

Ordenação por Divisão e Conquista Merge-sort

Estruturas de Dados e Análise de Algoritmos

Ordenação Por Divisão e Conquista - Merge Sort

- Algoritmo de Ordenação por Divisão e Conquista, ou *Merge Sort*;
- Baseia-se na ideia de se dividir um vetor em vetores menores, até se obter vários pares de vetores unitários, o que simplifica a ordenação entre os esses pares;
- Algoritmo recursivo, utiliza-se da estratégia “Dividir para Conquistar” (*Divide and Conquer*).

Estruturas de Dados e Análise de Algoritmos

Ordenação Por Divisão e Conquista - Merge Sort

- Sua complexidade no tempo é: $O(n \log_2 n)$;
- Sua complexidade no espaço é: $O(n)$.

Estruturas de Dados e Análise de Algoritmos

Ordenação Por Divisão e Conquista - Merge 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 Divisão e Conquista - Merge Sort

Vetor:

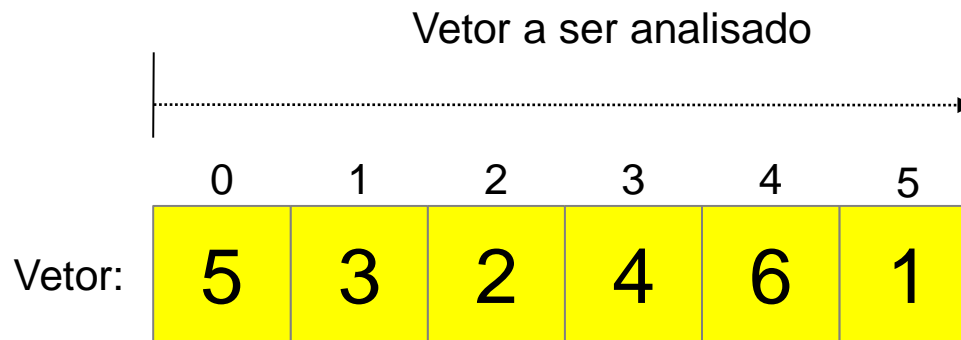
0	1	2	3	4	5
5	3	2	4	6	1



O algoritmo ordenará o vetor crescentemente

Estruturas de Dados e Análise de Algoritmos

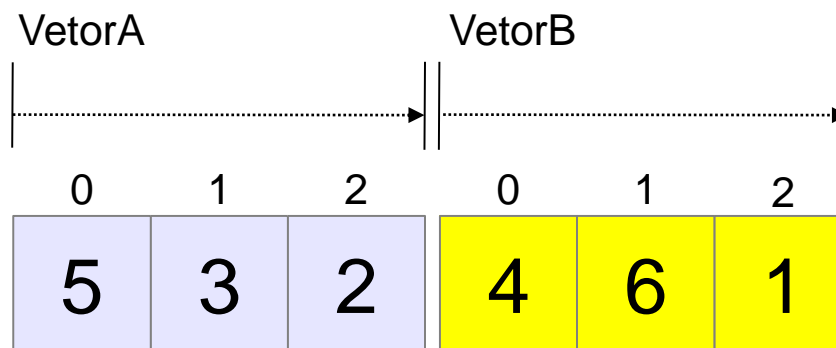
Ordenação Por Divisão e Conquista - Merge Sort



Etapa 1: Dividir o vetor em outros dois

Estruturas de Dados e Análise de Algoritmos

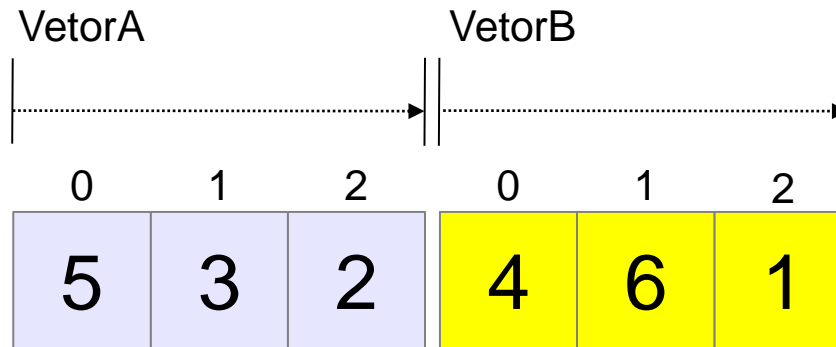
Ordenação Por Divisão e Conquista - Merge Sort



VetorA e VetorB

Estruturas de Dados e Análise de Algoritmos

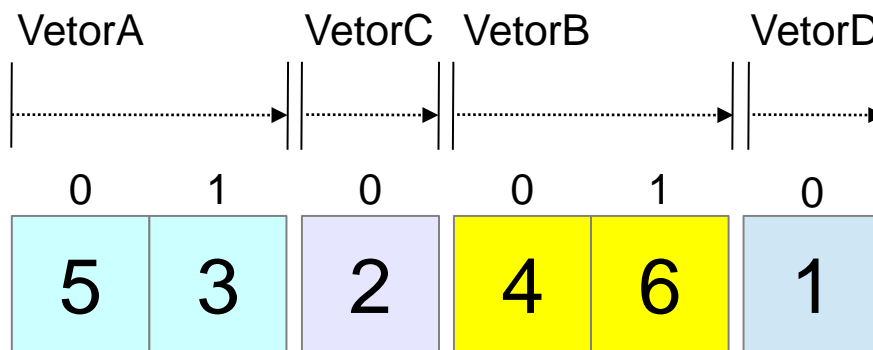
Ordenação Por Divisão e Conquista - Merge Sort



Etapa 2: Dividir cada vetor em outros dois

Estruturas de Dados e Análise de Algoritmos

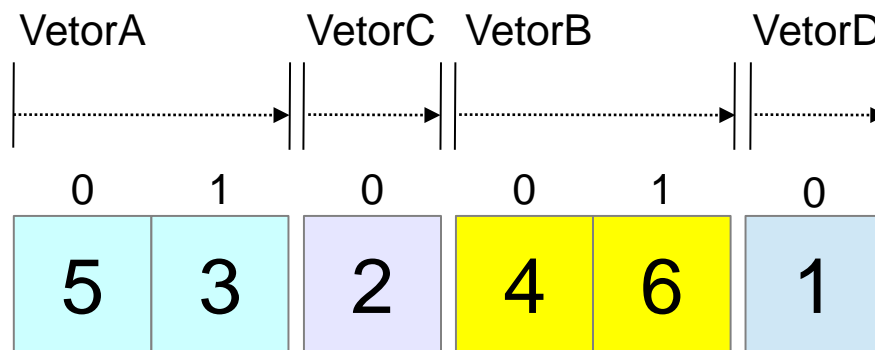
Ordenação Por Divisão e Conquista - Merge Sort



VetorA, VetorC, VetorB e VetorD

Estruturas de Dados e Análise de Algoritmos

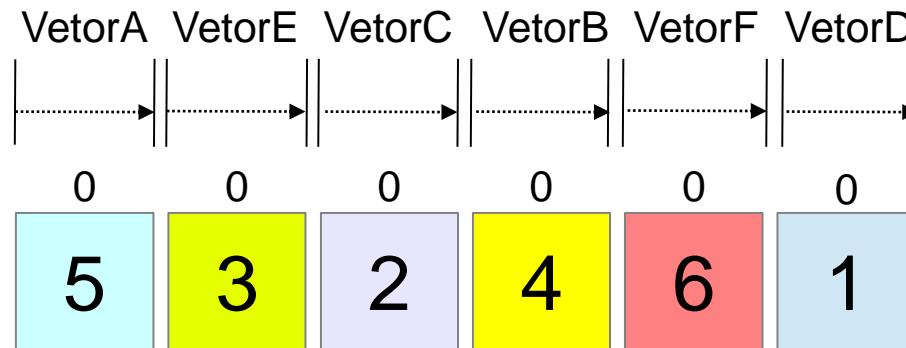
Ordenação Por Divisão e Conquista - Merge Sort



