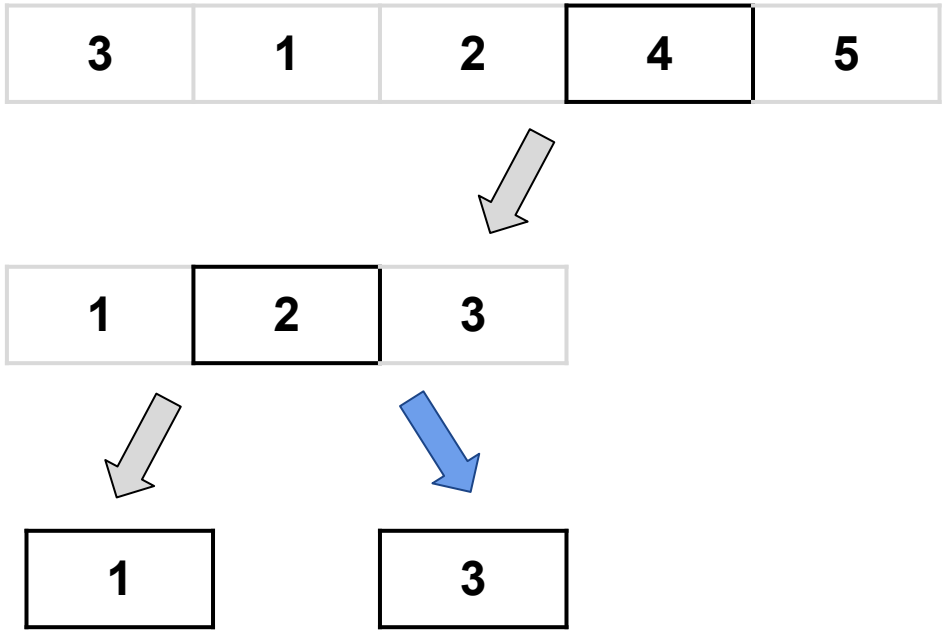
**Árvore**



# Iteração 4


p

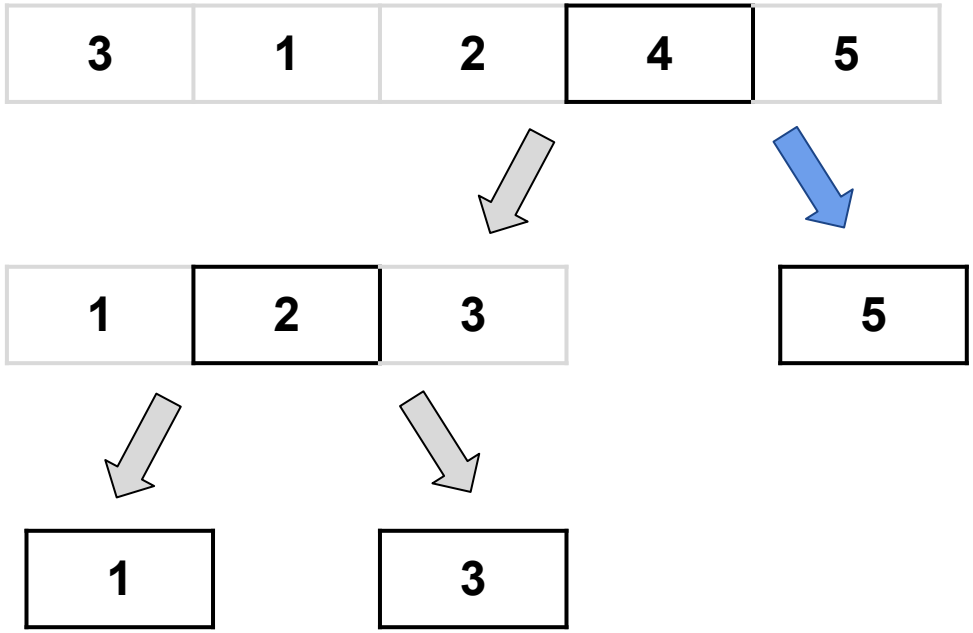
**Árvore**



# Iteração 5


p

**Árvore**



# Resultado

*Se a sub-lista (esquerda ou direita) tiver um ou nenhum elemento, considera-se já ordenada.*

# Exercício

- Ordene as listas a seguir utilizando Quick Sort;

| 9 | 12 | 2 | 3 | 8 | 0 | 6 |
|---|---|---|---|---|---|---|

| 21 | 23 | 2 | 34 | 245 | 33 | 66 |
|---|---|---|---|---|---|---|

# Implementação em sala

# Características

- Algoritmo do tipo dividir para conquistar;
- Bom desempenho;
- In place
- Não é Stable;
- Muito usado na prática;

# Laboratório23

- Implemente o algoritmo Quick Sort para ordenar uma lista encadeada;