



INSTITUTO FEDERAL  
Ceará

# Introdução à Análise de Algoritmos

Aula 24-05-2021

Divisão e Conquista

Prof: Glauber Cintra

[glauberfcintra@gmail.com](mailto:glauberfcintra@gmail.com)



# REGRAS GERAIS DE CONDUTA EM AULAS REMOTAS

A sala de aula virtual é uma extensão da sala de aula presencial e, portanto, o Regulamento da Organização Didática (ROD) é o documento que rege a sua dinâmica. Ao acessar a sala de aula virtual, você estará ciente de que a violação dessas regras é passível de medidas disciplinares, tanto no âmbito do IFCE como no âmbito civil e criminal. Para que possamos manter o ambiente harmônico, respeitoso e seguro entre todos, é necessário observar algumas regras de conduta, a saber:

## **Não compartilhe a gravação das aulas**

Você não deve copiar, distribuir, modificar, reproduzir, republicar, transmitir ou comercializar qualquer informação, texto e/ou documentos contidos nas aulas em qualquer meio eletrônico, nem criar qualquer trabalho utilizando imagens, textos ou documentos dessas aulas sem ter por escrito o prévio consentimento dos envolvidos na exposição.



## Tenha tolerância e paciência com possíveis falhas tecnológicas e eventuais limitações pessoais

Falhas técnicas poderão acontecer, seja com o professor, com colegas ou com você mesmo. Tenha paciência, procure manter a calma e contornar o problema com discrição e gentileza.

## Prepare-se para a aula virtual

Vista-se adequadamente e escolha na sua casa o local mais apropriado (se possível, separado de outras pessoas e das atividades que estiverem sendo realizadas por elas), para que haja o máximo de atenção na aula.

## Desative o microfone

Ao acionar seu aparelho, desative o microfone. Essa ação impedirá que, num momento de distração, você compartilhe uma fala ou ruídos indesejados. Seu celular deve ficar no silencioso. Evite também interromper a fala dos demais participantes e, pelo *chat*, peça a palavra ao professor quando quiser fazer algum comentário ou esclarecer alguma dúvida.



## Divisão e Conquista

A técnica de divisão e conquista consiste basicamente em dividir a instância a ser resolvida em instâncias menores **do mesmo tipo de problema**, resolvê-las e depois usar as soluções das instâncias menores para obter uma solução da instância original.

Para que um problema possa ser resolvido por divisão e conquista ele precisa ter duas propriedades:

- Decomponibilidade, ou seja, deve ser possível decompor qualquer instância não trivial do problema em instâncias menores do mesmo tipo de problema;
- A partir das soluções das instâncias obtidas com a decomposição deve ser sempre possível obter uma solução da instância original



INSTITUTO FEDERAL

Ceará

**Exemplo:**

## **Algoritmo Pot5**

***Entrada:***  $x \in \mathbb{N}$

***Saída:***  $2^x$

se  $x = 0$

    devolva 1

se não

    aux = Pot5( $\lfloor x/2 \rfloor$ )

    se  $x$  for par

        devolva aux \* aux

    se não

        devolva aux \* aux \* 2



## Algoritmo de Karatsuba-Ofman

Considere o problema de multiplicar dois números de  $n$  dígitos. O algoritmo clássico de multiplicação requer  $n^2$  multiplicações dígito a dígito.

Ex:

$$\begin{array}{r} 324 \\ 635 \\ \hline 1620 \\ 972 \\ 1944 \\ \hline 205740 \end{array}$$



O Algoritmo de Karatsuba-Ofman, baseado em divisão e conquista, permite multiplicar números com mais eficiência.

Sejam  $u$  e  $v$  números de  $n$  dígitos (vamos supor  $n$  par). Vamos decompor  $u$  nos números  $p$  e  $q$  de modo que  $p$  é constituído dos primeiros  $n/2$  dígitos de  $u$  e  $q$  é constituído dos últimos  $n/2$  dígitos de  $u$ . De forma análoga, vamos decompor  $v$  nos números  $r$  e  $s$  de modo que  $r$  é constituído dos primeiros  $n/2$  dígitos de  $v$  e  $s$  é constituído dos últimos  $n/2$  dígitos de  $v$ . Note que:

$$u = p \cdot 10^{n/2} + q$$

$$v = r \cdot 10^{n/2} + s$$

Assim:

$$u \cdot v = (p \cdot 10^{n/2} + q) \cdot (r \cdot 10^{n/2} + s)$$

$$u \cdot v = p \cdot r \cdot 10^n + p \cdot s \cdot 10^{n/2} + q \cdot r \cdot 10^{n/2} + q \cdot s$$

$$u \cdot v = p \cdot r \cdot 10^n + (p \cdot s + q \cdot r) \cdot 10^{n/2} + q \cdot s$$



$$u.v = p.r.10^n + (p.s + q.r).10^{n/2} + q.s$$

Usando essa fórmula, precisaremos de quatro multiplicações de números de  $n/2$  dígitos. O tempo requerido é dado por:

$$T(n) = 4T(n/2) + cn$$

$$T(1) = c$$

Vamos supor  $n = 2^k$

$$\text{Eq. 0} \quad T(n) = \cancel{4T(n/2)} + cn$$

$$\text{Eq. 1} \quad \cancel{4T(n/2)} = \cancel{16T(n/4)} + 2cn$$

$$\text{Eq. 2} \quad \cancel{16T(n/4)} = \cancel{64T(n/8)} + 4cn$$

...