Etapa 3: Dividir cada vetor em outros dois

Estruturas de Dados e Análise de Algoritmos

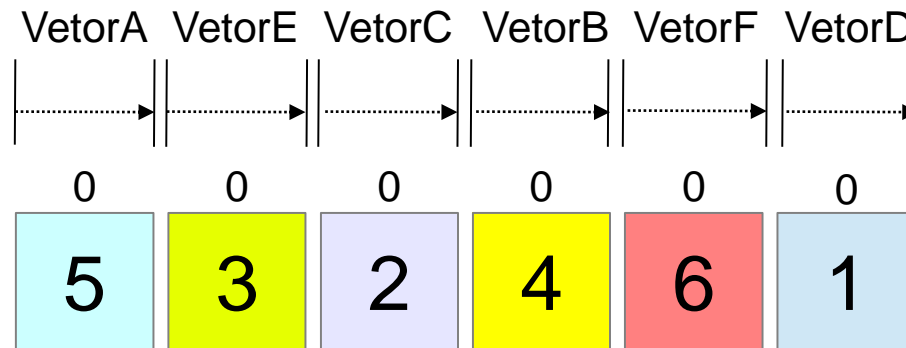
Ordenação Por Divisão e Conquista - Merge Sort



VetorA, VetorE, VetorC, VetorB, VetorF e VetorD

Estruturas de Dados e Análise de Algoritmos

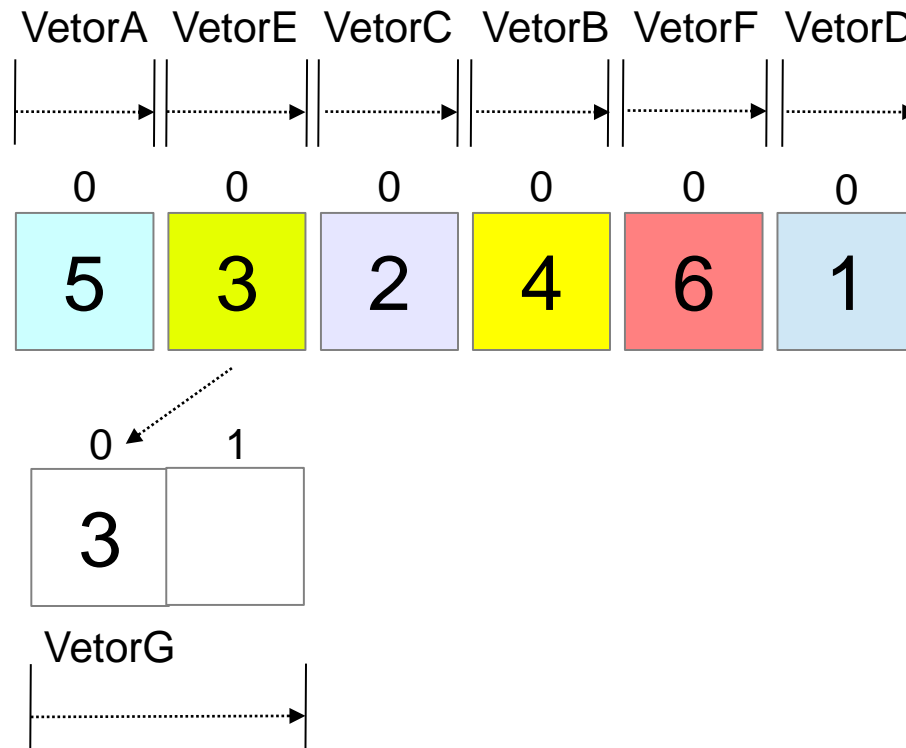
Ordenação Por Divisão e Conquista - Merge Sort



Etapa 4: Mesclar Ordenadamente os Pares

Estruturas de Dados e Análise de Algoritmos

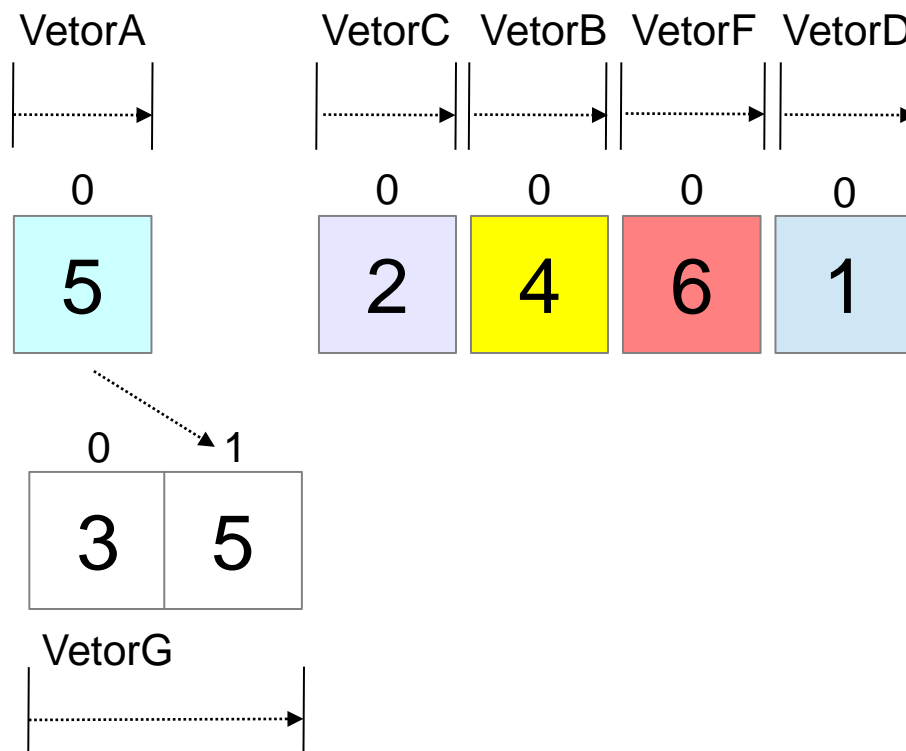
Ordenação Por Divisão e Conquista - Merge Sort



Cria VetorG para receber VetorA e VetorE

Estruturas de Dados e Análise de Algoritmos

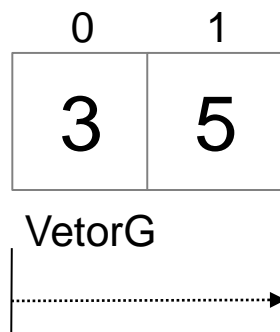
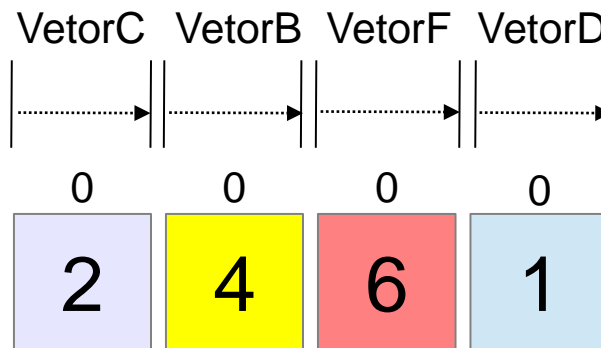
Ordenação Por Divisão e Conquista - Merge Sort



