

# *Ordenação por Inserção*

## *Insert-sort*

## *Ordenação por Inserção – Insert Sort*

- Algoritmo de Ordenação por Inserção, ou *Insert Sort*;
- Baseia-se na ideia de percorrer um vetor, deixando-o ordenado a cada elemento que avança;
- Semelhante à maneira que as pessoas ordenam documentos, cartas de baralho etc.;

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

- No melhor caso, esse algoritmo executa  $N-1$  operações relevantes, onde  $N$  representa o número de elementos do vetor. Menor número de trocas e comparações entre os algoritmos semelhantes;
- No pior caso, são feitas  $N^2$  operações relevantes, portanto, sua complexidade é de Ordem Quadrática;

Conclusão: Quanto maior for a quantidade de dados a serem ordenados, pior (ao quadrado) será seu desempenho.

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor:

0	1	2	3	4	5
5	3	2	4	6	1

Vetor qualquer, supostamente desordenado!

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor:

0	1	2	3	4	5
5	3	2	4	6	1

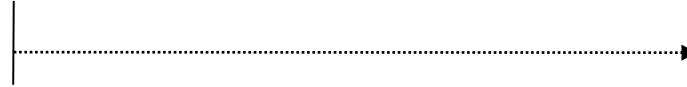


O algoritmo ordenará o vetor crescentemente

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

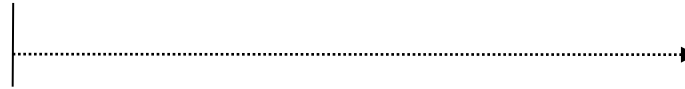
0	1	2	3	4	5
5	3	2	4	6	1

Iteração 0

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

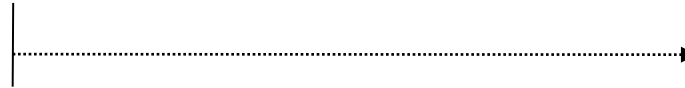
0	1	2	3	4	5
5	3	2	4	6	1

Elemento a ser analisado: Índice 1

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

0	1	2	3	4	5
5	3	2	4	6	1

Atribuição

Rascunho:

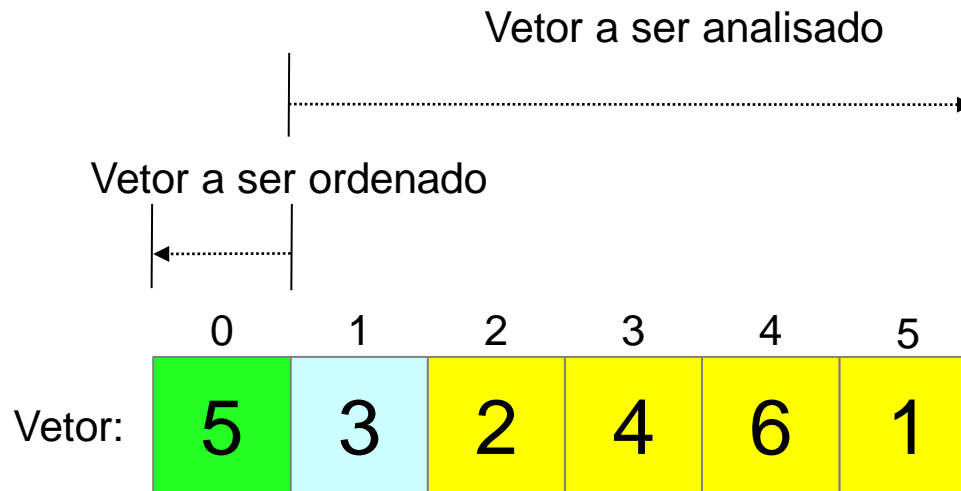


Armazena o elemento analisado: Índice 1



# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



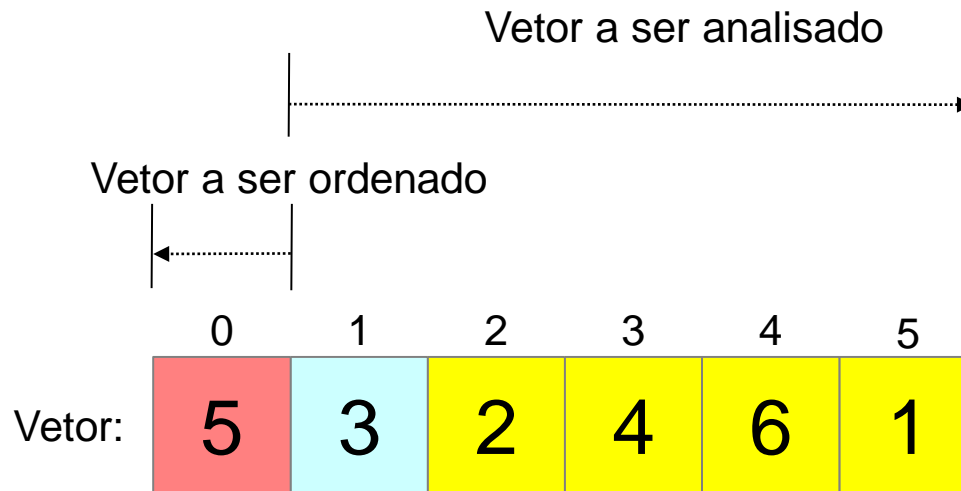
Rascunho:

3

Vetor a ser ordenado: do índice 0 ao índice 0

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



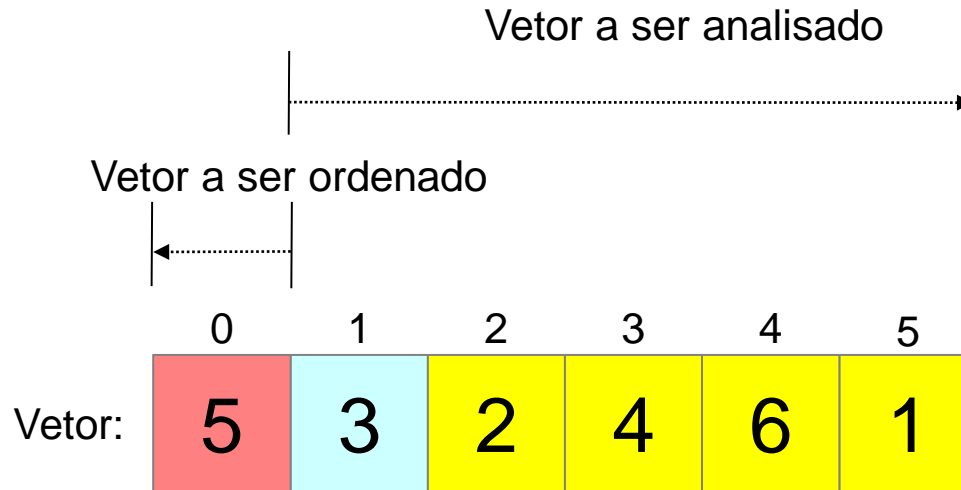
Rascunho:

3

Elemento do vetor a ser comparado: Índice 0

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

3

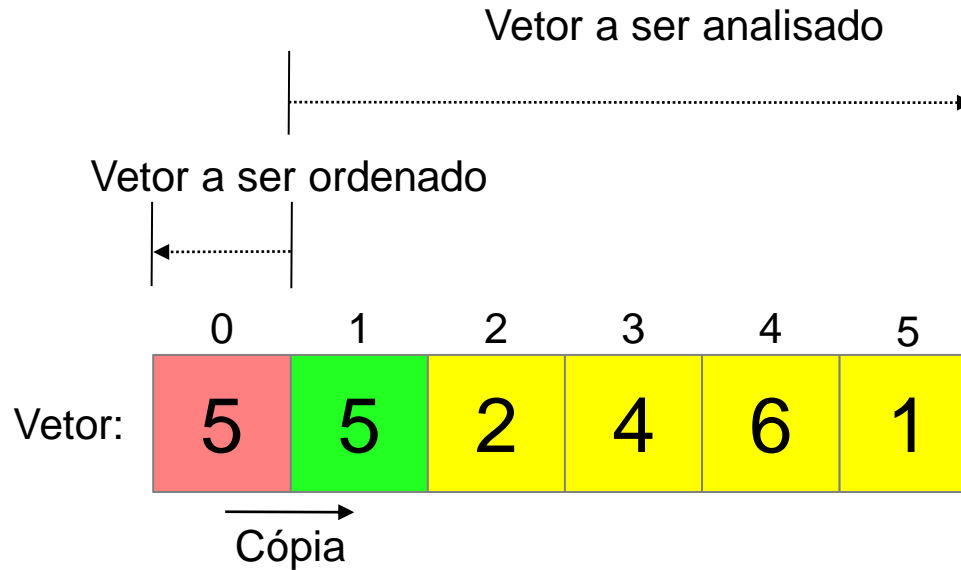
Se  $5 > 3$ , desloca;  
Senão, mantém.

**Portanto, desloca-se!**

Aplica-se a condição ao: Índice 0

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



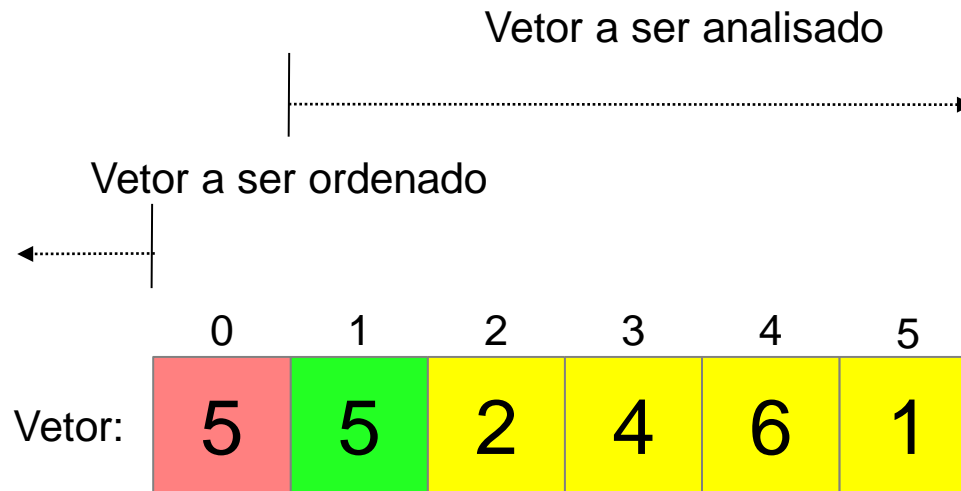
Rascunho:

3

Desloca-se: do índice 0 para o índice 1

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



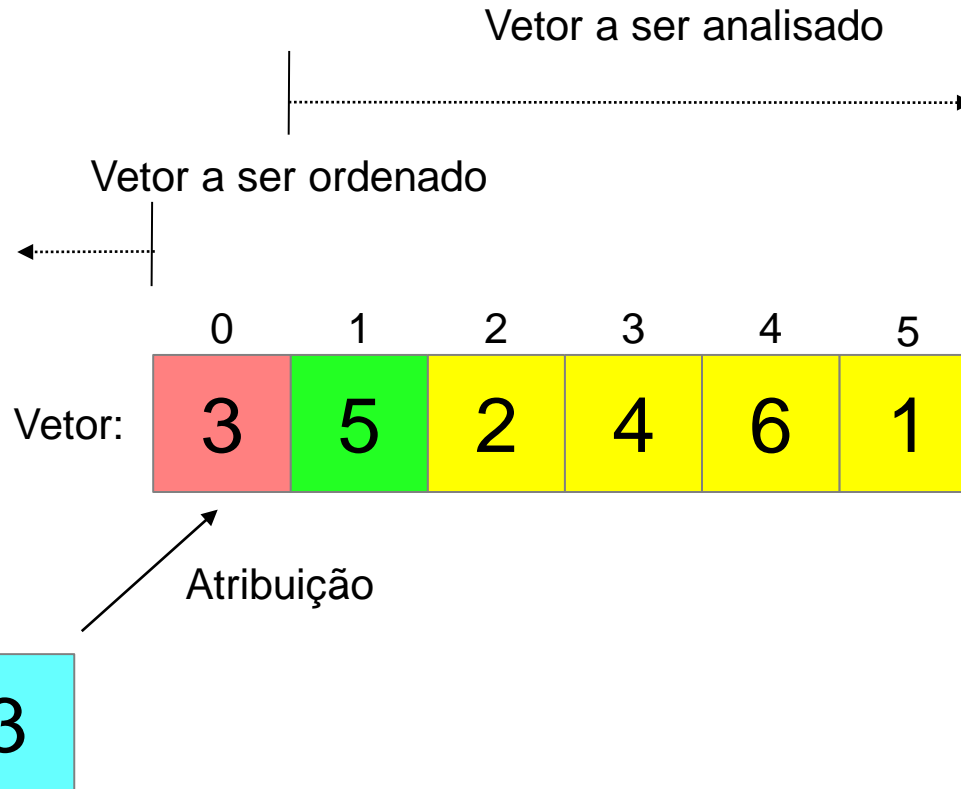
Rascunho:

3

Vetor a ser ordenado: encerrado!