$$\text{Eq. } k-1 \quad \cancel{4^{k-1}T(n/2^{k-1})} = \cancel{4^kT(n/2^k)} + 2^{k-1}cn$$

$$\text{Eq. } k \quad \cancel{4^kT(n/2^k)} = 4^kc$$

$$\begin{array}{l} \hline T(n) = cn + 2cn + \dots + 2^{k-1}cn + cn^2 = cn(1 + 2 + \dots + 2^{k-1}) + cn^2 = cn(2^k - 1) + cn^2 = 2cn^2 - cn \end{array}$$

Resolvendo essa fórmula, concluímos que o tempo requerido é  $\Theta(n^2)$  😞





No entanto, observe que:

$$(p + q).(r + s) = p.r + p.s + q.r + q.s$$

$$p.s + q.r = (p + q).(r + s) - p.r - q.s$$

Assim:

$$u.v = p.r.10^n + (p.s + q.r).10^{n/2} + q.s$$

$$u.v = p.r.10^n + ((p + q).(r + s) - p.r - q.s).10^{n/2} + q.s$$

Precisaremos então apenas 3 multiplicações de números de  $n/2$  dígitos 😊

Ex:  $u = 4815, v = 2952$

$$n = 4, p = 48, q = 15, r = 29, s = 52$$

$$4815 \times 2952 = 48 \times 29 \times 10^4 + ((48 + 15) \times (29 + 52) - 48 \times 29 - 15 \times 52) \times 10^2 + 15 \times 52$$

$$4815 \times 2952 = 13920000 + (63 \times 81 - 1392 - 780) \times 10^2 + 780$$

$$4815 \times 2952 = 13920000 + 293100 + 780$$

$$4815 \times 2952 = 14213880$$



A complexidade temporal do Algoritmo de Karatsuba-Ofman é dada por:

$$T(n) = 3T(n/2) + cn$$

$$T(1) = c$$

*Vamos supor  $n = 2^k \Rightarrow k = \log_2 n$*

$$\text{Eq. 0} \quad T(n) = \cancel{3T(n/2)} + cn$$

$$\text{Eq. 1} \quad \cancel{3T(n/2)} = \cancel{9T(n/4)} + \frac{3}{2}cn$$

$$\text{Eq. 2} \quad \cancel{9T(n/4)} = \cancel{27T(n/8)} + \frac{9}{4}cn$$

...

$$\text{Eq. } k-1 \quad \cancel{3^{k-1}T(n/2^{k-1})} = \cancel{3^kT(n/2^k)} + \left(\frac{3}{2}\right)^{k-1}cn$$

$$\text{Eq. } k \quad \cancel{3^kT(n/2^k)} = 3^kc$$

---

$$T(n) = cn + \frac{3}{2}cn + \dots + \left(\frac{3}{2}\right)^{k-1}cn + 3^kc = cn\left(1 + \frac{3}{2} + \dots + \left(\frac{3}{2}\right)^{k-1}\right) + 3^kc = 2\left(\left(\frac{3}{2}\right)^k - 1\right)cn + 3^kc$$

$$T(n) = 2cn\left(\frac{3}{2}\right)^k - 2cn + 3^kc = 2cn \frac{3^k}{2^k} - 2cn + 3^kc = 2c3^{\log_2 n} - 2cn + 3^{\log_2 n}c$$



$$T(n) = 2c3^{\log_2 n} - 2cn + 3^{\log_2 n}c$$

Lembre que:

$$a^{\log_c b} = b^{\log_c a}$$

Assim:

$$T(n) = 2cn^{\log_2 3} - 2cn + n^{\log_2 3}c = \quad \quad \quad \mathbf{3cn^{\log_2 3} - 2cn}$$

Concluimos que o tempo requerido é  $\Theta(n^{\log_2 3})$  😊



INSTITUTO FEDERAL  
Ceará

# Introdução à Análise de Algoritmos

Aula 28-05-2021

Mergesort e Quicksort

Prof: Glauber Cintra

[glauberfcintra@gmail.com](mailto:glauberfcintra@gmail.com)