VetorG é mescla de VetorA e VetorE

Estruturas de Dados e Análise de Algoritmos

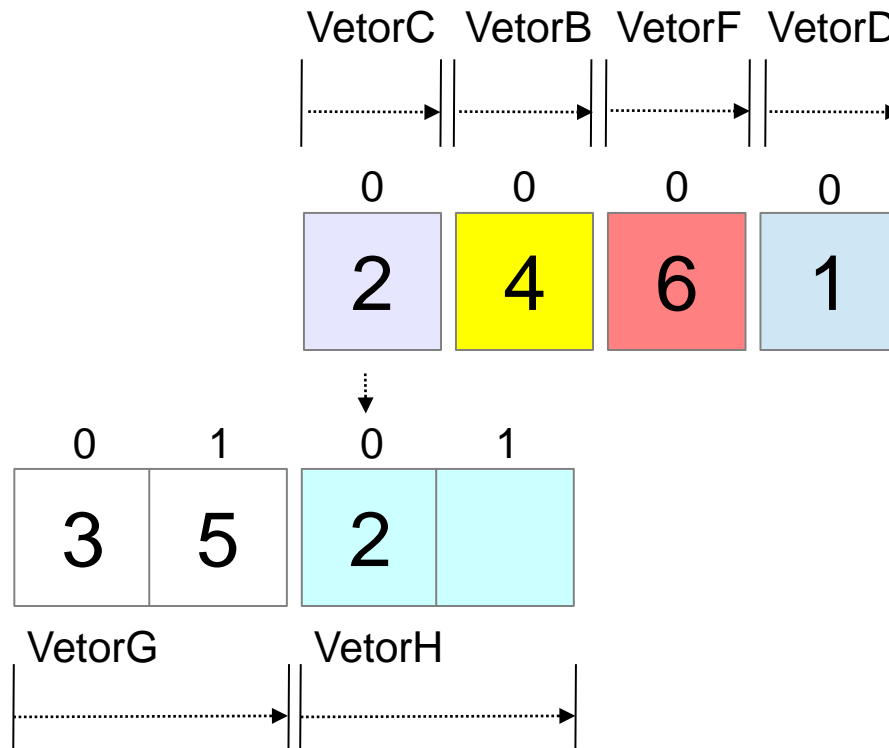
Ordenação Por Divisão e Conquista - Merge Sort



VetorG Ordenado!

Estruturas de Dados e Análise de Algoritmos

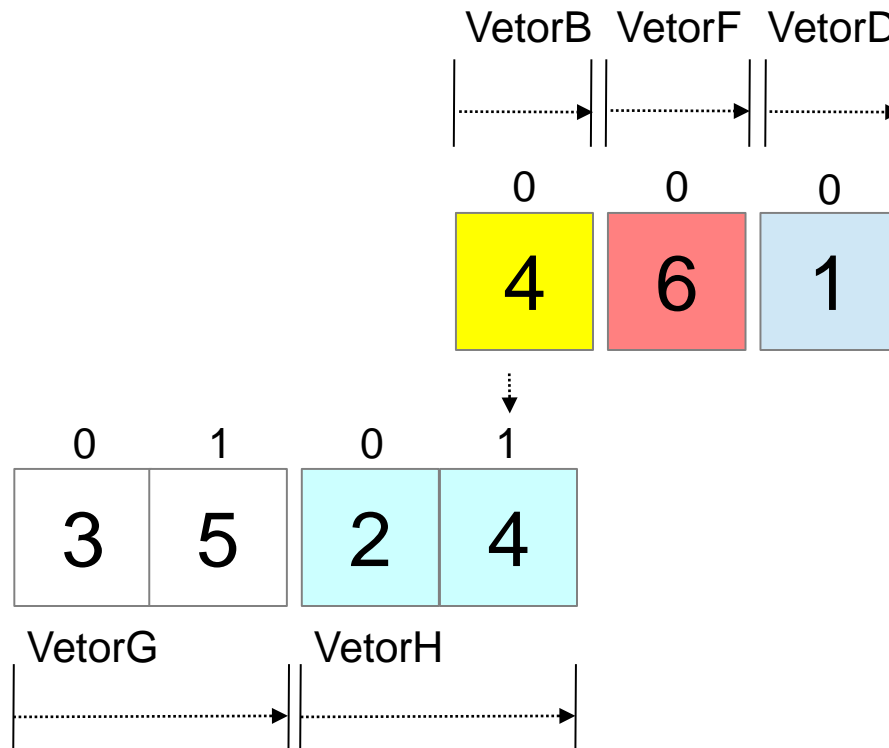
Ordenação Por Divisão e Conquista - Merge Sort



Cria VetorH para receber VetorC e VetorB

Estruturas de Dados e Análise de Algoritmos

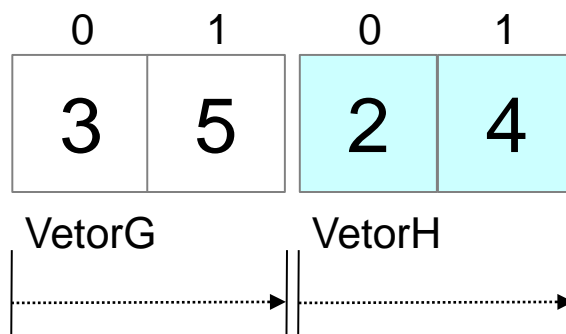
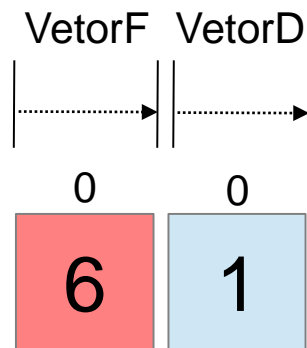
Ordenação Por Divisão e Conquista - Merge Sort



VetorH é mescla de VetorC e VetorB

Estruturas de Dados e Análise de Algoritmos

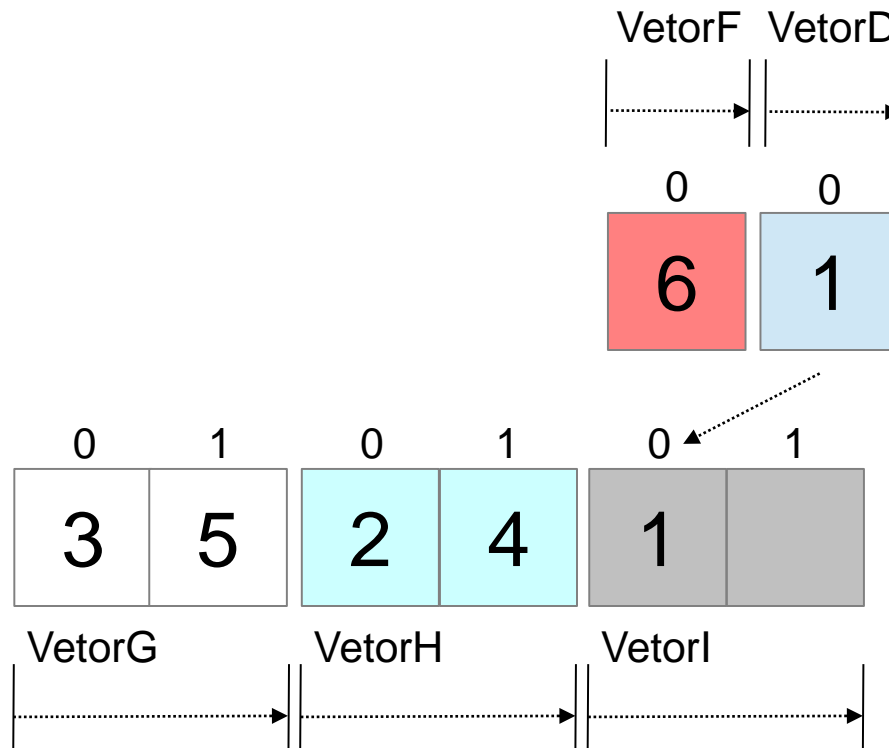
Ordenação Por Divisão e Conquista - Merge Sort



VetorH Ordenado!