# Estruturas de Dados e Análise de Algoritmos

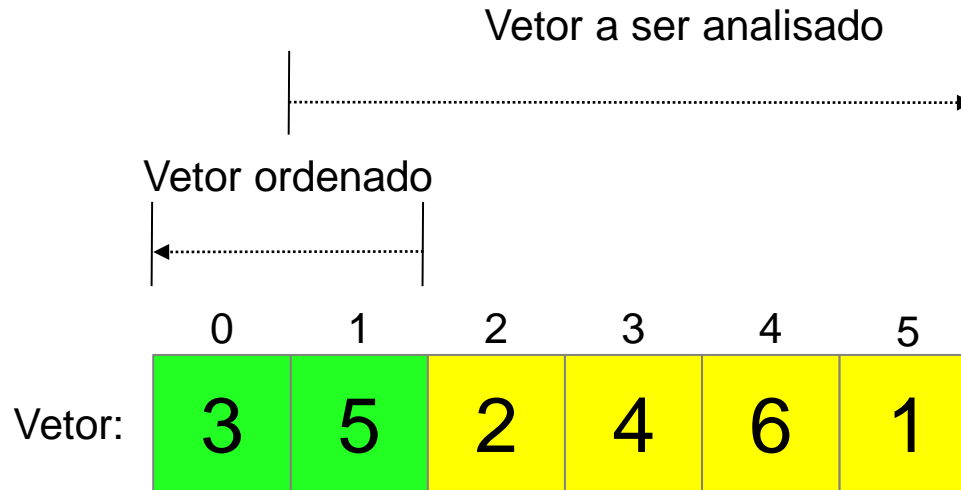
## Ordenação por Inserção – Insert Sort



Retorna o elemento armazenado: Índice 0

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

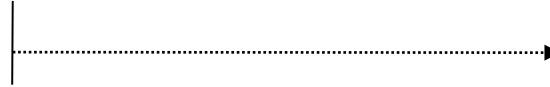


Vetor parcialmente ordenado!

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

0	1	2	3	4	5
3	5	2	4	6	1

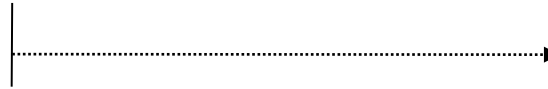
Iteração 1



# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

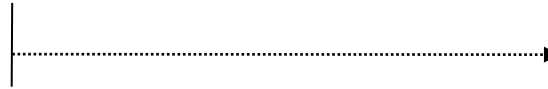
0	1	2	3	4	5
3	5	2	4	6	1

Elemento a ser analisado: Índice 2

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

0	1	2	3	4	5
3	5	2	4	6	1

Atribuição

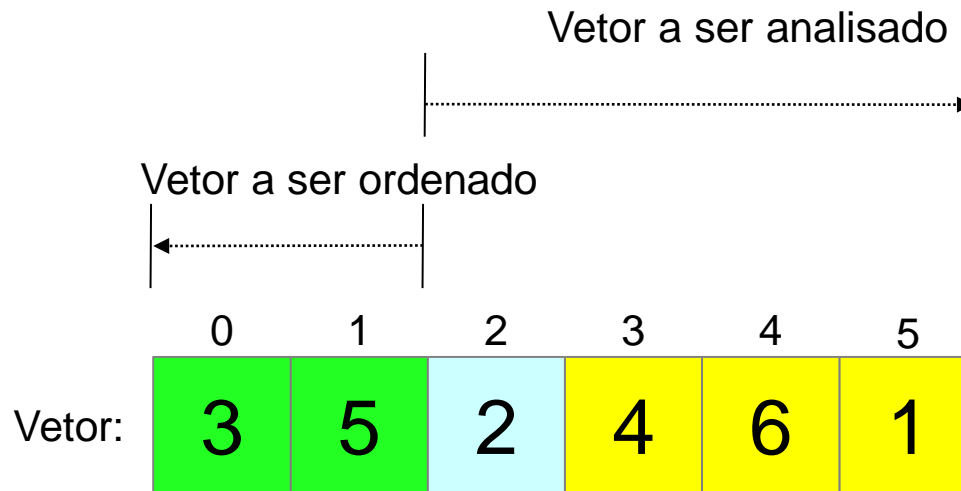
Rascunho:

2

Armazena o elemento analisado: Índice 2

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

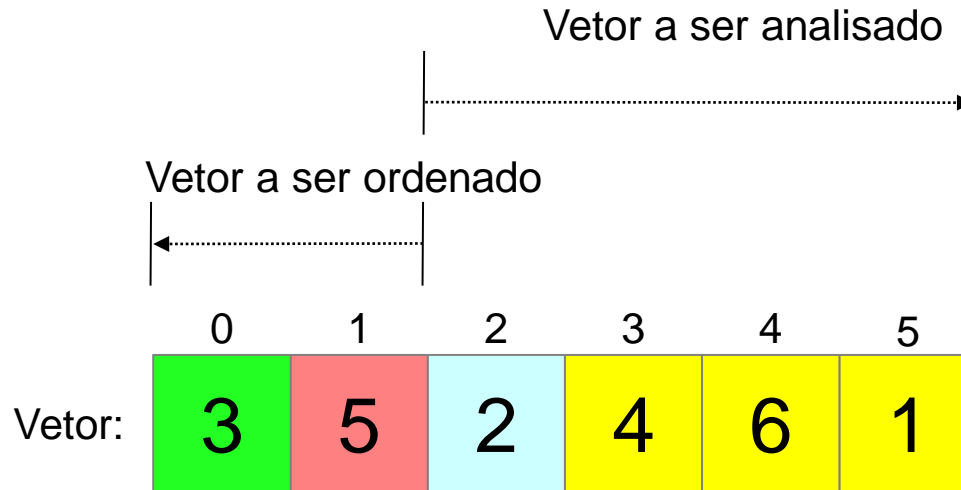


Rascunho: 2

Vetor a ser ordenado: do índice 0 ao índice 1

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

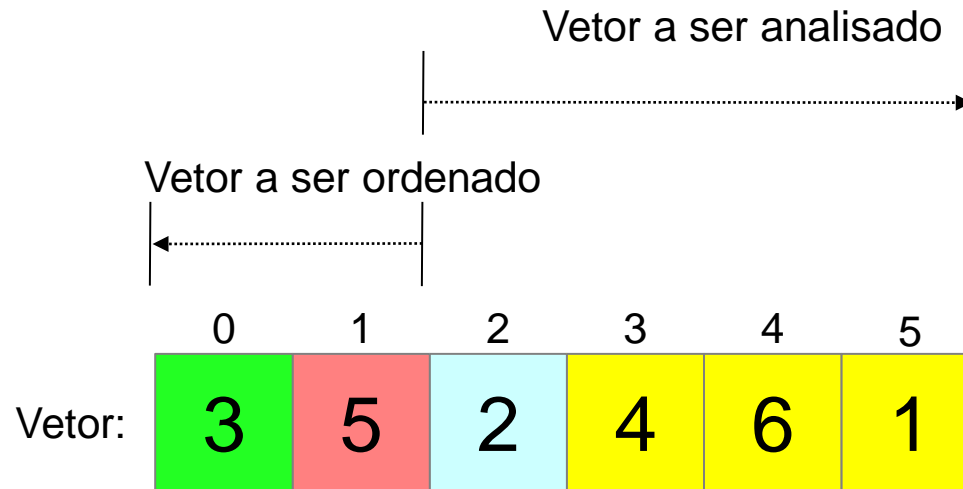


Rascunho: 2

Elemento do vetor a ser comparado: Índice 1

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

2

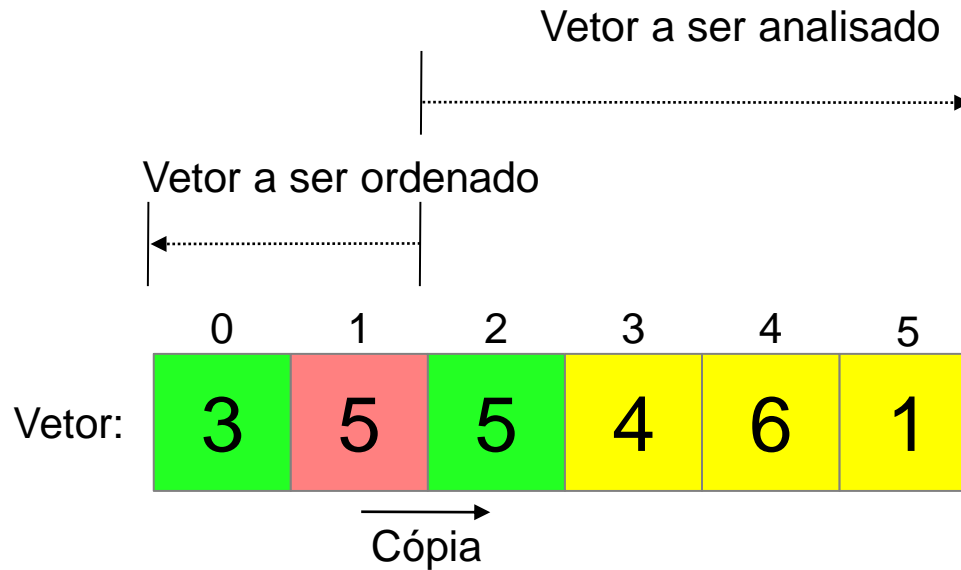
Se  $5 > 2$ , desloca;  
Senão, mantém.

**Portanto, desloca-se!**

Aplica-se a condição ao: Índice 1

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



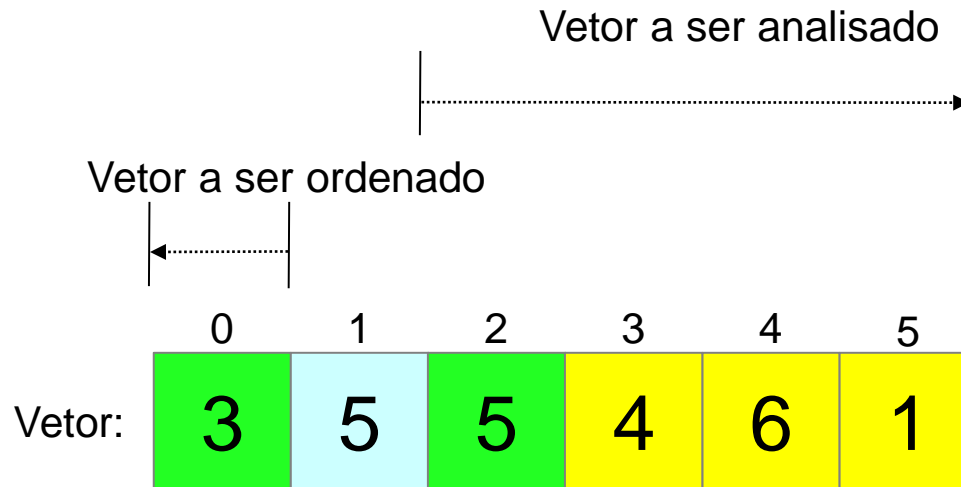
Rascunho:

2

Desloca-se: do índice 1 para o índice 2

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

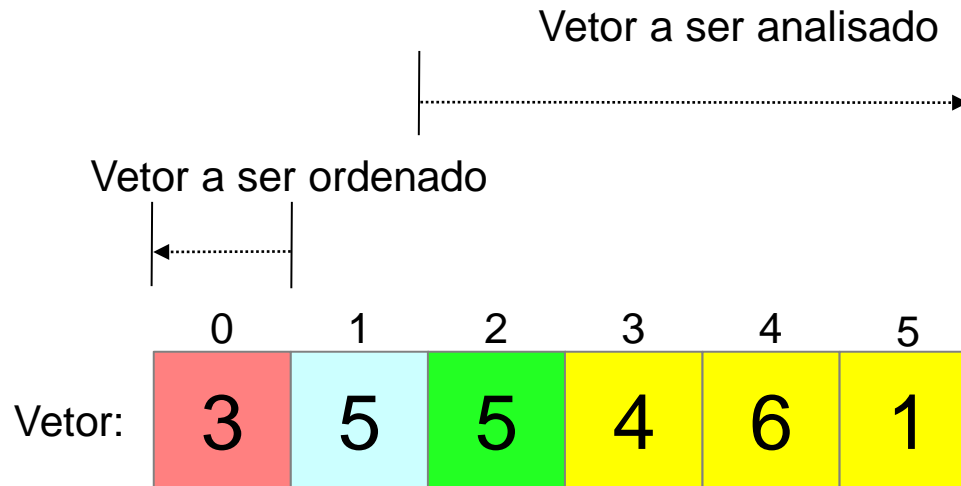


Rascunho: 2

Vetor a ser ordenado: do índice 0 ao índice 0

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

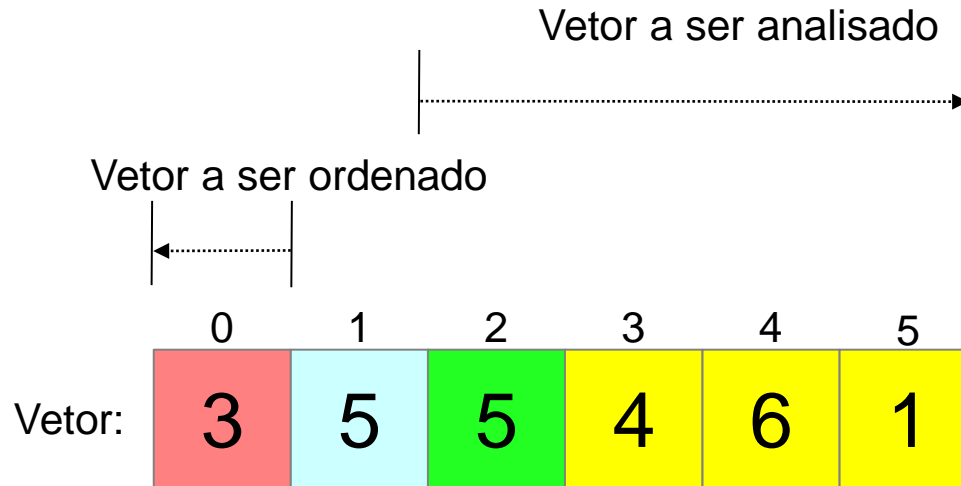
2

Elemento do vetor a ser comparado: Índice 0



# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

2

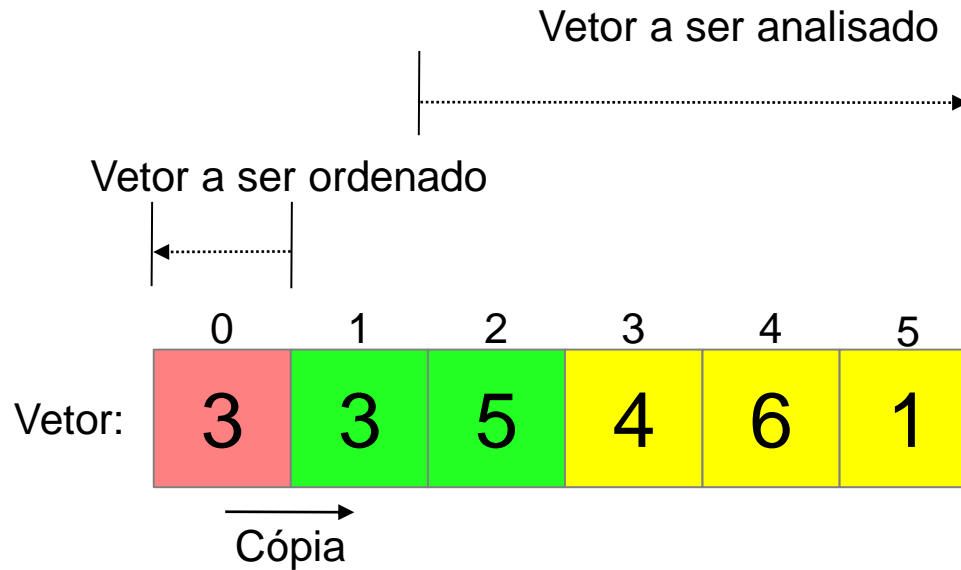
Se  $3 > 2$ , desloca;  
Senão, mantém.

**Portanto, desloca-se!**

Aplica-se a condição ao: Índice 0

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

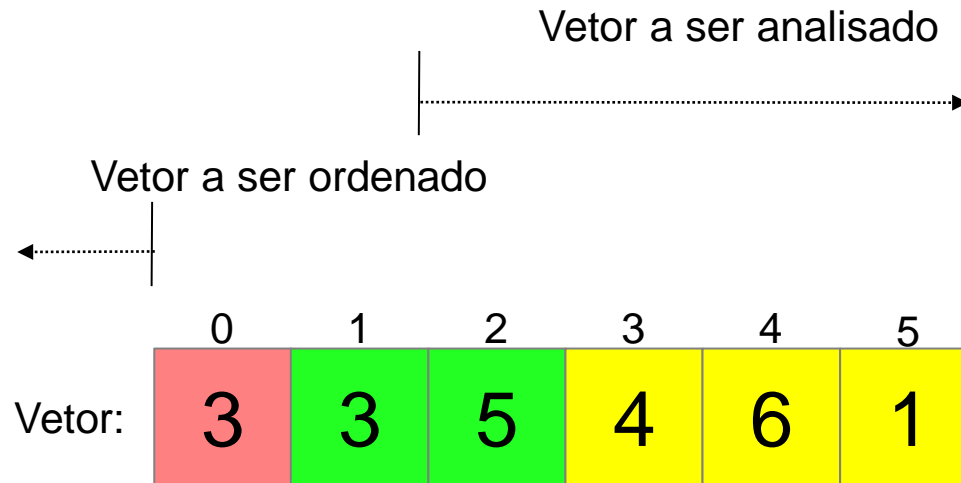


Rascunho: 2

Desloca-se: do índice 0 para o índice 1

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



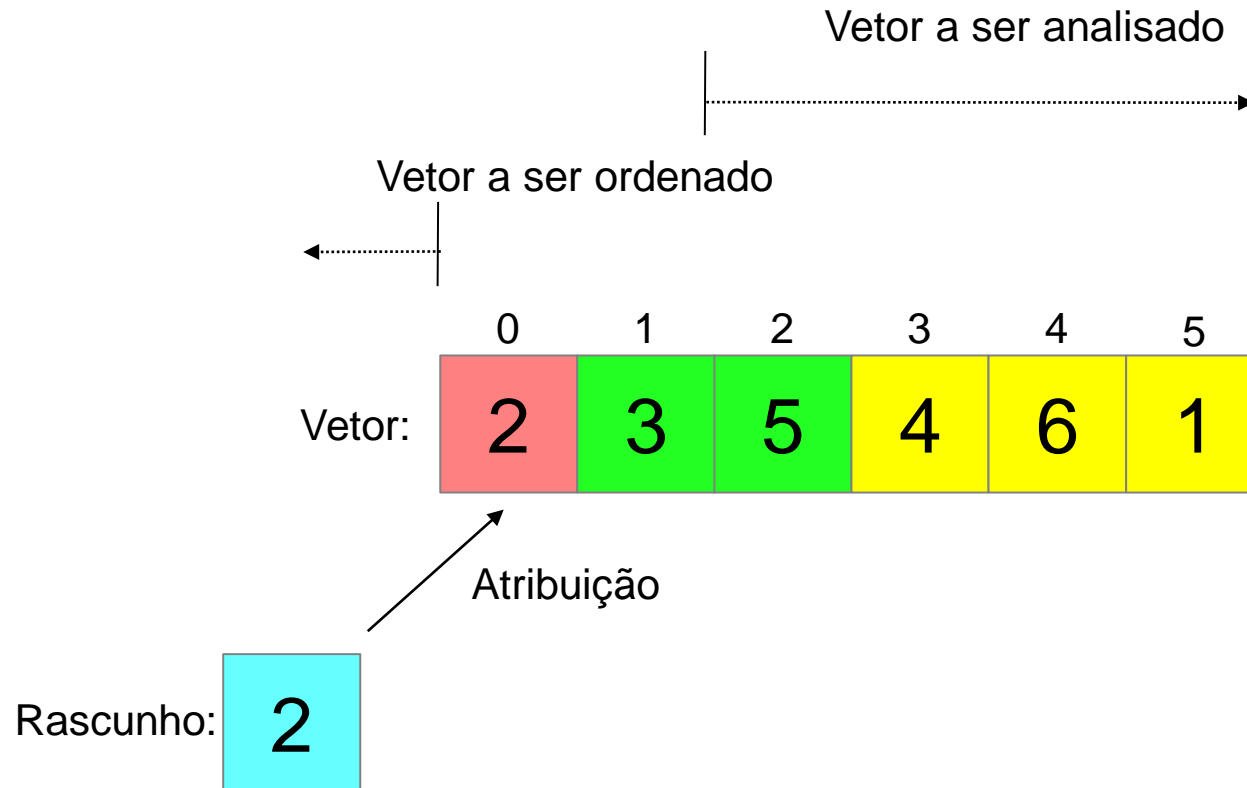
Rascunho:

2

Vetor a ser ordenado: encerrado!

# Estruturas de Dados e Análise de Algoritmos

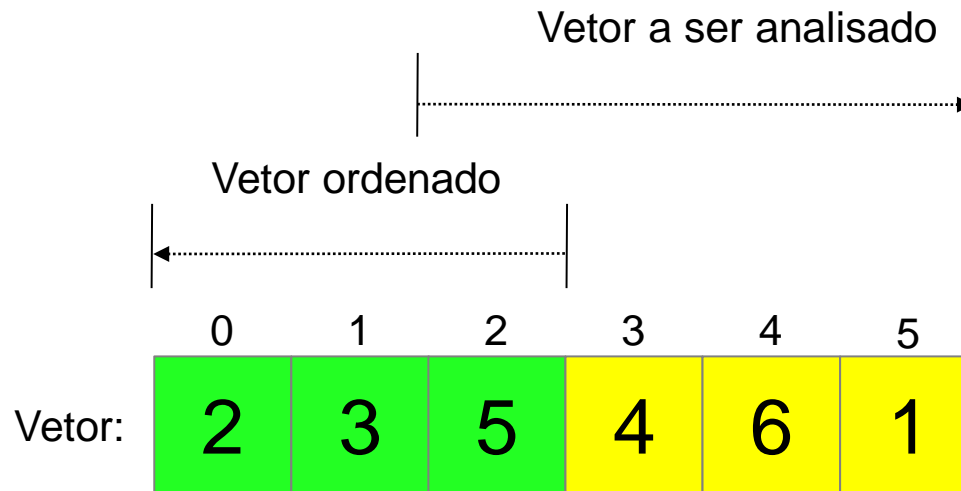
## Ordenação por Inserção – Insert Sort



Retorna o elemento armazenado: Índice 0

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

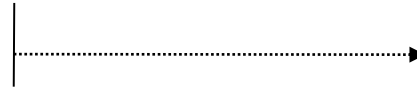


Vetor parcialmente ordenado!

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

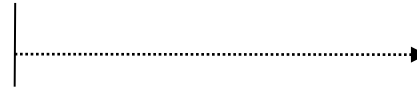
0	1	2	3	4	5
2	3	5	4	6	1

Iteração 2

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

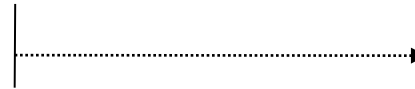
0	1	2	3	4	5
2	3	5	4	6	1

Elemento a ser analisado: Índice 3

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

0	1	2	3	4	5
2	3	5	4	6	1

Atribuição

Rascunho:

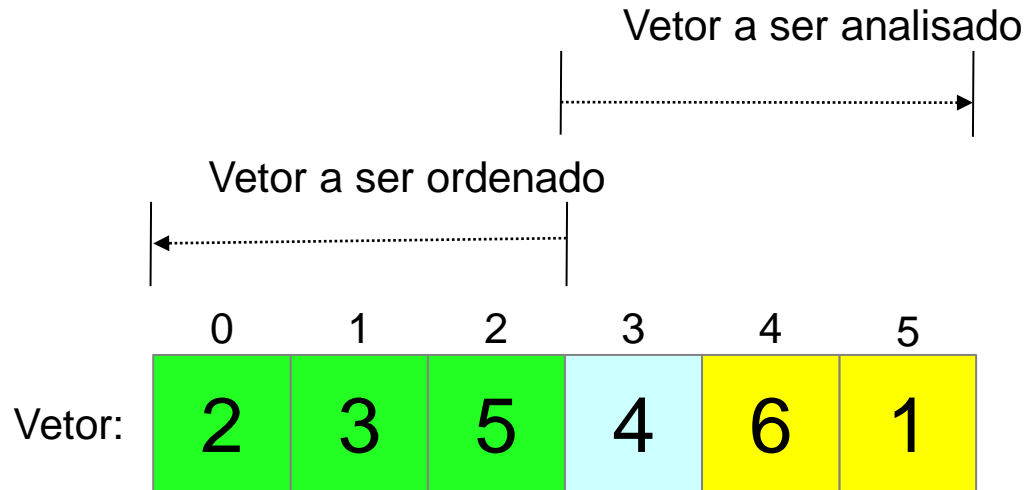


Armazena o elemento analisado: Índice 3



# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

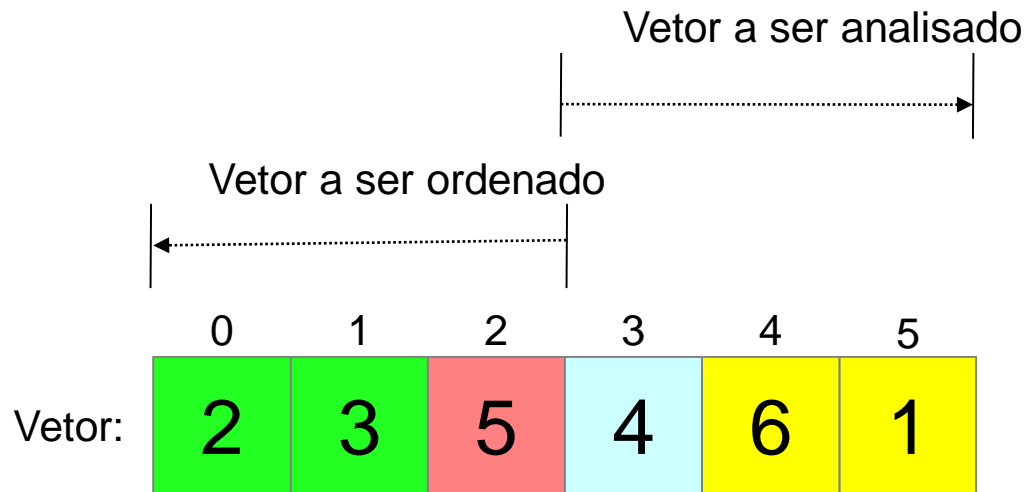


Rascunho: 4

Vetor a ser ordenado: do índice 0 ao índice 2

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

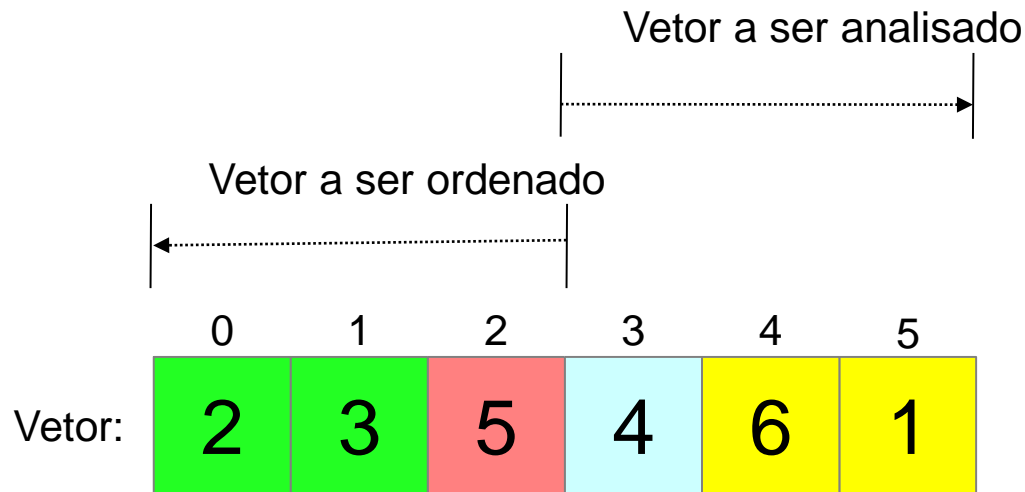


Rascunho: 4

Elemento do vetor a ser comparado: Índice 2

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

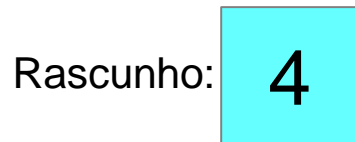
4

Se  $5 > 4$ , desloca;  
Senão, mantém.

**Portanto, desloca-se!**

Aplica-se a condição ao: Índice 2

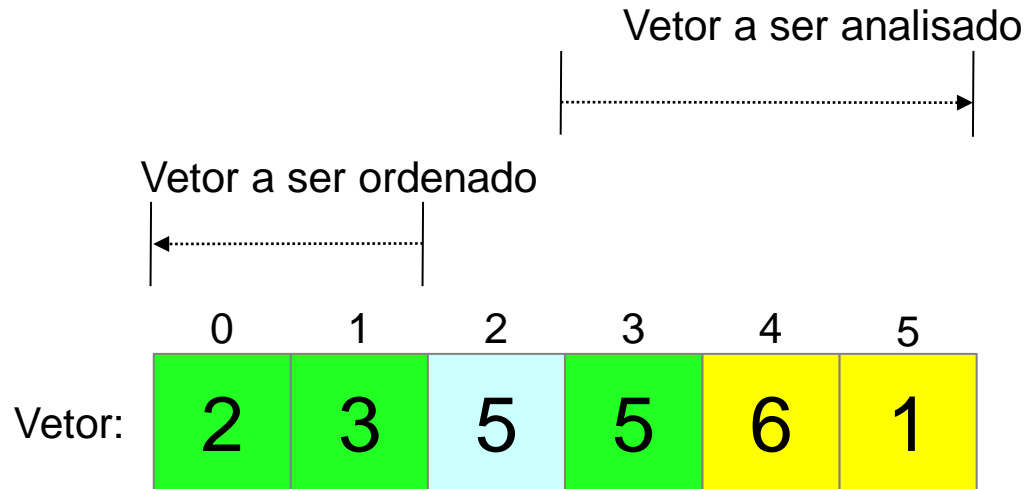
## Ordenação por Inserção – Insert Sort



Prof. Calvetti

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

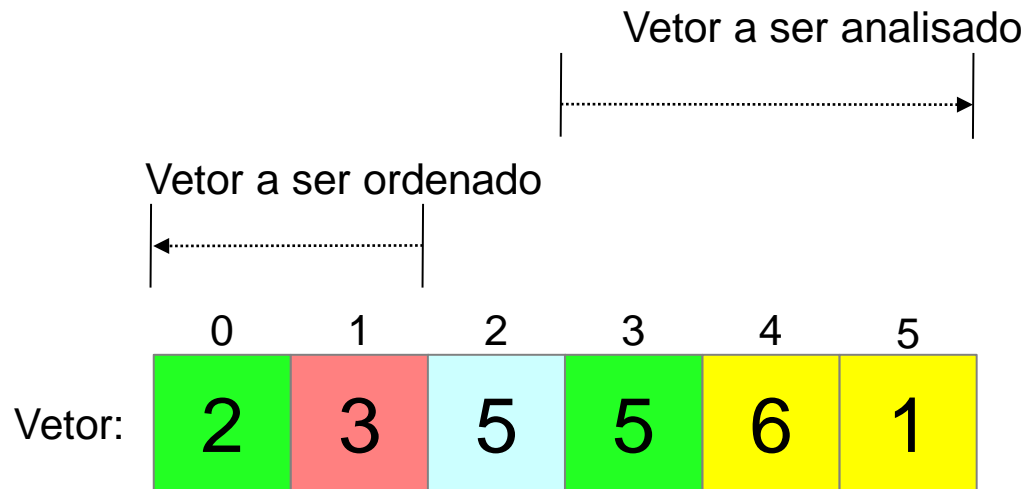


Rascunho: 4

Vetor a ser ordenado: do índice 0 ao índice 1

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

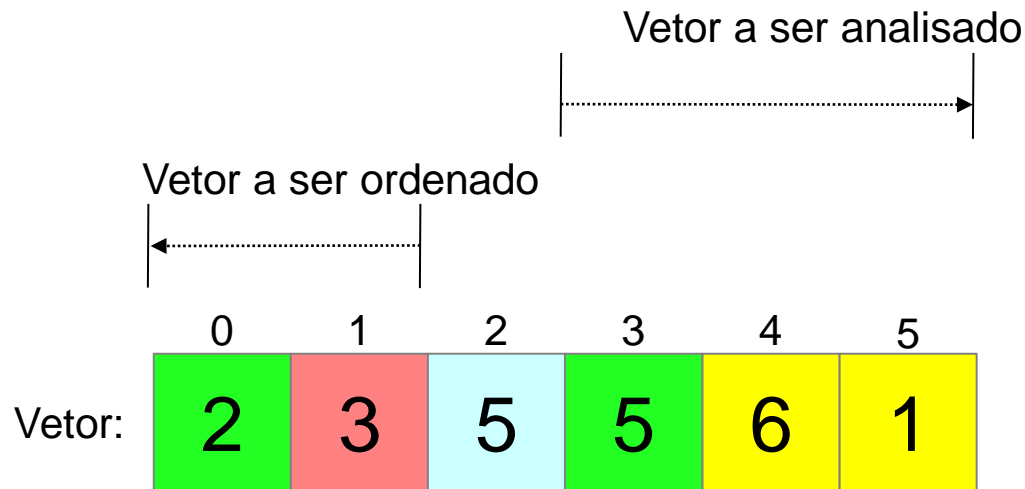


Rascunho: 4

Elemento do vetor a ser comparado: Índice 1

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

4

Se  $3 > 4$ , desloca;  
Senão, mantém.

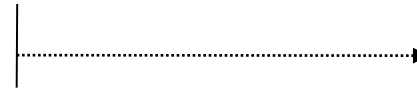
**Portanto, mantém-se!**

Aplica-se a condição ao: Índice 1

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

0	1	2	3	4	5
2	3	4	5	6	1

Atribuição

Rascunho:

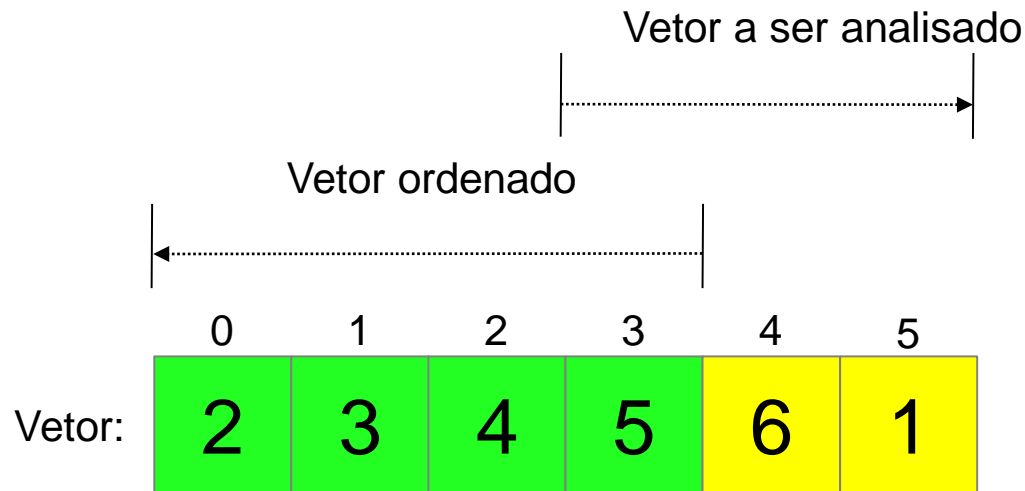
4

Retorna o elemento armazenado: Índice 2



# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

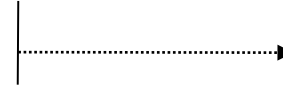


Vetor parcialmente ordenado!

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

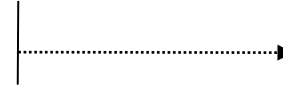
0	1	2	3	4	5
2	3	4	5	6	1

Iteração 3

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

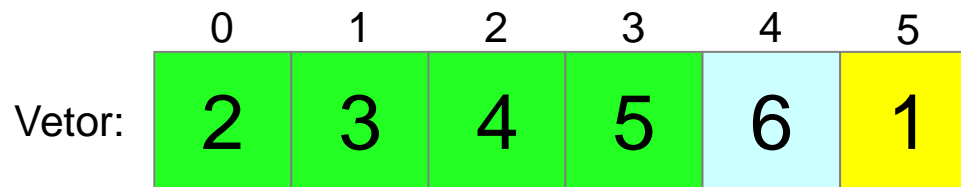
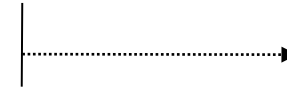
0	1	2	3	4	5
2	3	4	5	6	1

Elemento a ser analisado: Índice 4

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

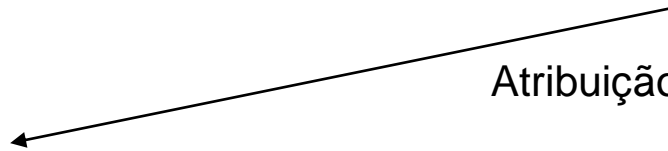
Vetor a ser analisado



Rascunho:



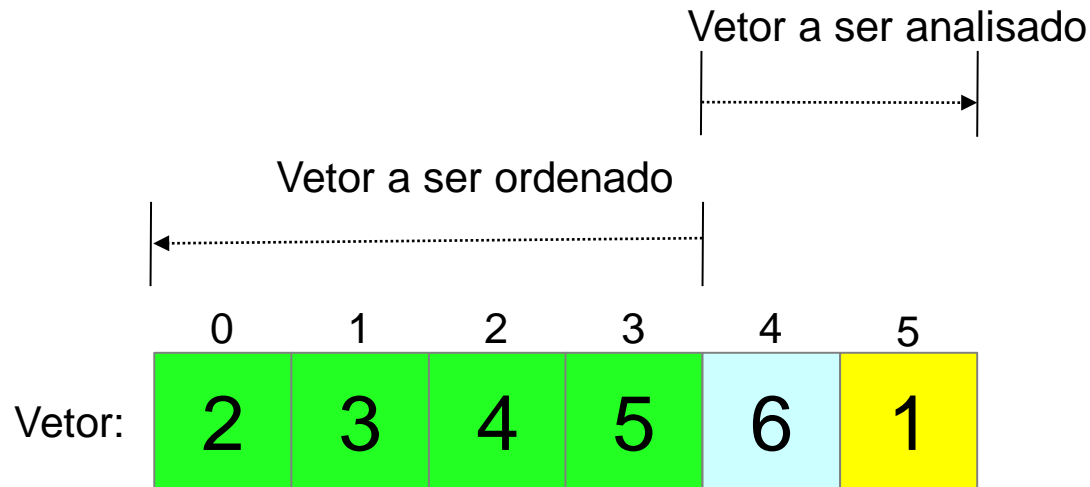
Atribuição



Armazena o elemento analisado: Índice 4

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

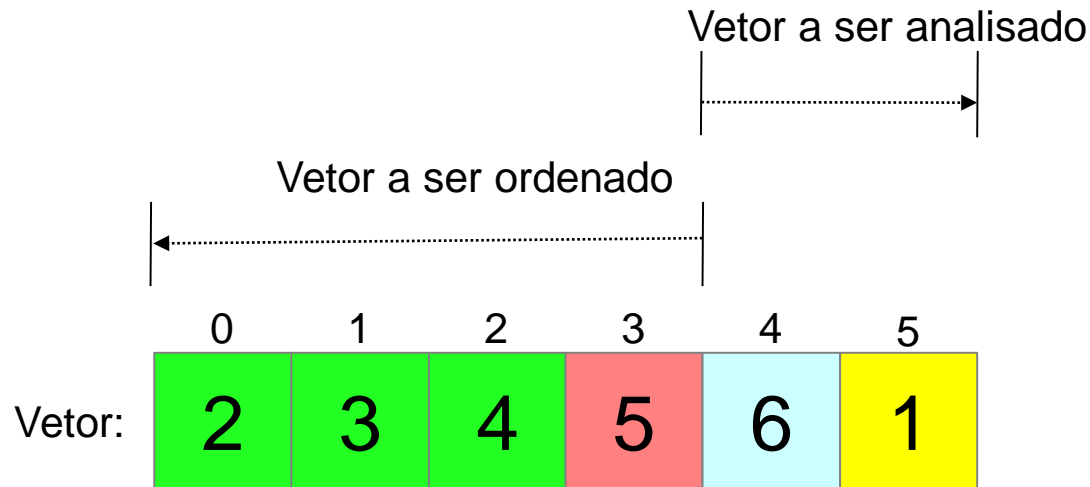


Rascunho: 6

Vetor a ser ordenado: do índice 0 ao índice 3

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

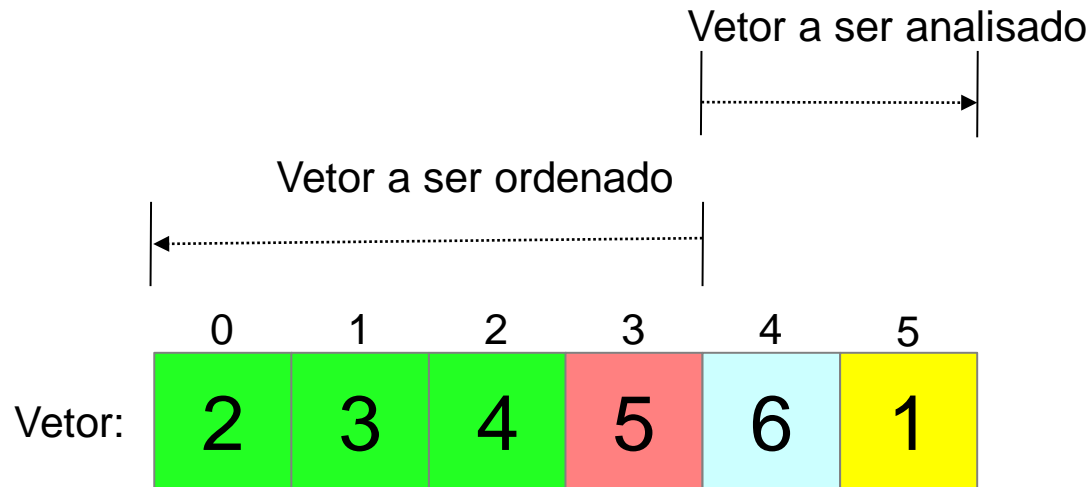


Rascunho: 6

Elemento do vetor a ser comparado: Índice 3

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

6

Se  $5 > 6$ , desloca;  
Senão, mantém.

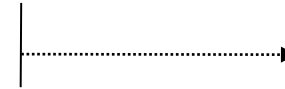
**Portanto, mantém-se!**

Aplica-se a condição ao: Índice 3

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

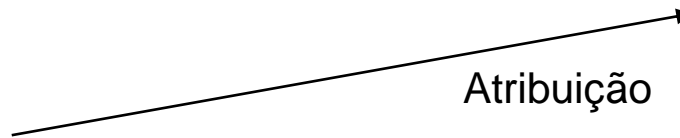
Vetor a ser analisado



Rascunho:



Atribuição

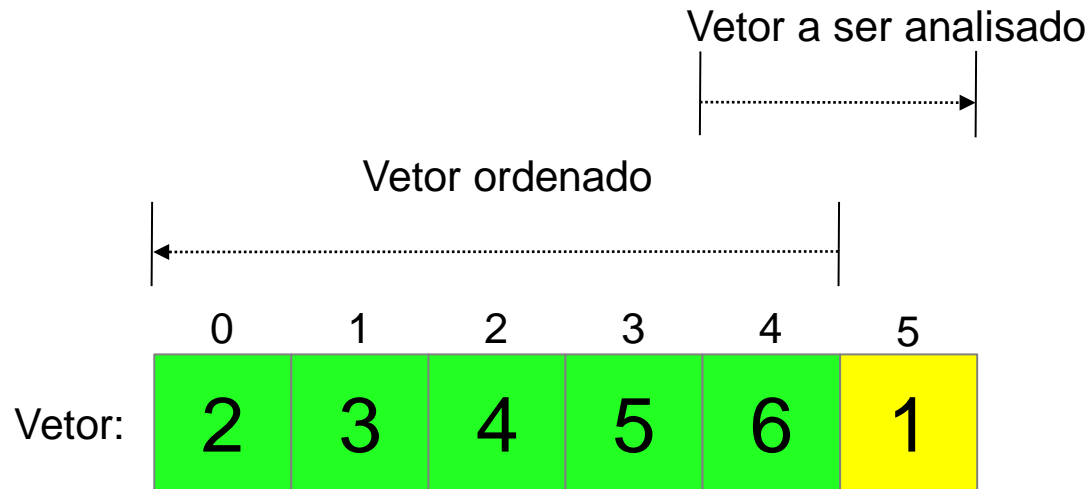


Retorna o elemento armazenado: Índice 4



# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

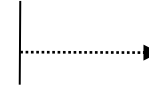


Vetor parcialmente ordenado!

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

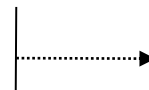
0	1	2	3	4	5
2	3	4	5	6	1

Iteração 4

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor:

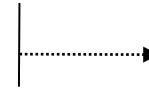
0	1	2	3	4	5
2	3	4	5	6	1

Elemento a ser analisado: Índice 5

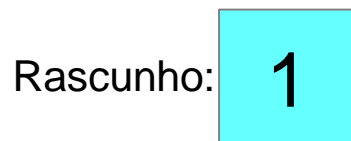
# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



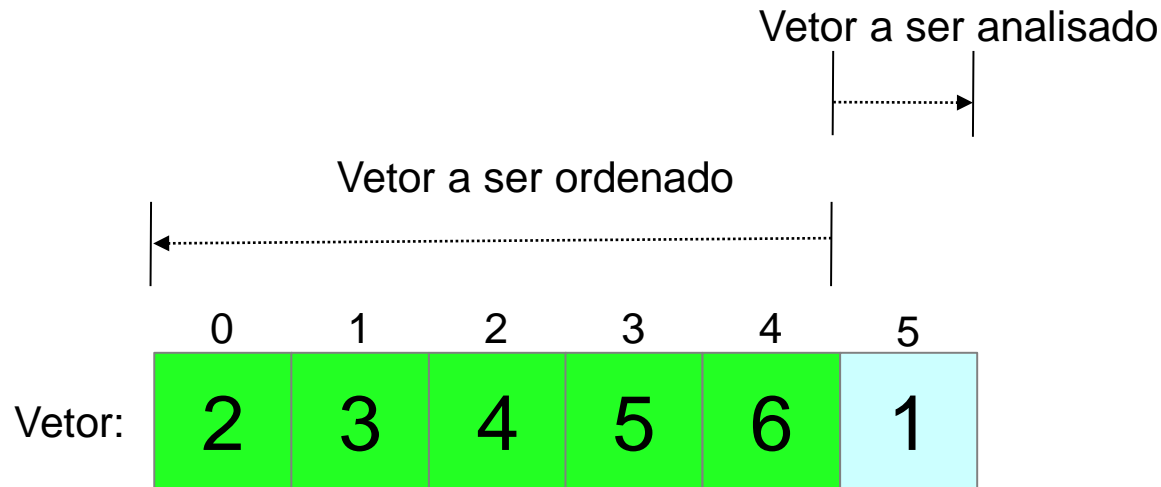
Atribuição



Armazena o elemento analisado: Índice 5

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



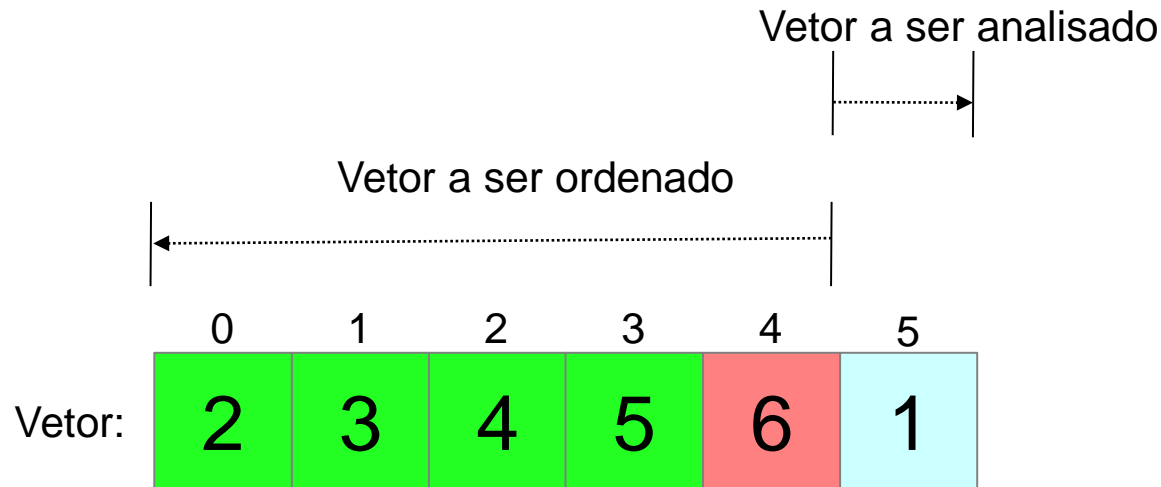
Rascunho:

1

Vetor a ser ordenado: do índice 0 ao índice 4

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



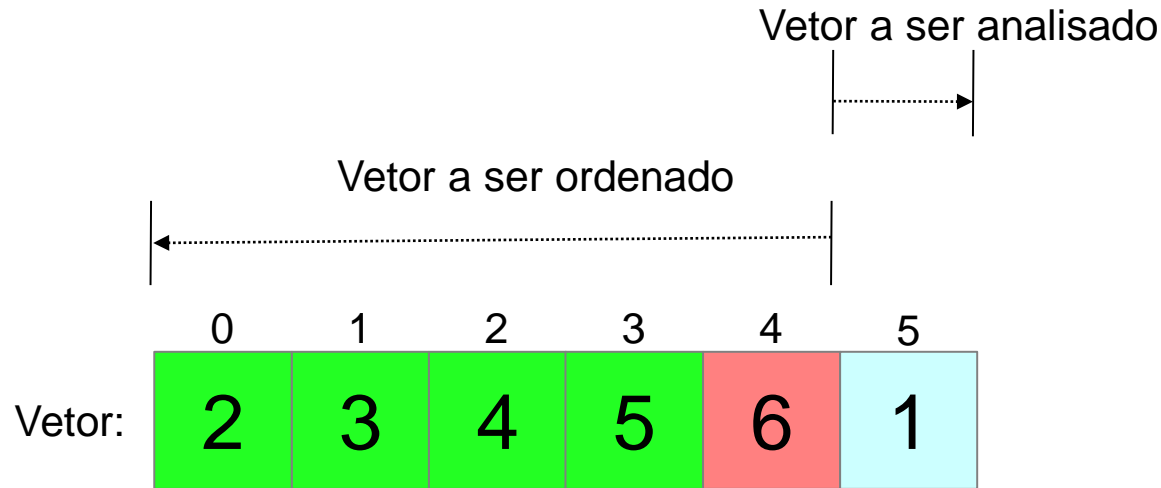
Rascunho:

1

Elemento do vetor a ser comparado: Índice 4

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho: 1

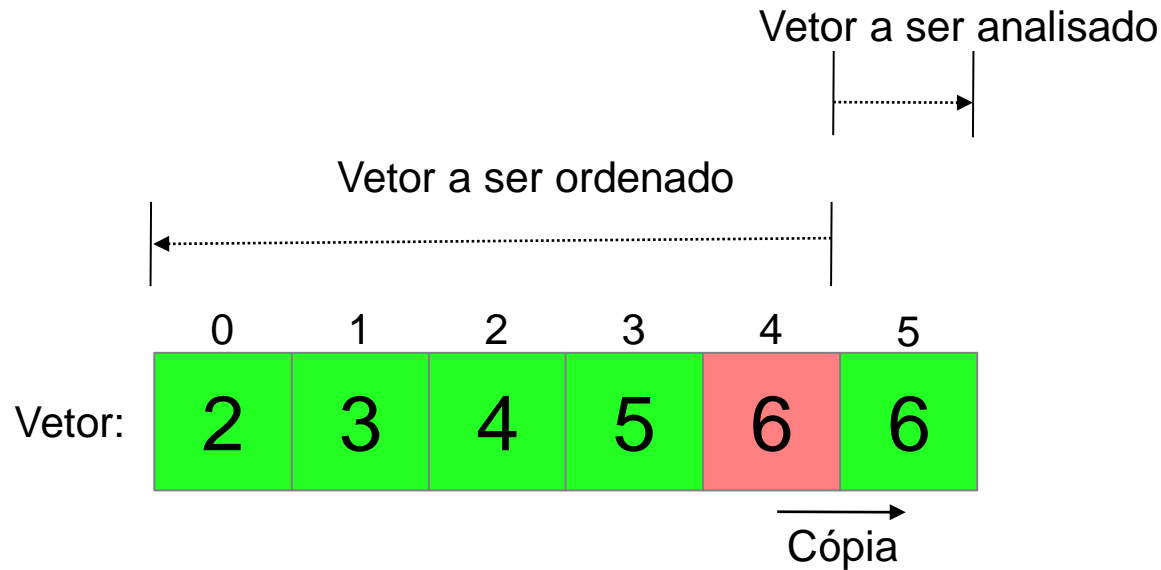
Se  $6 > 1$ , desloca;  
Senão, mantém.

**Portanto, desloca-se!**

Aplica-se a condição ao: Índice 4

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

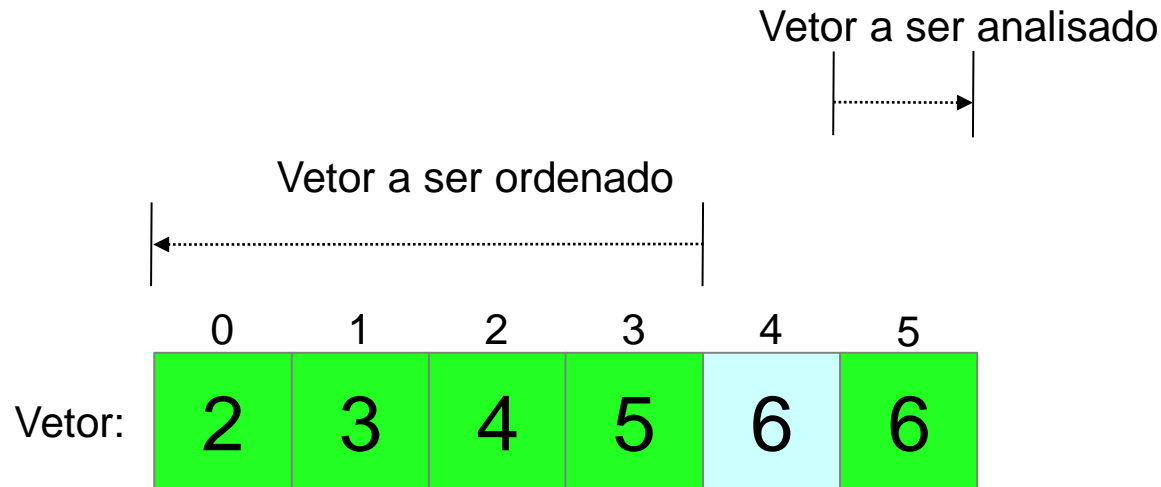
1

Desloca-se: do índice 4 para o índice 5



# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



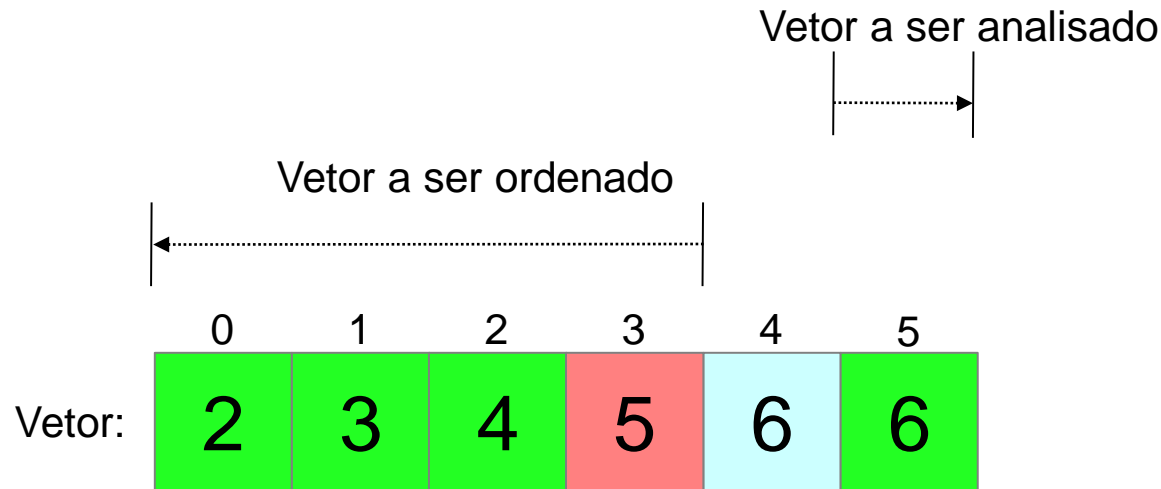
Rascunho:

1

Vetor a ser ordenado: do índice 0 ao índice 3

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



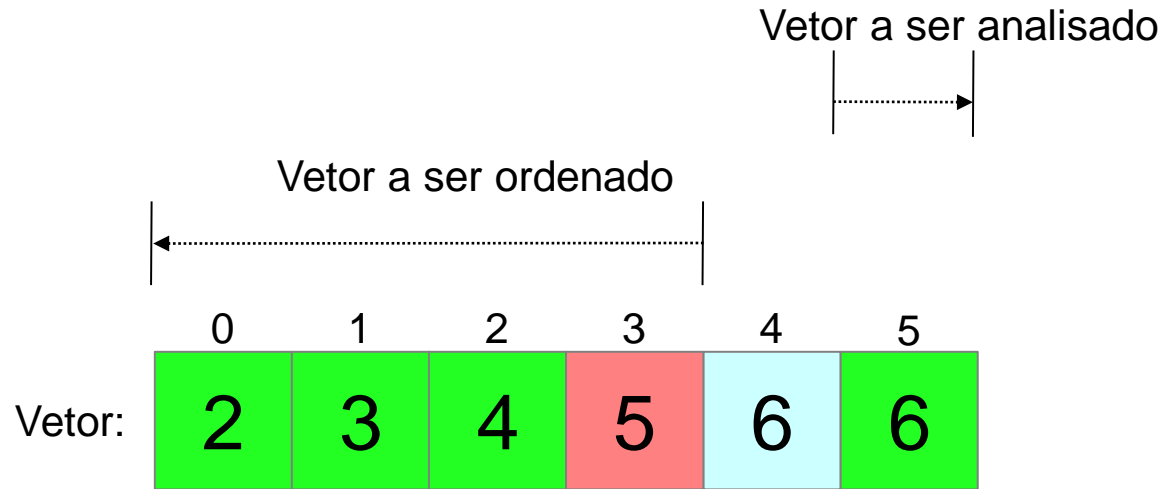
Rascunho:

1

Elemento do vetor a ser comparado: Índice 3

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

1

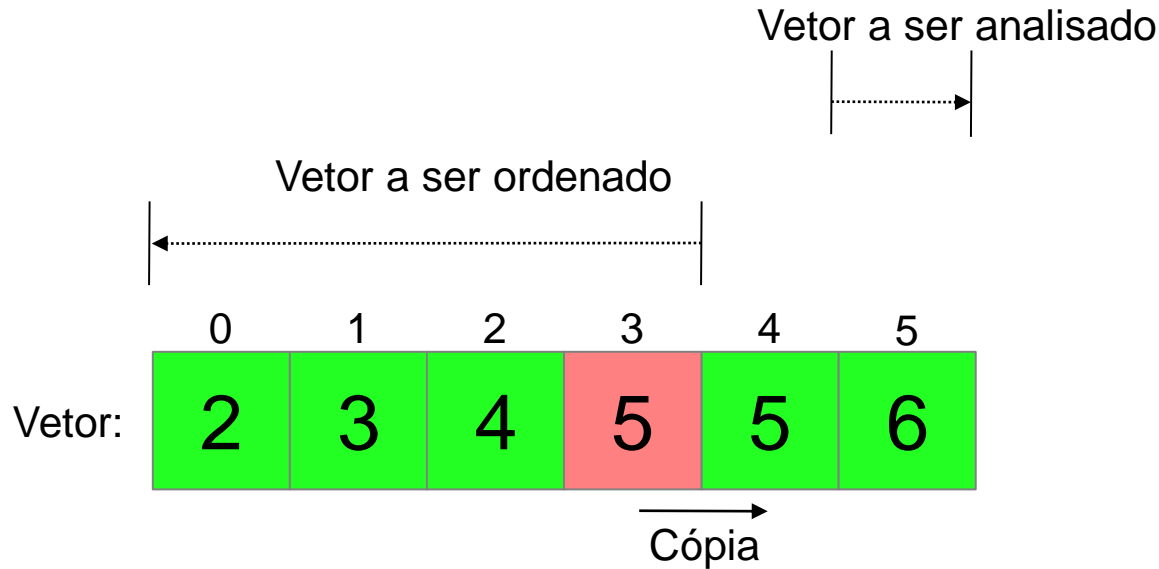
Se  $5 > 1$ , desloca;  
Senão, mantém.

**Portanto, desloca-se!**

Aplica-se a condição ao: Índice 3

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



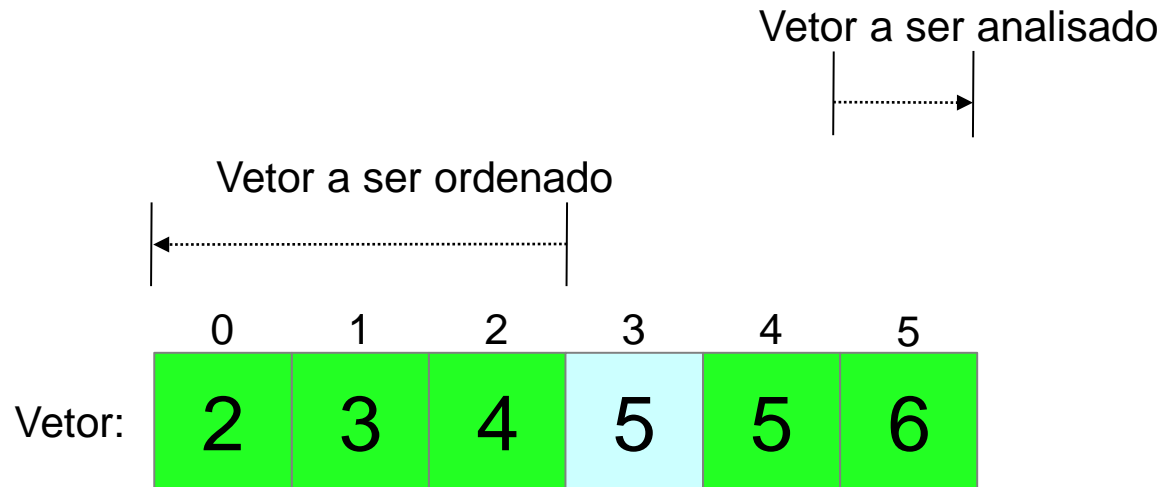
Rascunho:

1

Desloca-se: do índice 3 para o índice 4

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



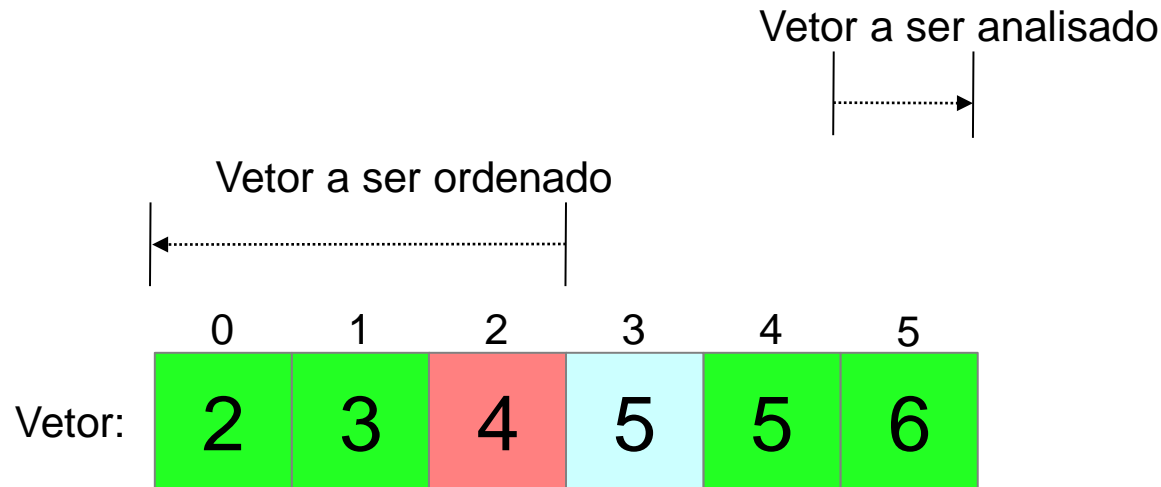
Rascunho:

1

Vetor a ser ordenado: do índice 0 ao índice 2

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



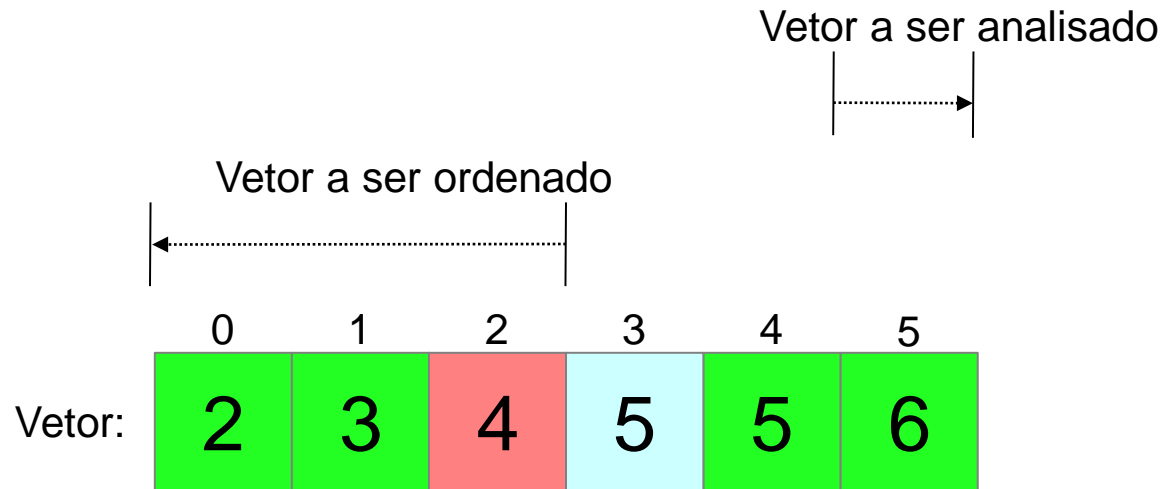
Rascunho:

1

Elemento do vetor a ser comparado: Índice 2

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

1

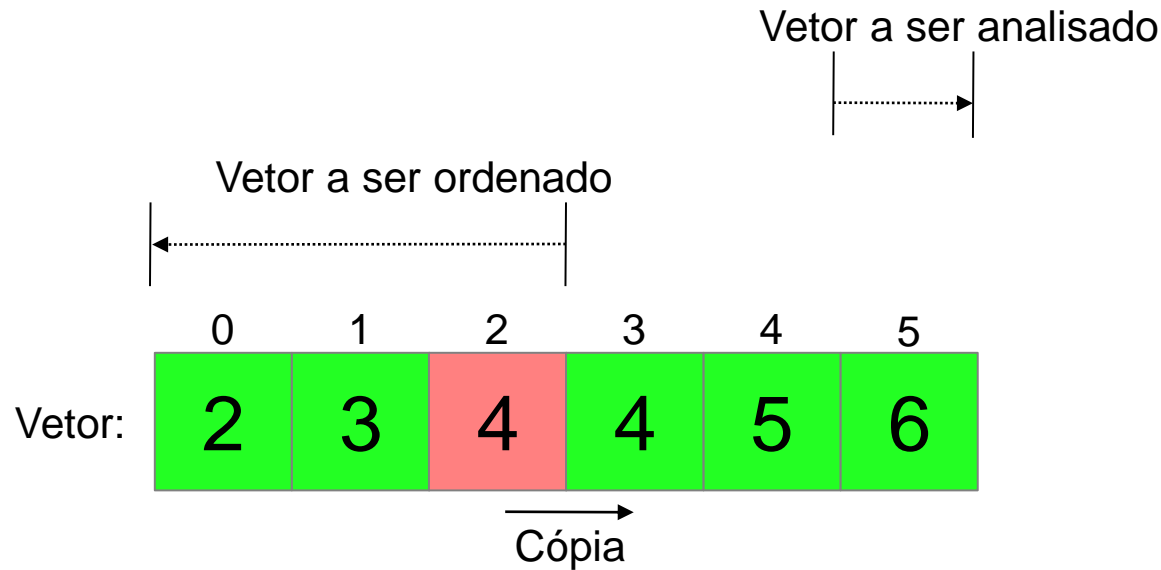
Se  $4 > 1$ , desloca;  
Senão, mantém.

**Portanto, desloca-se!**

Aplica-se a condição ao: Índice 2

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

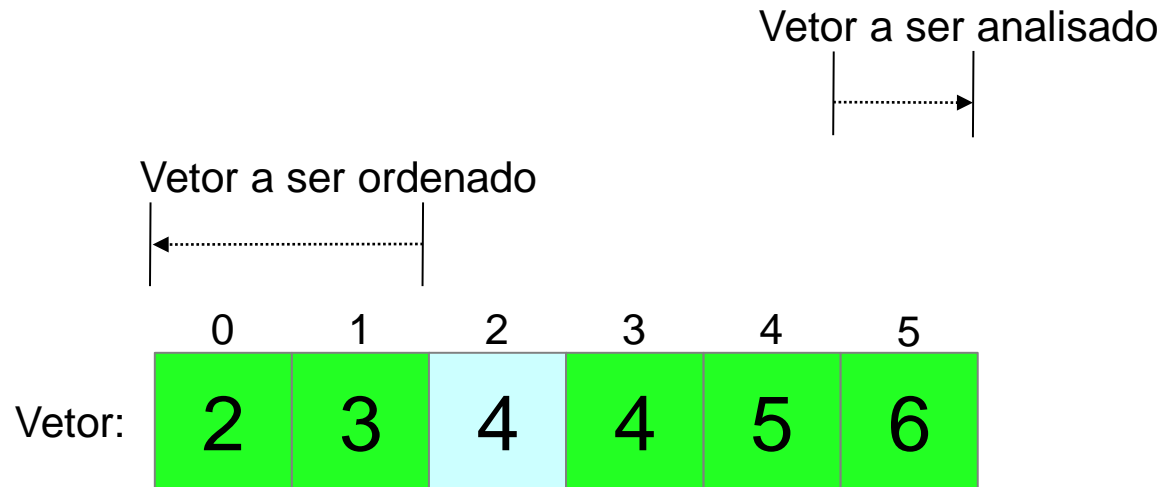
1

Desloca-se: do índice 2 para o índice 3



# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



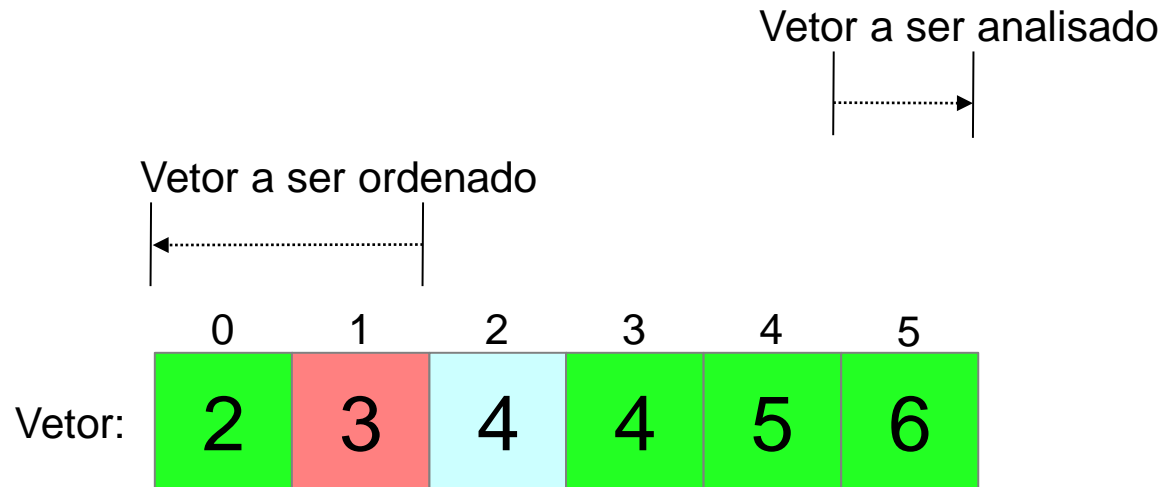
Rascunho:

1

Vetor a ser ordenado: do índice 0 ao índice 1

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



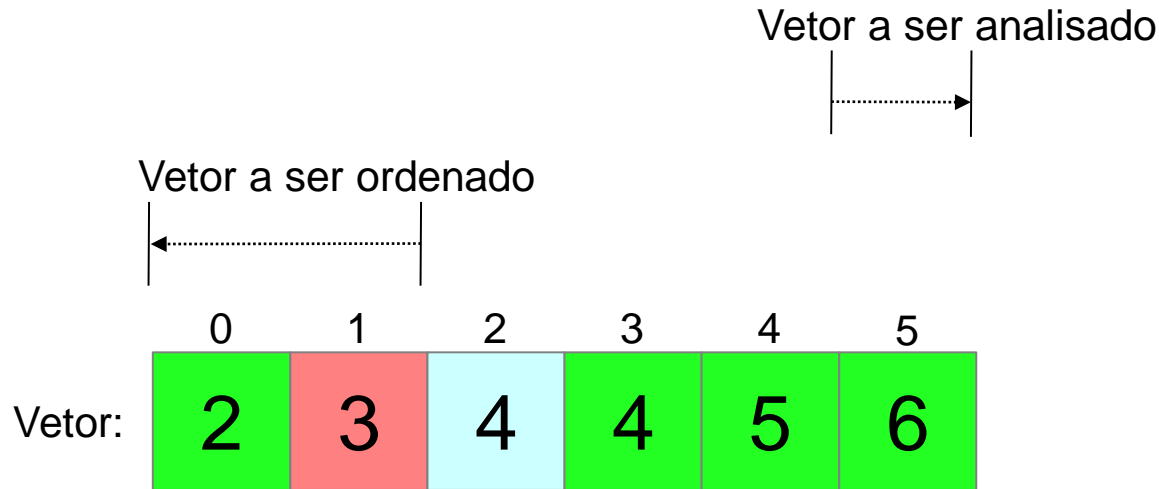
Rascunho:

1

Elemento do vetor a ser comparado: Índice 1

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

1

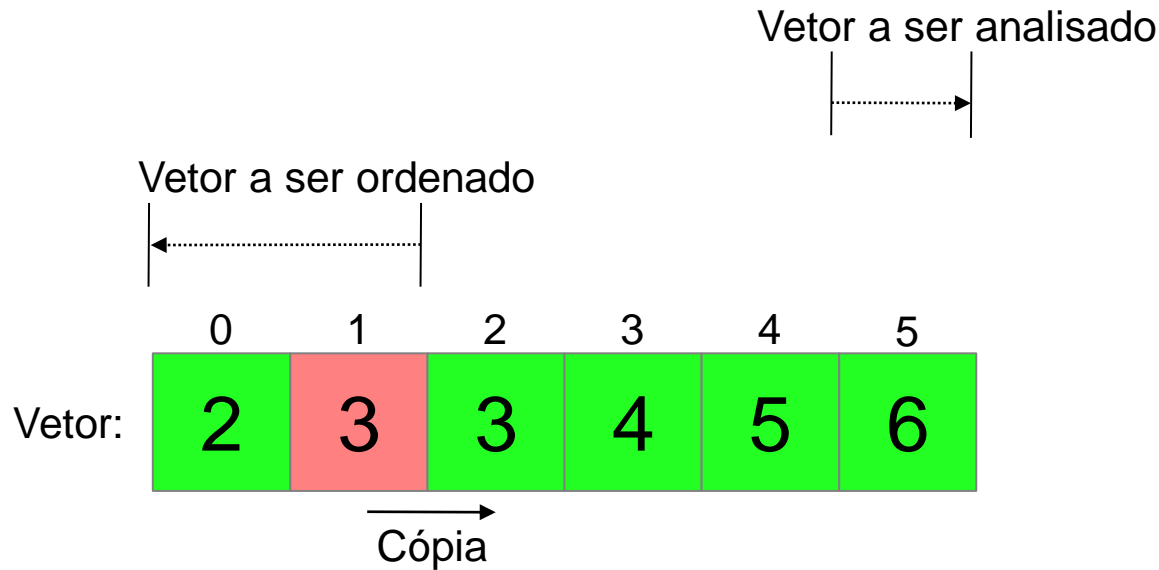
Se  $3 > 1$ , desloca;  
Senão, mantém.