# REGRAS GERAIS DE CONDUTA EM AULAS REMOTAS

A sala de aula virtual é uma extensão da sala de aula presencial e, portanto, o Regulamento da Organização Didática (ROD) é o documento que rege a sua dinâmica. Ao acessar a sala de aula virtual, você estará ciente de que a violação dessas regras é passível de medidas disciplinares, tanto no âmbito do IFCE como no âmbito civil e criminal. Para que possamos manter o ambiente harmônico, respeitoso e seguro entre todos, é necessário observar algumas regras de conduta, a saber:

## **Não compartilhe a gravação das aulas**

Você não deve copiar, distribuir, modificar, reproduzir, republicar, transmitir ou comercializar qualquer informação, texto e/ou documentos contidos nas aulas em qualquer meio eletrônico, nem criar qualquer trabalho utilizando imagens, textos ou documentos dessas aulas sem ter por escrito o prévio consentimento dos envolvidos na exposição.



## Tenha tolerância e paciência com possíveis falhas tecnológicas e eventuais limitações pessoais

Falhas técnicas poderão acontecer, seja com o professor, com colegas ou com você mesmo. Tenha paciência, procure manter a calma e contornar o problema com discrição e gentileza.

## Prepare-se para a aula virtual

Vista-se adequadamente e escolha na sua casa o local mais apropriado (se possível, separado de outras pessoas e das atividades que estiverem sendo realizadas por elas), para que haja o máximo de atenção na aula.

## Desative o microfone

Ao acionar seu aparelho, desative o microfone. Essa ação impedirá que, num momento de distração, você compartilhe uma fala ou ruídos indesejados. Seu celular deve ficar no silencioso. Evite também interromper a fala dos demais participantes e, pelo *chat*, peça a palavra ao professor quando quiser fazer algum comentário ou esclarecer alguma dúvida.



# Algoritmo Mergesort

O Mergesort é um conhecido algoritmo de ordenação baseado em divisão e conquista. Ele divide a lista a ser ordenada em duas metades, recursivamente ordena as duas metades e depois intercala as duas metades.