Estruturas de Dados e Análise de Algoritmos

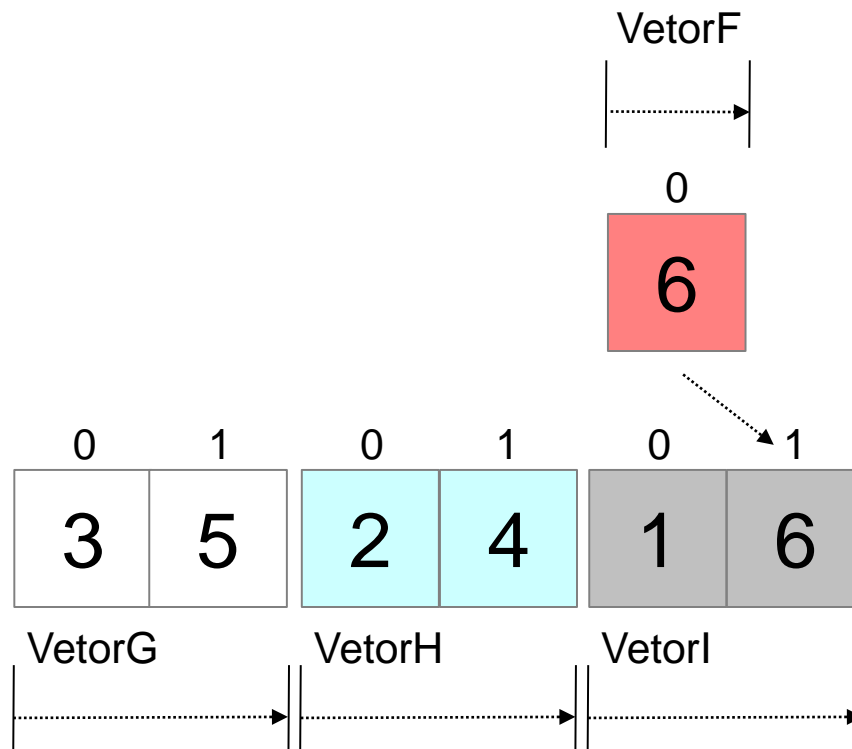
Ordenação Por Divisão e Conquista - Merge Sort



Cria VetorI para receber VetorF e VetorD

Estruturas de Dados e Análise de Algoritmos

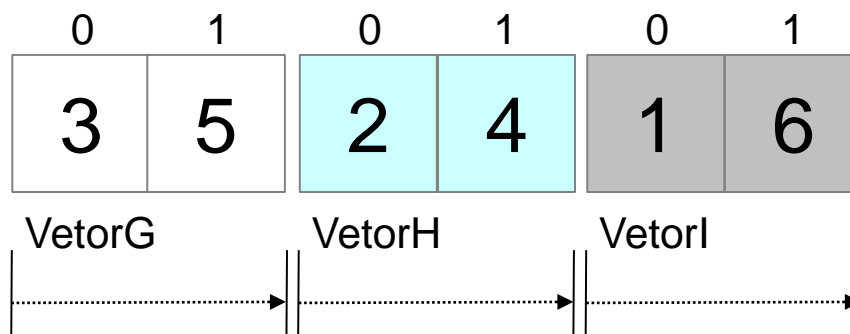
Ordenação Por Divisão e Conquista - Merge Sort



VetorI é mescla de VetorF e VetorD

Estruturas de Dados e Análise de Algoritmos

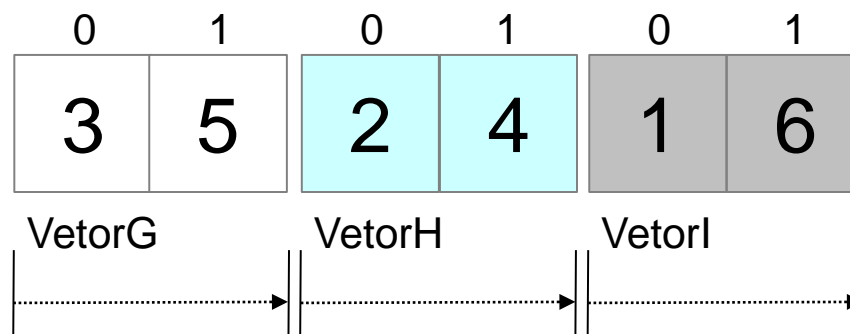
Ordenação Por Divisão e Conquista - Merge Sort



VetorI Ordenado!

Estruturas de Dados e Análise de Algoritmos

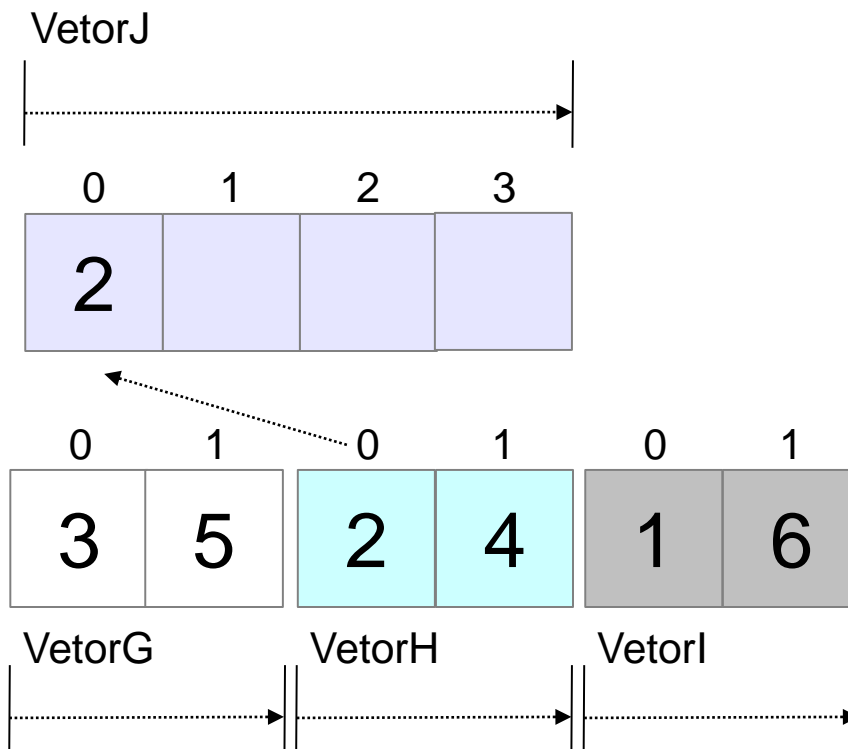
Ordenação Por Divisão e Conquista - Merge Sort