**Portanto, desloca-se!**

Aplica-se a condição ao: Índice 1

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



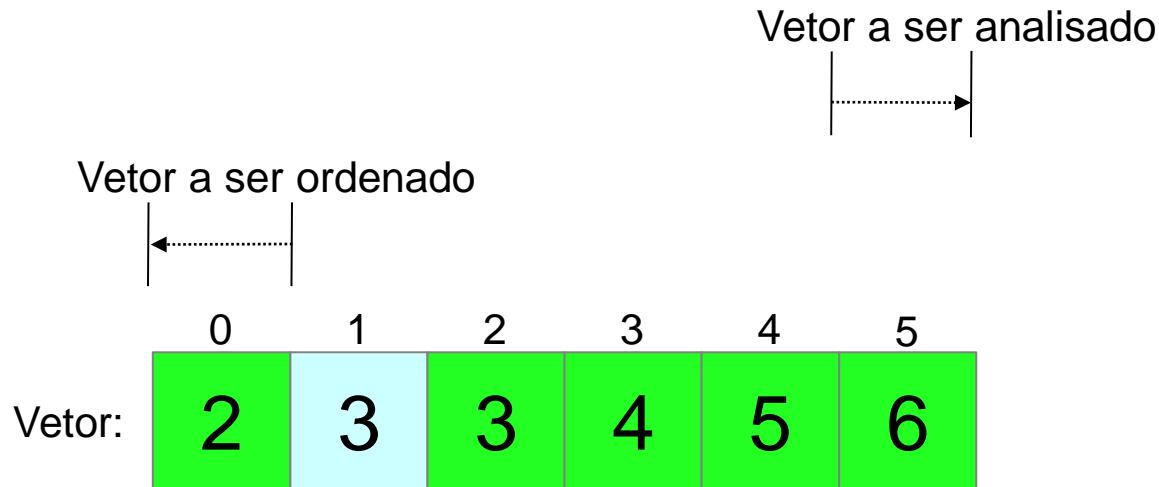
Rascunho:

1

Desloca-se: do índice 1 para o índice 2

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



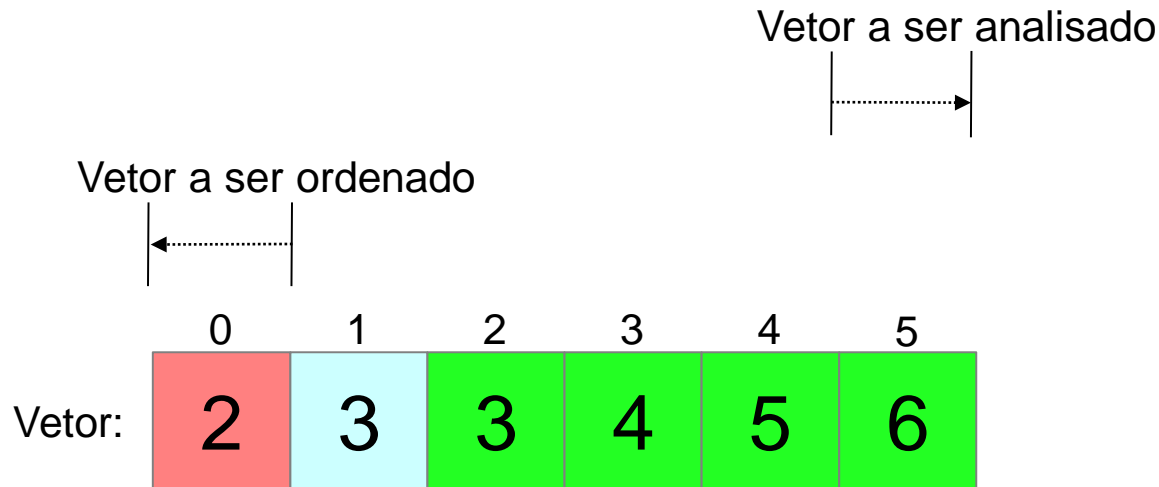
Rascunho:

1

Vetor a ser ordenado: do índice 0 ao índice 0

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

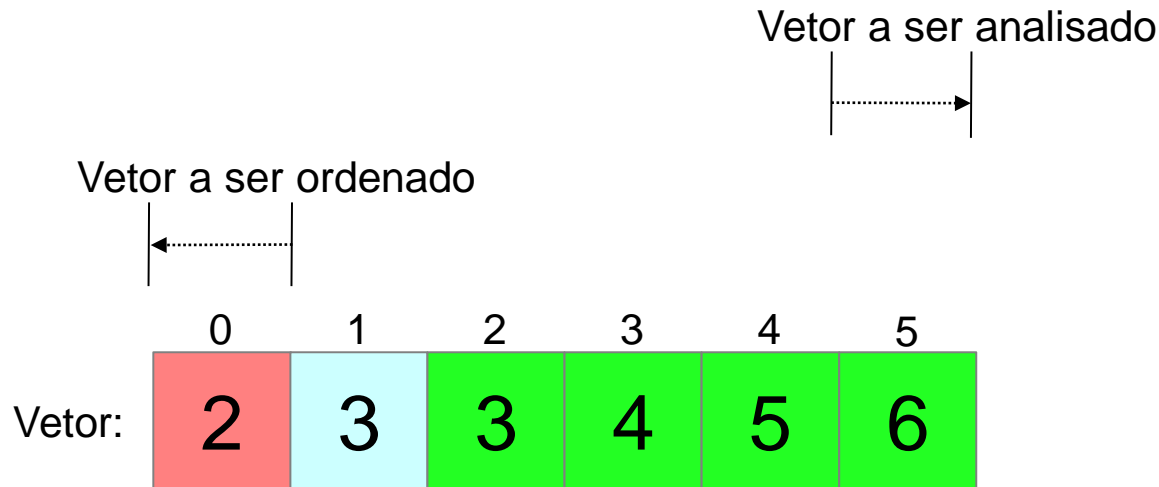


Rascunho: 1

Elemento do vetor a ser comparado: Índice 0

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

1

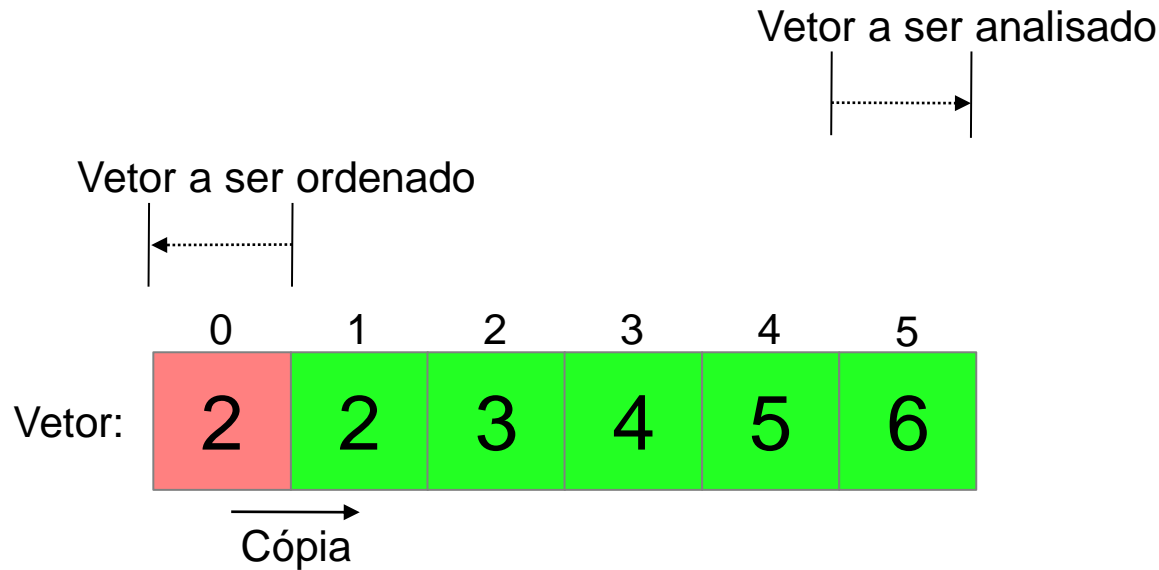
Se  $2 > 1$ , desloca;  
Senão, mantém.

**Portanto, desloca-se!**

Aplica-se a condição ao: Índice 0

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

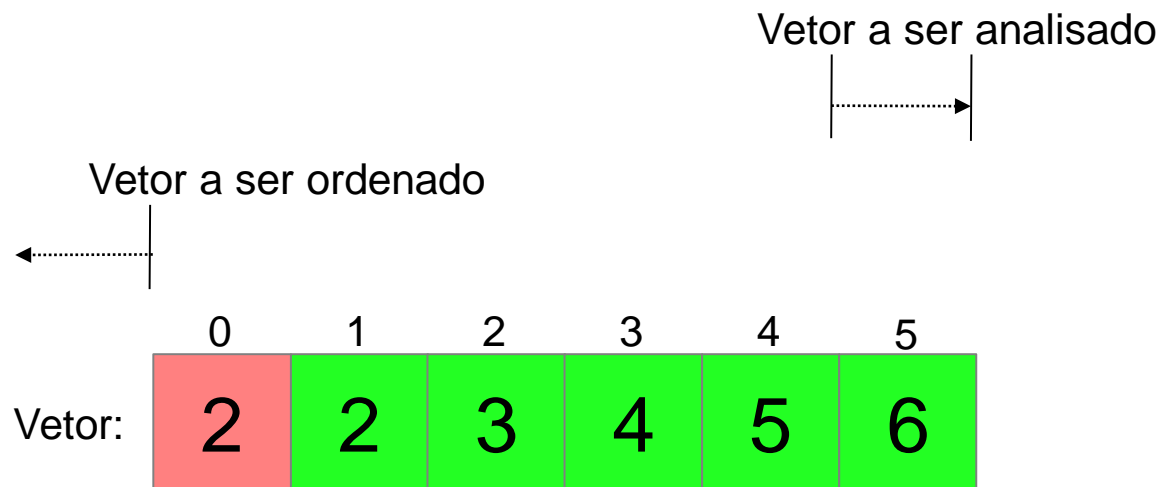
1

Desloca-se: do índice 0 para o índice 1



# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort



Rascunho:

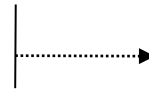
1

Vetor a ser ordenado: encerrado!

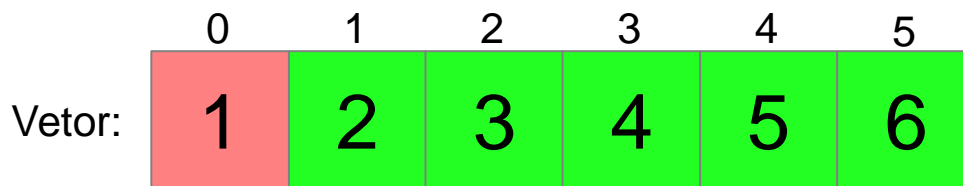
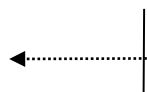
# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor a ser analisado



Vetor a ser ordenado



Atribuição

Rascunho:



Retorna o elemento armazenado: Índice 0

# Estruturas de Dados e Análise de Algoritmos

## Ordenação por Inserção – Insert Sort

Vetor:

0	1	2	3	4	5
1	2	3	4	5	6

Vetor Ordenado!

# Estruturas de Dados e Análise de Algoritmos

## *Ordenação por Inserção – Insert Sort*

```
public static void OrdenaçãoInserçãoInt(int iVet[])
{
    int    iA,iB;
    int    iT;

    for(iA=1; iA < iVet.length; iA++)
    {
        iT=iVet[iA];
        iB=iA-1;
        while(iB >= 0 && iT < iVet[iB])
        {
            iVet[iB+1]=iVet[iB];
            iB--;
        }
        iVet[iB+1]=iT;
    }
}
```

# Estruturas de Dados e Análise de Algoritmos

## *Ordenação por Inserção – Insert Sort*

FIM