## Algoritmo Mergesort

*Entrada: Um vetor  $L$  e as posições início e fim*

*Saída: O vetor  $L$  em ordem crescente da posição início até a posição fim*

Se início < fim

    meio = (início + fim) / 2                      // divisão inteira

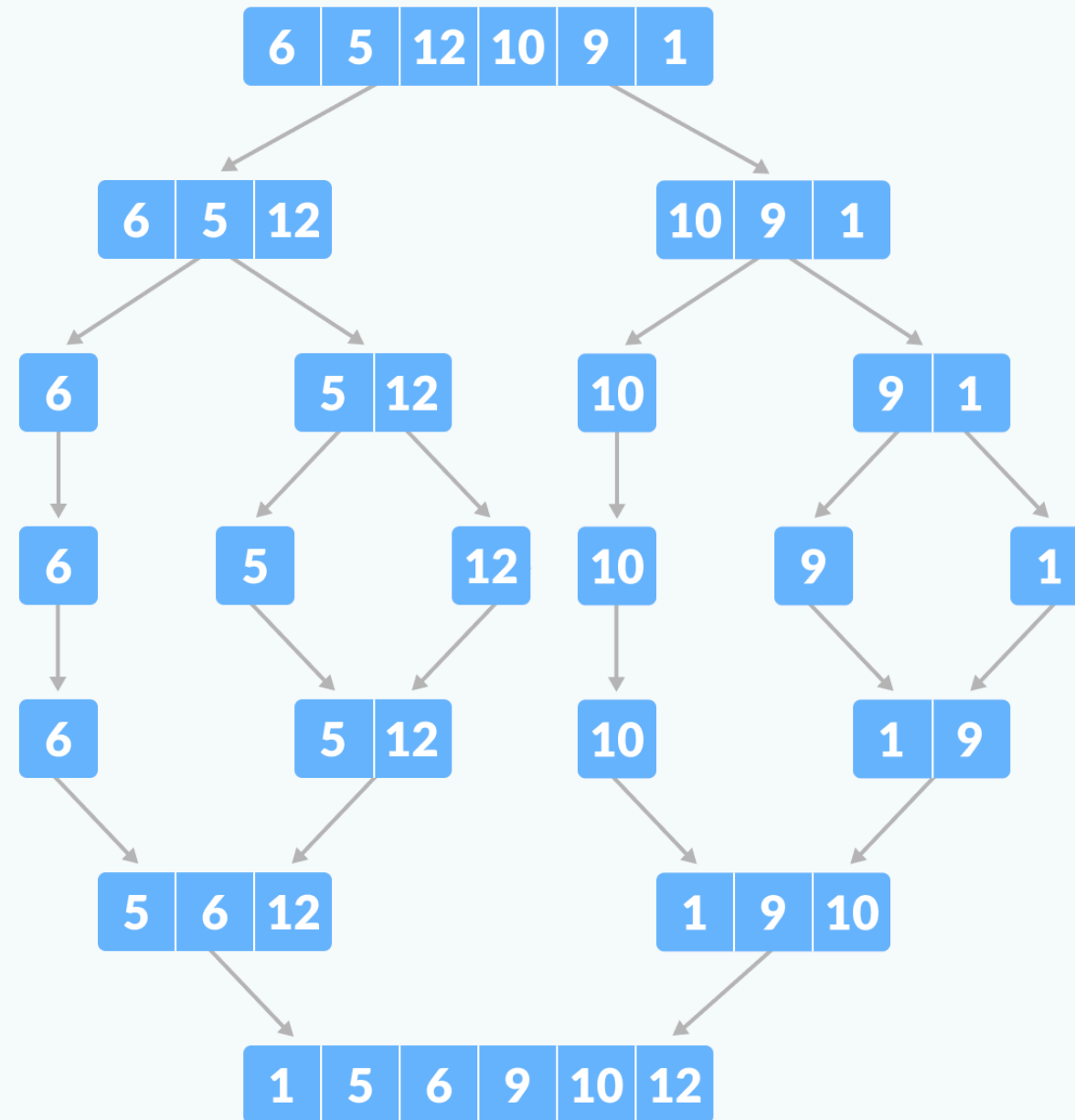
    se início < meio

        mergesort( $L$ , início, meio)

    se meio + 1 < fim

        mergesort( $L$ , meio + 1, fim)

    merge( $L$ , início, meio, fim)



Simulação do  
Mergesort





A complexidade temporal do Mergesort é dada por:

$$T(n) = 2T(n/2) + cn$$

$$T(1) = c$$

*Vamos supor  $n = 2^k \Rightarrow k = \log_2 n$*

$$\text{Eq. 0} \quad T(n) = 2T(n/2) + cn$$

$$\text{Eq. 1} \quad 2T(n/2) = 4T(n/4) + cn$$

$$\text{Eq. 2} \quad 4T(n/4) = 8T(n/8) + cn$$

...

$$\text{Eq. } k-1 \quad 2^{k-1}T(n/2^{k-1}) = 2^kT(n/2^k) + cn$$

$$\text{Eq. } k \quad 2^kT(n/2^k) = 2^kc$$

---

$$T(n) = cnk + 2^kc = cn\log_2 n + cn$$

Concluimos que o tempo requerido é  $\Theta(n \log_2 n)$ .



# Algoritmo Quicksort

O Quicksort é um algoritmo de ordenação também baseado em divisão e conquista. No quicksort é escolhido um *pivô* e a lista é dividida em duas sublistas, a da esquerda com os elementos menores ou iguais ao pivô e a da direita com os elementos maiores que o pivô. O pivô é colocado entre as duas sublistas. Após isso, cada sublista é ordenada recursivamente.

## Algoritmo Quicksort

*Entrada: Um vetor  $L$  e as posições início e fim*

*Saída: O vetor  $L$  em ordem crescente da posição início até a posição fim*

Se início < fim

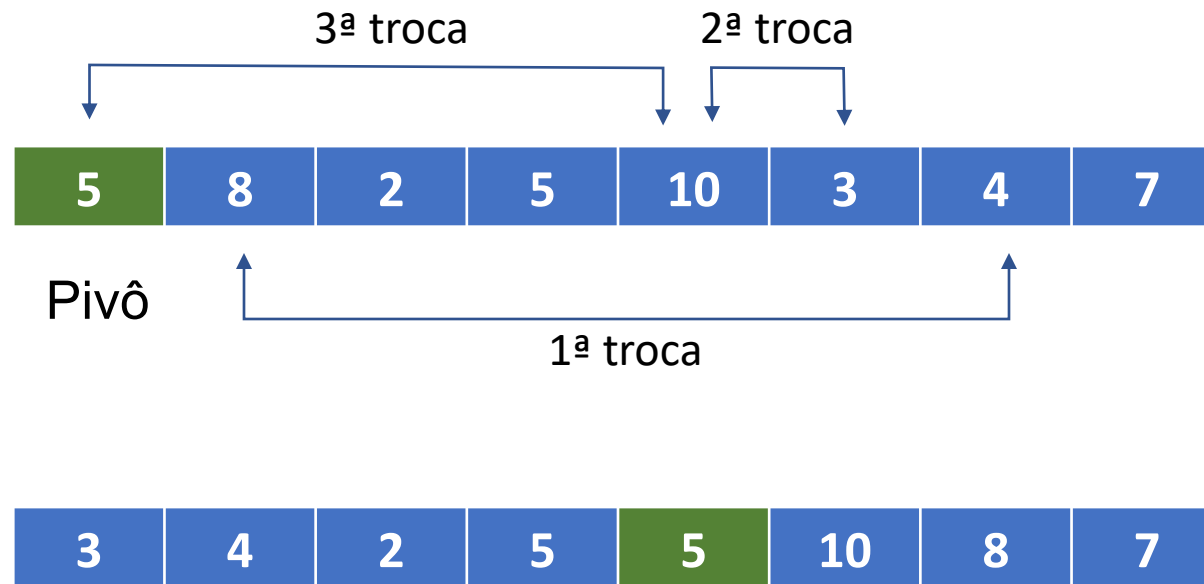
$j = \text{partição}(L, \text{início}, \text{fim})$