Etapa 5: Mesclar Ordenadamente os Pares

Estruturas de Dados e Análise de Algoritmos

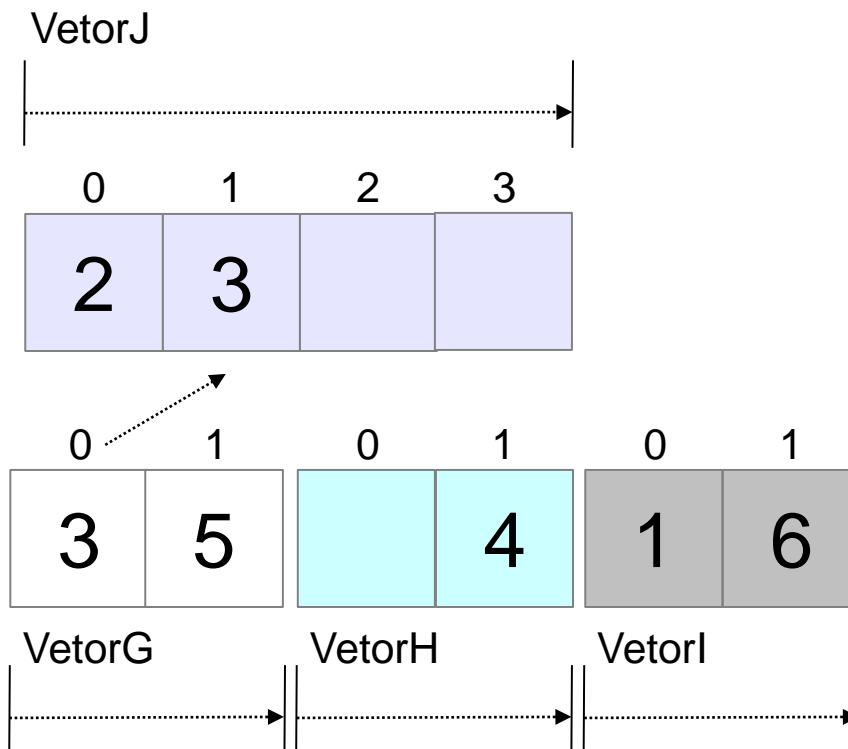
Ordenação Por Divisão e Conquista - Merge Sort



Cria VetorJ para receber VetorG e VetorH

Estruturas de Dados e Análise de Algoritmos

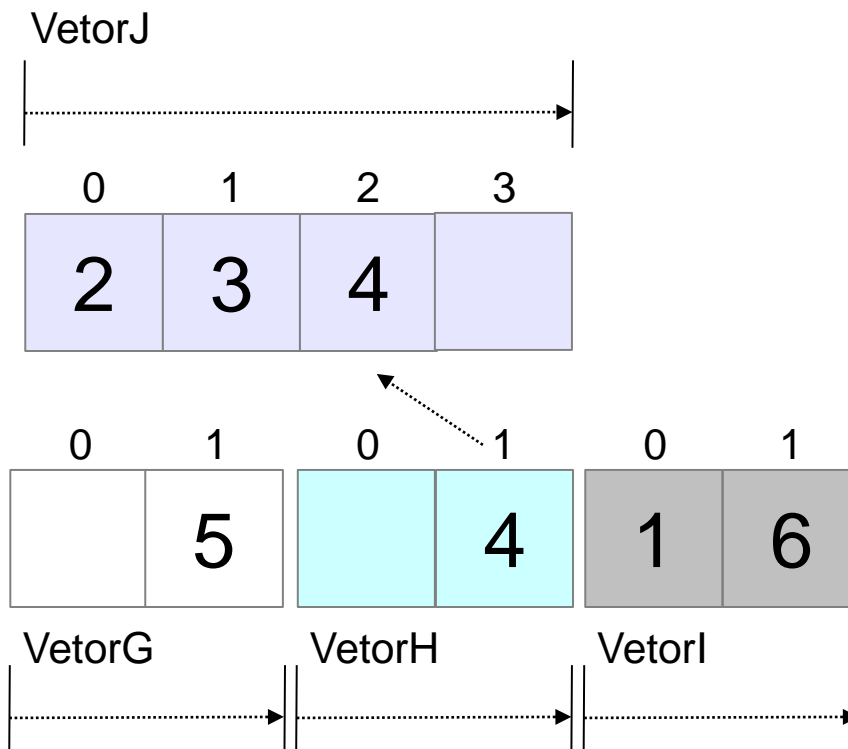
Ordenação Por Divisão e Conquista - Merge Sort



VetorJ é mescla de VetorG e VetorH

Estruturas de Dados e Análise de Algoritmos

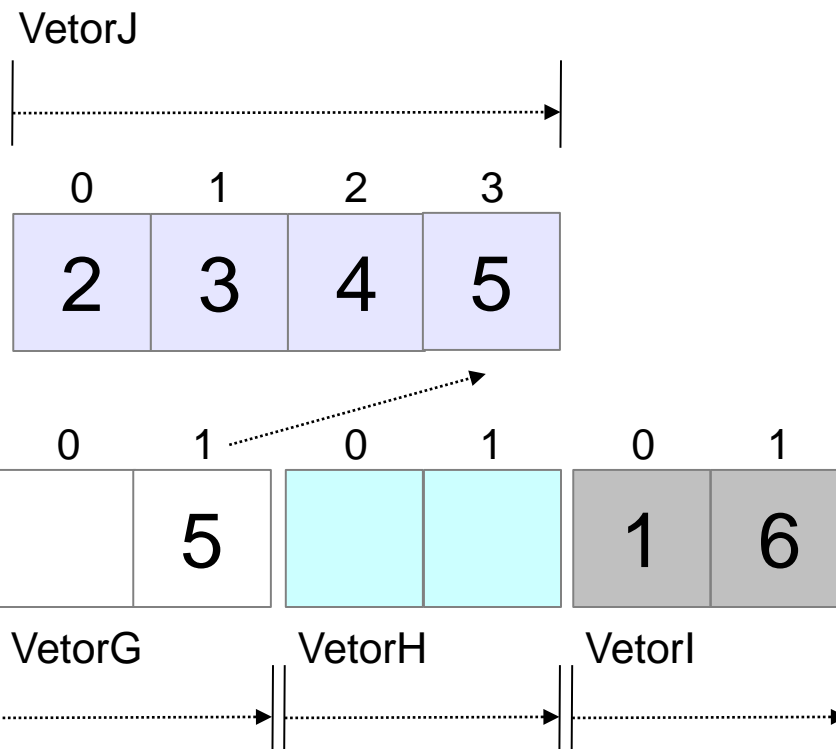
Ordenação Por Divisão e Conquista - Merge Sort



VetorJ é mescla de VetorG e VetorH

Estruturas de Dados e Análise de Algoritmos

Ordenação Por Divisão e Conquista - Merge Sort



VetorJ é mescla de VetorG e VetorH

Estruturas de Dados e Análise de Algoritmos

Ordenação Por Divisão e Conquista - Merge Sort

VetorJ

0	1	2	3
2	3	4	5

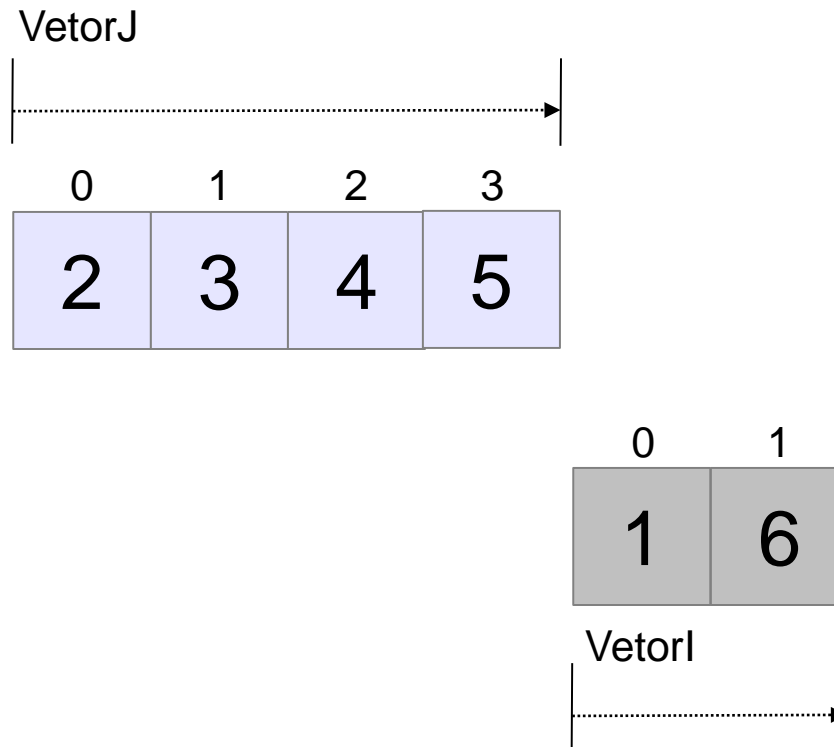
0	1
1	6

VetorI

VetorJ Ordenado!

