MERGESORT

Algoritmos e Estrutura de Dados II

Prof. André Vignatti

INTERCALAÇÃO DE VETORES ORDENADOS

Intercalação de Vetores Ordenados	
Instância:	(v, a, m, b) onde $v[a..m]$ e $v[m+1..b]$ são ambos vetores ordenados.
Resposta:	o vetor v modificado de tal forma que $v[a..b]$ é um vetor ordenado.

Intercala(v,a,m,b)

```
Se a ≥ b
    Devolva v
i ← a
j ← m + 1
Para k ← 0 to b − a
    Se j > b ou (i ≤ m e v[i] ≤ v[j])
        u[k] ← v[i]
        i ← i + 1
    Senão
        u[k] ← v[j]
        j ← j + 1
v[a..b] ← u[0..b − a]
Devolva v
```

Executar no vetor $v = [2, 7, 15|3, 5, 20]$, $v = [4, 5, 6|1, 2, 3]$ e $v[1, 2, 3|4, 5, 6]$

```
Intercala(v,a,m,b)
---------------------------------------------
Se a ≥ b
    Devolva v
i ← a
j ← m + 1
Para k ← 0 to b − a
    Se j > b ou (i ≤ m e v[i] ≤ v[j])
        u[k] ← v[i]
        i ← i + 1
    Senão
        u[k] ← v[j]
        j ← j + 1
v[a..b] ← u[0..b − a]
Devolva v
```

INTERCALA: ANÁLISE

$C_I(n)$ é o número de comparações do Intercala para vetores de tamanho n

Toda vez execução da iteração (**para**)

o **se** é executado

é feita uma comparação com elemento de vetor

Assim,

$$\boxed{C_I(n) = n}$$

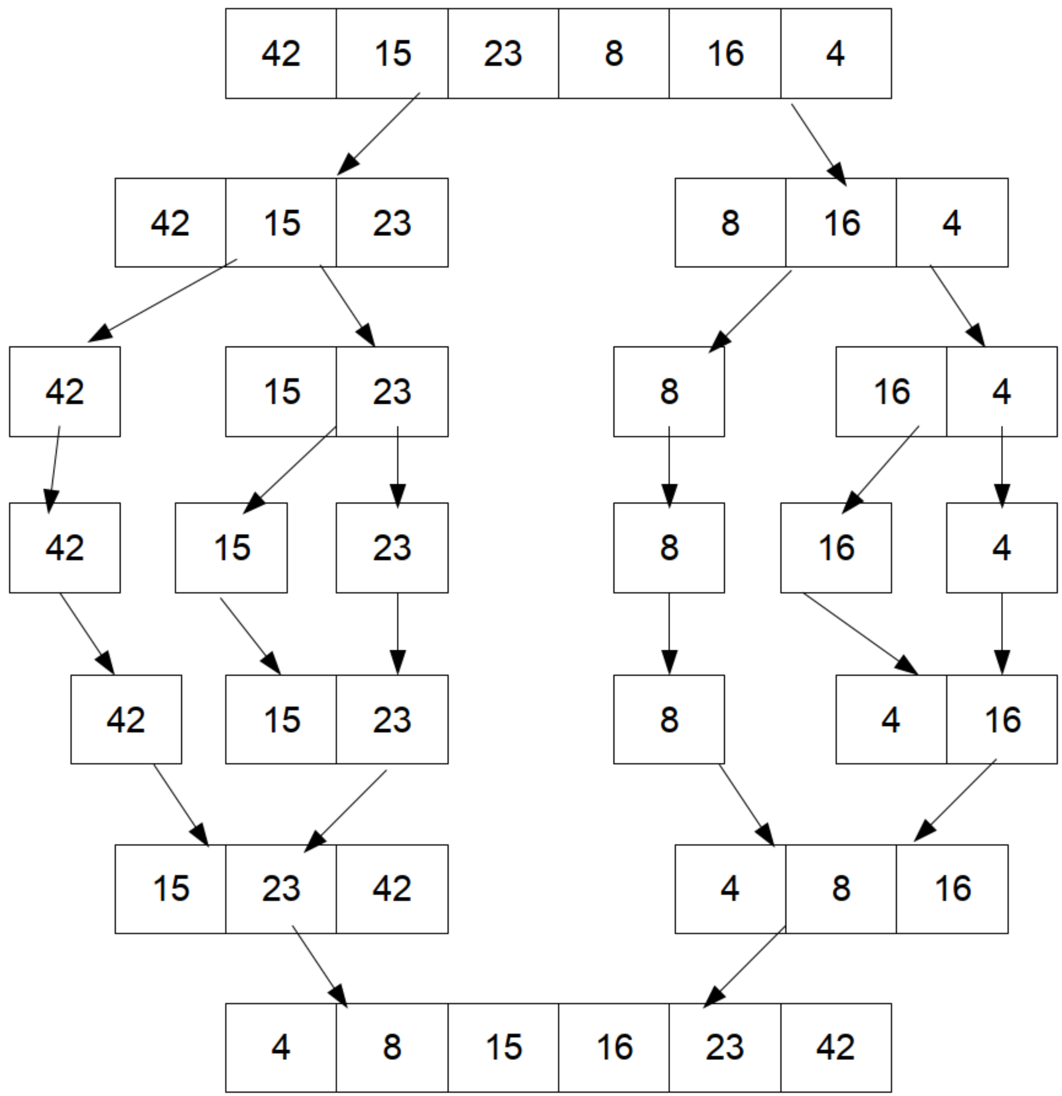
```
Intercala(v,a,m,b)
Se a ≥ b
    Devolva v
i ← a
j ← m + 1
Para k ← 0 to b − a
    Se j > b ou (i ≤ m e v[i] ≤ v[j])
        u[k] ← v[i]
        i ← i + 1
    Senão
        u[k] ← v[j]
        j ← j + 1
v[a..b] ← u[0..b − a]
Devolva v
```