    se início <  $j - 1$

        quicksort( $L$ , início,  $j - 1$ )

    se  $j + 1 < \text{fim}$

        quicksort( $L$ ,  $j + 1$ , fim)



Simulação do  
procedimento de partição



O melhor caso do Quicksort ocorre quando todas as escolhas do pivô recaem sobre a mediana do intervalo a ser particionado. Nesse caso, a complexidade temporal do Quicksort é dada por:

$$T(n) = 2T(n/2) + cn$$

$$T(1) = c$$

Concluimos que o tempo requerido no melhor caso é  $\Theta(n \log_2 n)$ .

O pior caso do Quicksort ocorre quando todas as escolhas do pivô recaem sobre um elemento extremo do intervalo a ser particionado. Nesse caso, a complexidade temporal do Quicksort é dada por:

$$T(n) = T(n - 1) + cn$$

$$T(1) = c$$

Concluimos que o tempo requerido no pior caso é  $\Theta(n^2)$ .



INSTITUTO FEDERAL  
Ceará

# Introdução à Análise de Algoritmos

Aula 31-05-2021

Programação Dinâmica

Prof: Glauber Cintra

[glauberfcintra@gmail.com](mailto:glauberfcintra@gmail.com)



# REGRAS GERAIS DE CONDUTA EM AULAS REMOTAS

A sala de aula virtual é uma extensão da sala de aula presencial e, portanto, o Regulamento da Organização Didática (ROD) é o documento que rege a sua dinâmica. Ao acessar a sala de aula virtual, você estará ciente de que a violação dessas regras é passível de medidas disciplinares, tanto no âmbito do IFCE como no âmbito civil e criminal. Para que possamos manter o ambiente harmônico, respeitoso e seguro entre todos, é necessário observar algumas regras de conduta, a saber:

## **Não compartilhe a gravação das aulas**

Você não deve copiar, distribuir, modificar, reproduzir, republicar, transmitir ou comercializar qualquer informação, texto e/ou documentos contidos nas aulas em qualquer meio eletrônico, nem criar qualquer trabalho utilizando imagens, textos ou documentos dessas aulas sem ter por escrito o prévio consentimento dos envolvidos na exposição.



## Tenha tolerância e paciência com possíveis falhas tecnológicas e eventuais limitações pessoais

Falhas técnicas poderão acontecer, seja com o professor, com colegas ou com você mesmo. Tenha paciência, procure manter a calma e contornar o problema com discrição e gentileza.

## Prepare-se para a aula virtual

Vista-se adequadamente e escolha na sua casa o local mais apropriado (se possível, separado de outras pessoas e das atividades que estiverem sendo realizadas por elas), para que haja o máximo de atenção na aula.

## Desative o microfone

Ao acionar seu aparelho, desative o microfone. Essa ação impedirá que, num momento de distração, você compartilhe uma fala ou ruídos indesejados. Seu celular deve ficar no silencioso. Evite também interromper a fala dos demais participantes e, pelo *chat*, peça a palavra ao professor quando quiser fazer algum comentário ou esclarecer alguma dúvida.



# Programação Dinâmica

A *programação dinâmica* (PD) pode ser vista como uma generalização da técnica de divisão e conquista. A ideia básica da programação dinâmica é decompor a instância a ser resolvida de diversas maneiras, obtendo em instâncias menores **do mesmo tipo de problema**, resolvê-las e depois usar as soluções das instâncias menores para obter uma solução da instância original.

Para evitar que a mesma instância seja resolvida várias vezes, na PD resolvemos as instâncias começando pelas menores até as maiores, armazenando o resultado numa estrutura de dados.

Para que um problema possa ser resolvido por programação dinâmica ele precisa ter duas propriedades:

- *Decomponibilidade*, ou seja, deve ser possível decompor qualquer instância não trivial do problema em instâncias menores do mesmo tipo de problema;
- *Subestrutura ótima*, ou seja, deve existir alguma decomposição tal que a partir das soluções das instâncias obtidas com a decomposição deve ser sempre possível obter uma solução da instância original





## Subsequência Crescente Máxima

Seja  $S = (s_1, s_2, \dots, s_n)$  uma sequência de números. Dizemos que  $S'$  é uma *subsequência* de  $S$  se é possível obter  $S'$  a partir de  $S$  removendo alguns dos elementos de  $S$ .