Estruturas de Dados e Análise de Algoritmos

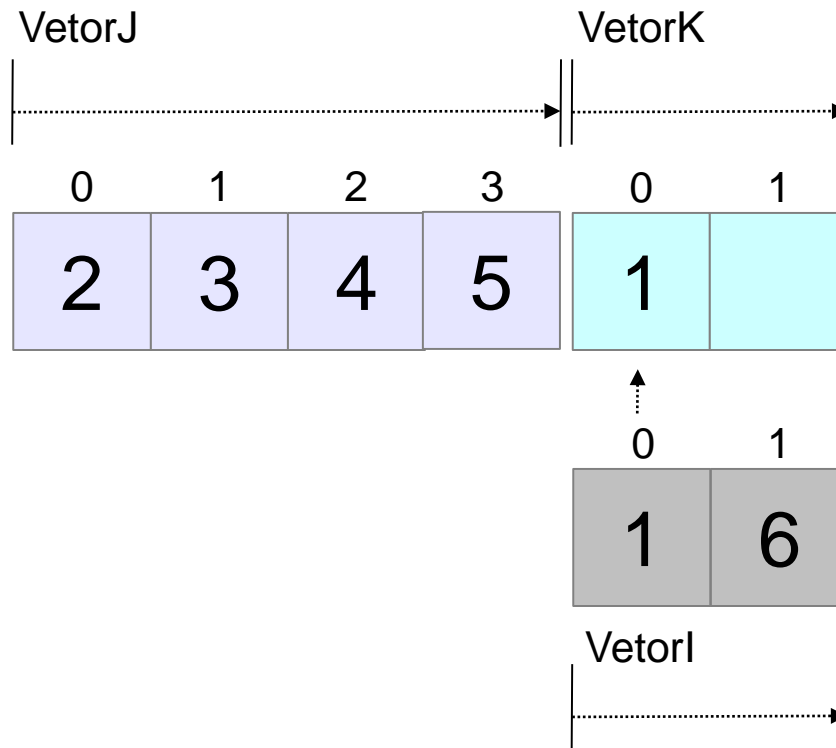
Ordenação Por Divisão e Conquista - Merge Sort



Etapa 6: Mesclar Ordenadamente os Pares

Estruturas de Dados e Análise de Algoritmos

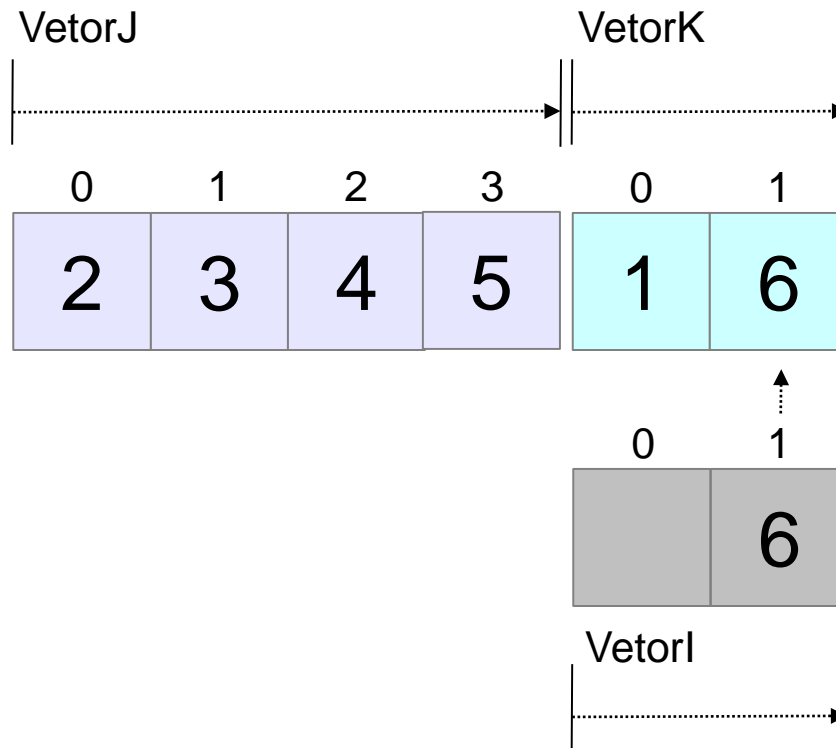
Ordenação Por Divisão e Conquista - Merge Sort



Cria VetorK para receber VetorI

Estruturas de Dados e Análise de Algoritmos

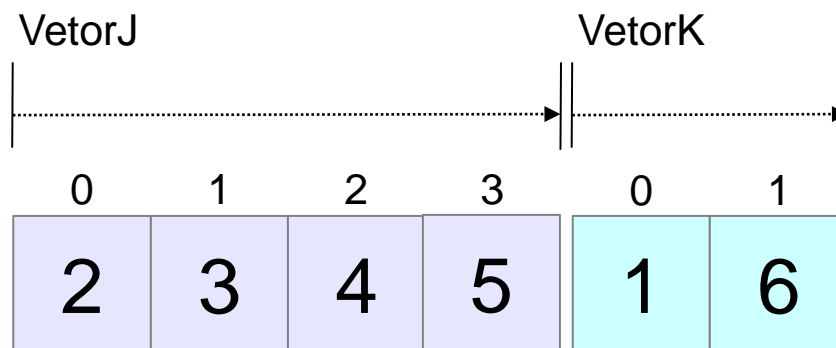
Ordenação Por Divisão e Conquista - Merge Sort



VetorK é o VetorI

Estruturas de Dados e Análise de Algoritmos

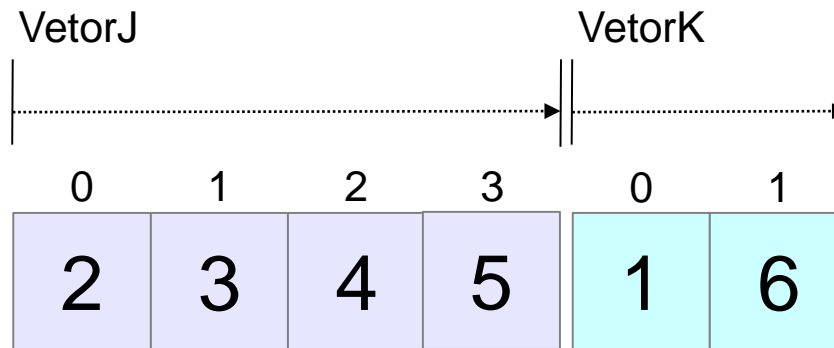
Ordenação Por Divisão e Conquista - Merge Sort



VetorK Ordenado!