MERGESORT

$Ordena_M(v,a,b)$

Se $a \geq b$
    Devolva v
$m \leftarrow \lfloor \frac{a+b}{2} \rfloor$
Ordena$_m(v, a, m)$
Ordena$_m(v, m+1, b)$
Devolva *Intercala*(v, a, m, b)

Executar $\mathsf{Ordena}_m(v, 1, 6)$



ANÁLISE

$C(n)$: núm. comparações do Mergesort (Ordena$_M$) em vetores de tamanho n

$$C(n) = \begin{cases} 0, & \text{se } n \leq 1 \\ C\left(\left\lceil \frac{n}{2} \right\rceil\right) + C\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + C_I(n), & \text{se } n > 1 \end{cases}$$

onde, $C_I(n) = n$

ANÁLISE

Teorema. *A relação de recorrência:*

$$C(n) = \begin{cases} 0, & \text{se } n \leq 1 \\ C\left(\left\lceil \frac{n}{2} \right\rceil\right) + C\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + n, & \text{se } n > 1 \end{cases}$$

tem como solução $C(n) \approx n \log_2 n$.

A solução da recorrência acima é complicada

será vista em Análise de Algoritmos

Mas, supondo simplificações, podemos resolvê-la.

ANÁLISE

$\boxed{\textbf{Simplificação: } \text{Supor que } n \text{ é potência de 2, ou seja, } n = 2^k}$

A simplificação serve para tirar pisos e tetos. Assim,

$$C(n) = \begin{cases} 0, & \text{se } n \leq 1 \\ 2C\left(\frac{n}{2}\right) + n, & \text{se } n > 1 \end{cases}$$

Resolver a recorrência com os alunos no quadro

$$C(n) = \begin{cases} 0, & \text{se } n \leq 1 \\ 2C\left(\frac{n}{2}\right) + n, & \text{se } n > 1 \end{cases}$$