Ex:  $S = (8, 4, 9, 3, 4, 7, 5, 10, 3)$

$(4, 7, 1, 10)$  Não é subsequência de  $S$

$(4, 4, 10, 5)$  Não é subsequência de  $S$

$(8, 3, 7, 10, 3)$  Subsequência de  $S$

Uma subsequência é crescente se os seus elementos estão em ordem crescente.

Ex:  $(8, 9, 10)$  Subsequência crescente de  $S$

No *Problema da Subsequência Crescente Máxima* (SCM) é dada uma sequência  $S$  e desejamos encontrar o comprimento de uma subsequência crescente de  $S$  que tenha a maior quantidade possível de elementos.

Ex:  $(4, 4, 7, 10)$  Subsequência crescente máxima de  $S$ ?



O SCM não possui subestrutura ótima, no entanto um problema correlato tem: o *problema da subsequência crescente máxima com término fixo* (SCMTF).

No SCMTF é dada uma sequência de números  $S = (s_1, s_2, \dots, s_n)$  e um elemento da sequência  $s_i$  ( $1 \leq i \leq n$ ). Desejamos encontrar a maior subsequência crescente de  $S$  que termine em  $s_i$ .

Para resolver o SCMTF vamos denotar por  $c_i$  o comprimento da maior subsequência crescente de  $S$  que termina em  $s_i$ . Note que:

$$c_1 = 1$$

$$c_i = \max(\{c_j + 1 \mid s_j \leq s_i, j = 1, 2, \dots, i - 1\}, 1)$$

Se calcularmos  $c_1, c_2, \dots, c_n$ , o maior desses valores será a solução do SCM.



## Algoritmo SCM\_PD

**Entrada:** uma sequência de números  $S = (s_1, s_2, \dots, s_n)$

**Saída:** o comprimento de uma SCM de  $S$

para  $i = 1$  até  $n$

$c_i = 1$

para  $j = 1$  até  $i - 1$

se  $s_j \leq s_i$  e  $c_j + 1 > c_i$

$c_i = c_j + 1$

devolva o maior valor entre  $c_1, c_2, \dots, c_n$

O Algoritmo SCM\_PD requer tempo  $\Theta(n^2)$  e espaço  $\Theta(n)$ .

Ex:  $S = (8, 4, 9, 3, 4, 7, 5, 10, 3)$

$i$	1	2	3	4	5	6	7	8	9
$s_i$	8	4	9	3	4	7	5	10	3
$c_i$	1	1	2	1	2	3	3	4	2



INSTITUTO FEDERAL  
Ceará

## Diversão para casa

Adaptar o Algoritmo SCM\_PD para que ele devolva uma SCM.



INSTITUTO FEDERAL  
Ceará

# Introdução à Análise de Algoritmos

Aula 07-06-2021

Problema da Mochila

Prof: Glauber Cintra

[glauberfcintra@gmail.com](mailto:glauberfcintra@gmail.com)



# REGRAS GERAIS DE CONDUTA EM AULAS REMOTAS

A sala de aula virtual é uma extensão da sala de aula presencial e, portanto, o Regulamento da Organização Didática (ROD) é o documento que rege a sua dinâmica. Ao acessar a sala de aula virtual, você estará ciente de que a violação dessas regras é passível de medidas disciplinares, tanto no âmbito do IFCE como no âmbito civil e criminal. Para que possamos manter o ambiente harmônico, respeitoso e seguro entre todos, é necessário observar algumas regras de conduta, a saber:

## **Não compartilhe a gravação das aulas**

Você não deve copiar, distribuir, modificar, reproduzir, republicar, transmitir ou comercializar qualquer informação, texto e/ou documentos contidos nas aulas em qualquer meio eletrônico, nem criar qualquer trabalho utilizando imagens, textos ou documentos dessas aulas sem ter por escrito o prévio consentimento dos envolvidos na exposição.



## Tenha tolerância e paciência com possíveis falhas tecnológicas e eventuais limitações pessoais

Falhas técnicas poderão acontecer, seja com o professor, com colegas ou com você mesmo. Tenha paciência, procure manter a calma e contornar o problema com discrição e gentileza.

## Prepare-se para a aula virtual

Vista-se adequadamente e escolha na sua casa o local mais apropriado (se possível, separado de outras pessoas e das atividades que estiverem sendo realizadas por elas), para que haja o máximo de atenção na aula.

## Desative o microfone

Ao acionar seu aparelho, desative o microfone. Essa ação impedirá que, num momento de distração, você compartilhe uma fala ou ruídos indesejados. Seu celular deve ficar no silencioso. Evite também interromper a fala dos demais participantes e, pelo *chat*, peça a palavra ao professor quando quiser fazer algum comentário ou esclarecer alguma dúvida.