Estruturas de Dados e Análise de Algoritmos

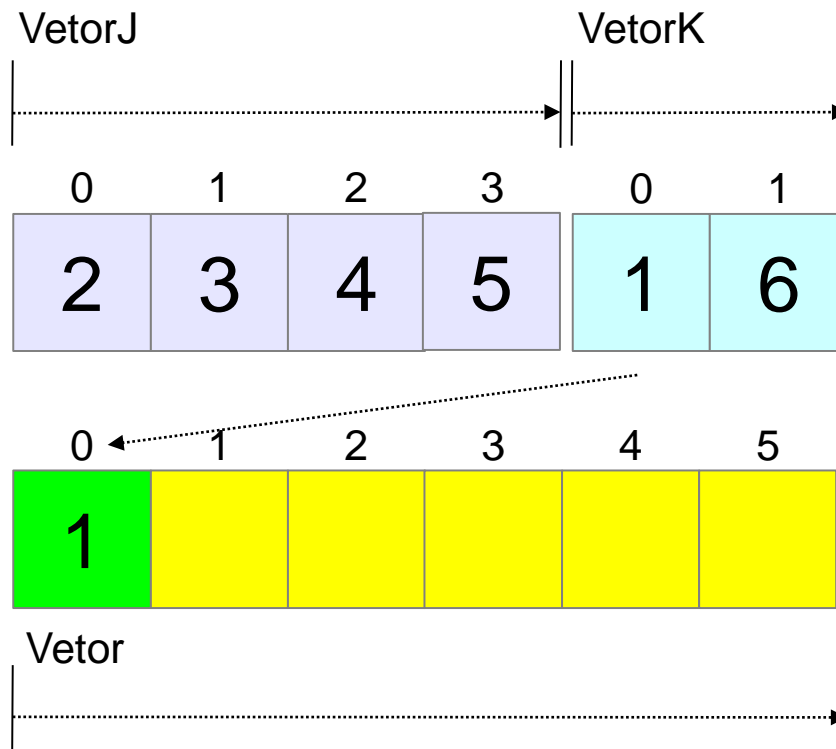
Ordenação Por Divisão e Conquista - Merge Sort



Etapa 7: Mesclar Ordenadamente os Pares

Estruturas de Dados e Análise de Algoritmos

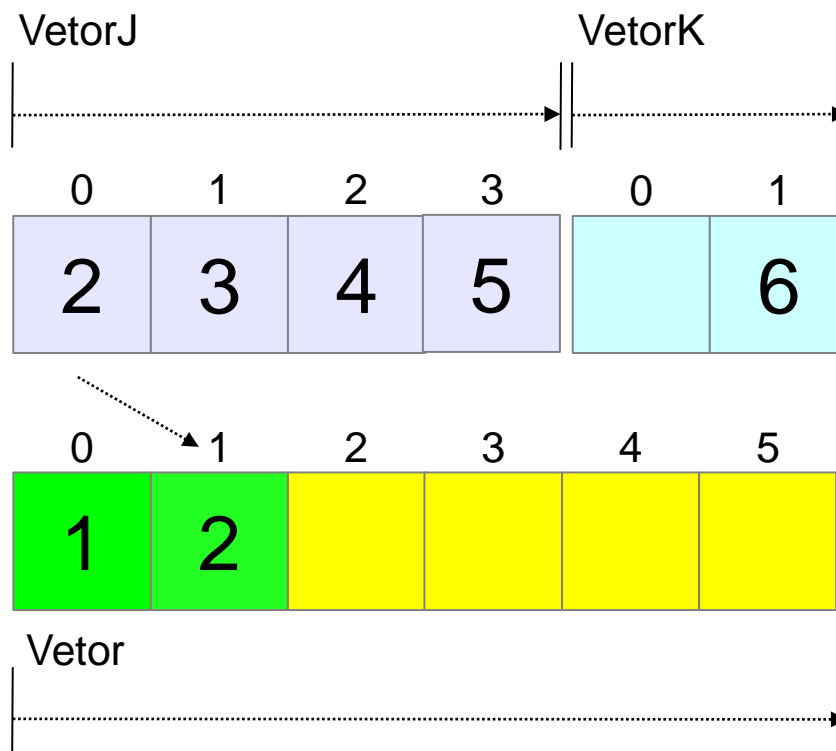
Ordenação Por Divisão e Conquista - Merge Sort



Usa Vetor para receber VetorJ e VetorK

Estruturas de Dados e Análise de Algoritmos

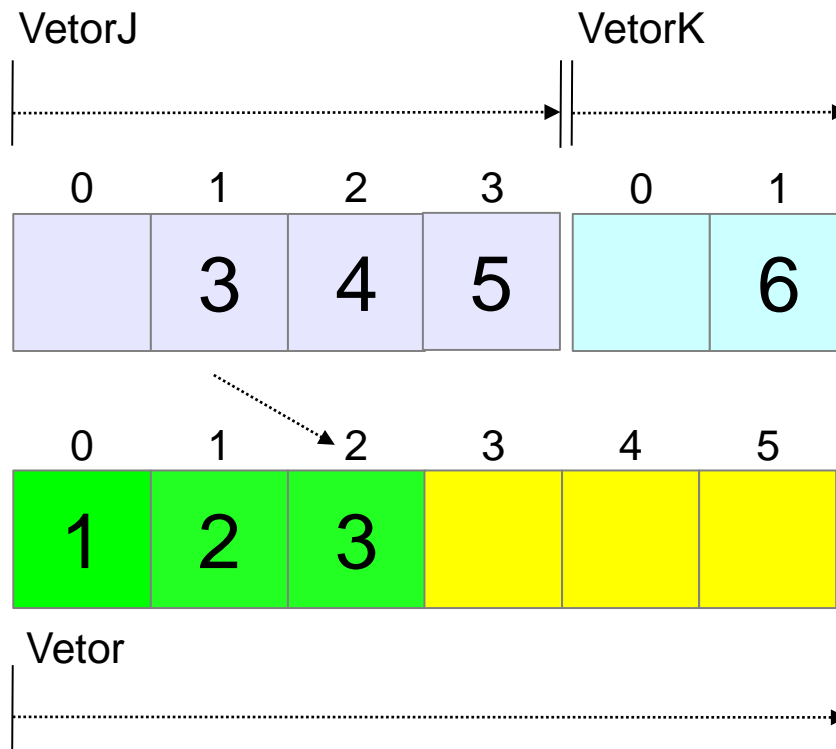
Ordenação Por Divisão e Conquista - Merge Sort



Vetor é mescla de VetorJ e VetorK

Estruturas de Dados e Análise de Algoritmos

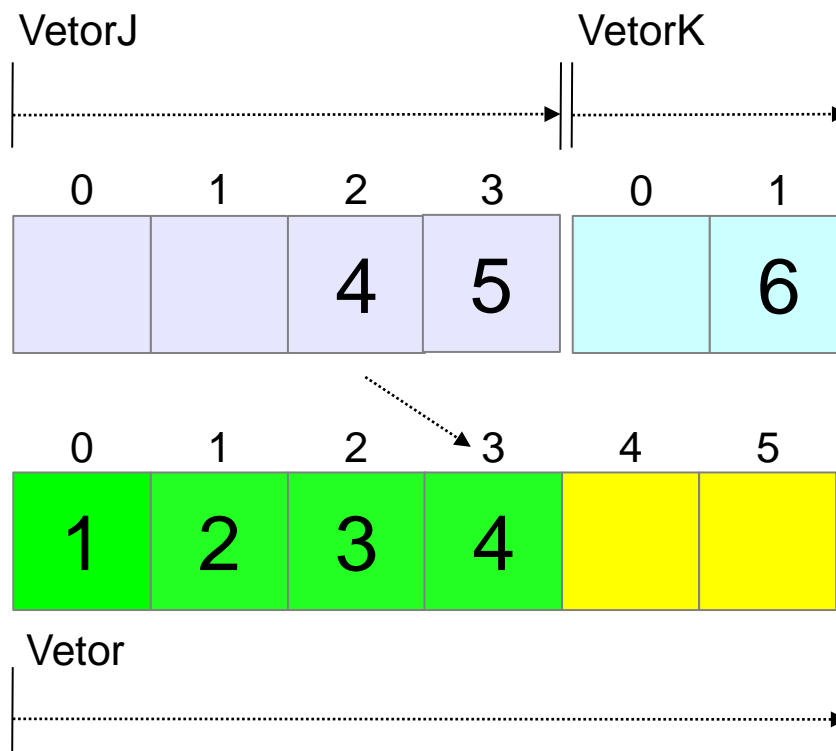
Ordenação Por Divisão e Conquista - Merge Sort