# Problema da Mochila

A *Problema da Mochila* é um problema que aparece em diversas situações práticas e possui diversas variantes.

Uma dessas variantes é chamada de *Problema da Mochila Binária* (PMB). No PMB temos uma mochila de capacidade  $C$  e  $n$  itens. Cada item  $i$  possui peso  $p_i$  e valor  $v_i$ .

Precisamos decidir quais itens devem ser colocados na mochila de modo que a soma dos pesos dos itens colocados na mochila não exceda sua capacidade e que a soma dos valores desses itens seja a maior possível.

Se a capacidade da mochila e os pesos dos itens forem todos inteiros, podemos resolver o PMB usando programação dinâmica.

Para isso, vamos denotar por  $v(i, j)$  o maior valor que pode ser obtido numa mochila de capacidade  $i$  se podemos colocar nela os itens de 1 a  $j$ . Note que:

$$v(0, j) = 0$$

$$v(i, 0) = 0$$

$$v(i, j) = \max(v_j + v(i - p_j, j - 1), v(i, j - 1))$$

Podemos usar PD para calcular  $v(C, n)$





## Algoritmo PMB\_PD

**Entrada:**  $C, p_1, p_2, \dots, p_n, v_1, v_2, \dots, v_n$

**Saída:**  $v(C, n)$

para  $i = 0$  até  $C$

$v(i, 0) = 0$

para  $j = 1$  até  $n$

para  $i = 0$  até  $C$

se  $p_j \leq i$  e  $v_j + v(i - p_j, j - 1) > v(i, j - 1)$

$v(i, j) = v_j + v(i - p_j, j - 1)$

se não

$v(i, j) = v(i, j - 1)$

devolva  $v(C, n)$

O Algoritmo PMB\_PD requer tempo e espaço  $\Theta(nC)$ .

Ele é **pseudo-polinomial**.



Ex:  $C = 10$ ,  $n = 4$

$i$	1	2	3	4
$p_i$	5	3	7	4
$v_i$	52	28	66	43

$V$	0	1	2	3	4
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	28	28	28
4	0	0	28	28	43
5	0	52	52	52	52
6	0	52	52	52	52
7	0	52	52	66	71
8	0	52	80	80	80
9	0	52	80	80	95
10	0	52	80	94	95



## Diversão para casa

Adaptar o Algoritmo PBM\_PD para que ele indique quais itens devem ser colocados na mochila.



INSTITUTO FEDERAL  
Ceará

# Introdução à Análise de Algoritmos

Aula 11-06-2021

Enumeração Explícita

Prof: Glauber Cintra

[glauberfcintra@gmail.com](mailto:glauberfcintra@gmail.com)



# REGRAS GERAIS DE CONDUTA EM AULAS REMOTAS

A sala de aula virtual é uma extensão da sala de aula presencial e, portanto, o Regulamento da Organização Didática (ROD) é o documento que rege a sua dinâmica. Ao acessar a sala de aula virtual, você estará ciente de que a violação dessas regras é passível de medidas disciplinares, tanto no âmbito do IFCE como no âmbito civil e criminal. Para que possamos manter o ambiente harmônico, respeitoso e seguro entre todos, é necessário observar algumas regras de conduta, a saber:

## **Não compartilhe a gravação das aulas**

Você não deve copiar, distribuir, modificar, reproduzir, republicar, transmitir ou comercializar qualquer informação, texto e/ou documentos contidos nas aulas em qualquer meio eletrônico, nem criar qualquer trabalho utilizando imagens, textos ou documentos dessas aulas sem ter por escrito o prévio consentimento dos envolvidos na exposição.



## Tenha tolerância e paciência com possíveis falhas tecnológicas e eventuais limitações pessoais

Falhas técnicas poderão acontecer, seja com o professor, com colegas ou com você mesmo. Tenha paciência, procure manter a calma e contornar o problema com discrição e gentileza.

## Prepare-se para a aula virtual

Vista-se adequadamente e escolha na sua casa o local mais apropriado (se possível, separado de outras pessoas e das atividades que estiverem sendo realizadas por elas), para que haja o máximo de atenção na aula.

## Desative o microfone

Ao acionar seu aparelho, desative o microfone. Essa ação impedirá que, num momento de distração, você compartilhe uma fala ou ruídos indesejados. Seu celular deve ficar no silencioso. Evite também interromper a fala dos demais participantes e, pelo *chat*, peça a palavra ao professor quando quiser fazer algum comentário ou esclarecer alguma dúvida.



## Enumeração Explícita

A técnica de *enumeração explícita* (EE), às vezes chamada de *força bruta*, consiste basicamente em enumerar todos os candidatos a solução e dentre eles encontrar uma solução para a instância a ser resolvida.

Um problema cuja solução pode ser adaptada para resolver vários outros problemas por EE é, dado um conjunto  $C = \{e_1, e_2, \dots, e_n\}$ , enumerar todos os subconjuntos de  $C$ .