Vetor é mescla de VetorJ e VetorK

Estruturas de Dados e Análise de Algoritmos

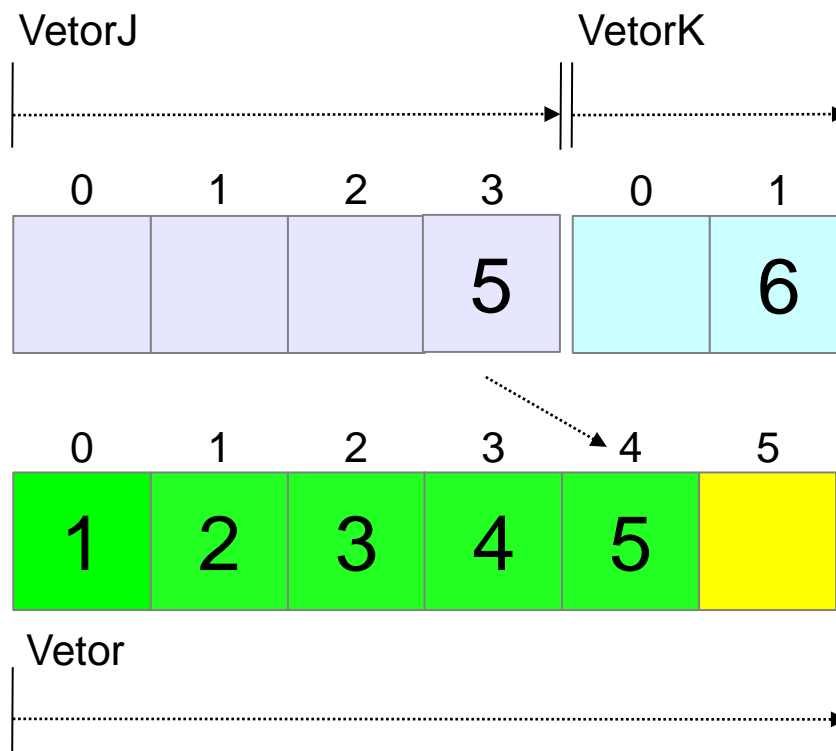
Ordenação Por Divisão e Conquista - Merge Sort



Vetor é mescla de VetorJ e VetorK

Estruturas de Dados e Análise de Algoritmos

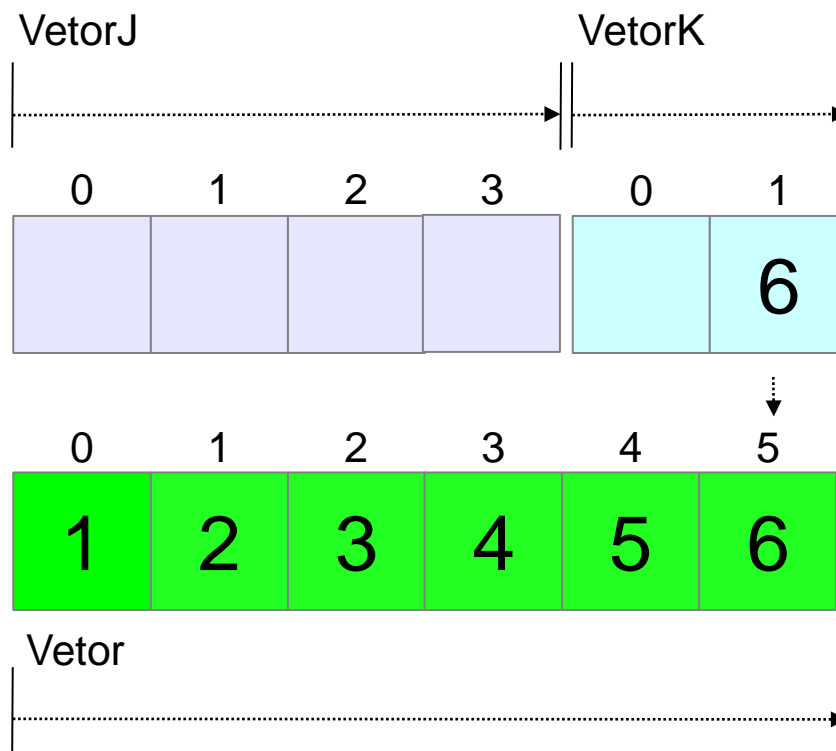
Ordenação Por Divisão e Conquista - Merge Sort



Vetor é mescla de VetorJ e VetorK

Estruturas de Dados e Análise de Algoritmos

Ordenação Por Divisão e Conquista - Merge Sort



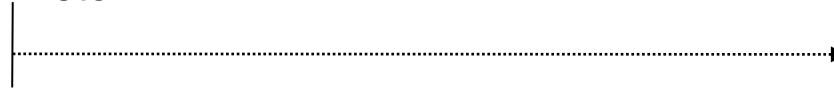
Vetor é mescla de VetorJ e VetorK

Estruturas de Dados e Análise de Algoritmos

Ordenação Por Divisão e Conquista - Merge Sort

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

Vetor



Vetor Ordenado!

Estruturas de Dados e Análise de Algoritmos

Ordenação Por Divisão e Conquista - Merge Sort

Vetor:

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

Vetor Original Ordenado!

Estruturas de Dados e Análise de Algoritmos

Ordenação Por Divisão e Conquista - Merge Sort

```
public static void mergeSort(int v[])
{
    if(v.length > 1)
    {
        int t1 = v.length/2;
        int t2 = v.length-t1;
        int v1[] = new int[t1];
        int v2[] = new int[t2];

        for(int i = 0; i < t1; i++)
        {
            v1[i] = v[i];
        }
        for(int i = t1; i < (t1+t2); i++)
        {
            v2[i-t1] = v[i];
        }
        mergeSort(v1);
        mergeSort(v2);
        merge(v, v1, v2);
    }
}
```

Estruturas de Dados e Análise de Algoritmos

Ordenação Por Divisão e Conquista - Merge Sort

```
public static void merge(int w[], int w1[], int w2[])
{
    int i = 0, j = 0, k = 0;

    while(w1.length != j && w2.length != k)
    {
        if(w1[j] >= w2[k])
        {
            w[i] = w1[j];
            i++;
            j++;
        }
        else
        {
            w[i] = w2[k];
            i++;
            k++;
        }
    }
    while(w1.length != j)
    {
        w[i] = w1[j];
        i++;
        j++;
    }
    while(w2.length != k)
    {
        w[i] = w2[k];
        i++;
        k++;
    }
}
```

Estruturas de Dados e Análise de Algoritmos

Ordenação Por Divisão e Conquista - Merge Sort

FIM