Observe que cada um dos subconjuntos de  $C$  pode ser representado por um conjunto de  $n$  bits no qual um bit 1 significa que o elemento correspondente a esse bit pertence ao subconjunto e 0 significa que não pertence.

Por exemplo, seja  $C = \{4, 8, 5\}$ . A sequência de bits 101 representa o subconjunto  $\{4, 5\}$ .

Note que o próprio  $C$  é representado pela sequência de  $n$  bits todos iguais a 1, que equivale ao número decimal  $2^n - 1$ . Já o conjunto vazio é representado pela sequência de  $n$  bits todos iguais a 0. Assim, as representações binárias dos números de 0 a  $2^n - 1$  codificam todos os subconjuntos de  $C$ .



## Algoritmo imprime\_subconjuntos

**Entrada:**  $C = \{e_1, e_2, \dots, e_n\}$

**Saída:** imprime todos os subconjuntos de  $C$

para  $i = 0$  até  $2^n - 1$

    imprime\_subconjunto( $C, i$ )

## Procedimento imprime\_subconjunto

**Entrada:**  $C = \{e_1, e_2, \dots, e_n\}$  e um valor  $i$

**Saída:** imprime o subconjunto de  $C$  codificado pelos bits de  $i$

para  $j = 1$  até  $n$

    Se  $i \% 2 = 1$

        imprima  $e_j$

$i = i / 2$  // divisão inteira

O procedimento imprime\_subconjunto requer tempo  $\Theta(n)$ .

O algoritmo imprime\_subconjuntos requer tempo  $\Theta(n2^n)$  😞

Apesar de ineficiente, este algoritmo é de cota inferior 😊





## Procedimento imprime\_subconjunto

**Entrada:**  $C = \{e_1, e_2, \dots, e_n\}$  e um valor  $i$

**Saída:** imprime o subconjunto de  $C$  codificado pelos bits de  $i$   
para  $j = 1$  até  $n$

Se  $i \% 2 = 1$

imprima  $e_j$

$i = i / 2$  // divisão inteira

Ex:  $C = \{4, 8, 5\}$ ,  $i = 5$

$j = 1, i = 5, i \% 2 = 1 \rightarrow$  imprime  $e_1 = 4$

$j = 2, i = 2, i \% 2 = 0 \rightarrow$  não imprime  $e_2 = 8$

$j = 3, i = 1, i \% 2 = 1 \rightarrow$  imprime  $e_3 = 5$

Subconjunto impresso:  $\{4, 5\}$



## Algoritmo imprime\_subconjuntos

**Entrada:**  $C = \{e_1, e_2, \dots, e_n\}$

**Saída:** imprime todos os subconjuntos de  $C$

para  $i = 0$  até  $2^n - 1$

    imprime\_subconjunto( $C, i$ )

Ex:  $C = \{4, 8, 5\}$ ,  $n = 3$

$i$	binário	subconjunto
0	000	$\{\}$
1	001	$\{4\}$
2	010	$\{8\}$
3	011	$\{4, 8\}$
4	100	$\{5\}$
5	101	$\{4, 5\}$
6	110	$\{8, 5\}$
7	111	$\{4, 8, 5\}$



INSTITUTO FEDERAL  
Ceará

## Diversão para casa

Adapte o Algoritmo `imprime_subconjuntos` para resolver o problema da subsequência crescente máxima.



INSTITUTO FEDERAL  
Ceará

# Introdução à Análise de Algoritmos

Aula 14-06-2021

Exemplos de Algoritmos de Enumeração Explícita

Prof: Glauber Cintra

[glauberfcintra@gmail.com](mailto:glauberfcintra@gmail.com)



# REGRAS GERAIS DE CONDUTA EM AULAS REMOTAS

A sala de aula virtual é uma extensão da sala de aula presencial e, portanto, o Regulamento da Organização Didática (ROD) é o documento que rege a sua dinâmica. Ao acessar a sala de aula virtual, você estará ciente de que a violação dessas regras é passível de medidas disciplinares, tanto no âmbito do IFCE como no âmbito civil e criminal. Para que possamos manter o ambiente harmônico, respeitoso e seguro entre todos, é necessário observar algumas regras de conduta, a saber:

## **Não compartilhe a gravação das aulas**

Você não deve copiar, distribuir, modificar, reproduzir, republicar, transmitir ou comercializar qualquer informação, texto e/ou documentos contidos nas aulas em qualquer meio eletrônico, nem criar qualquer trabalho utilizando imagens, textos ou documentos dessas aulas sem ter por escrito o prévio consentimento dos envolvidos na exposição.



## Tenha tolerância e paciência com possíveis falhas tecnológicas e eventuais limitações pessoais

Falhas técnicas poderão acontecer, seja com o professor, com colegas ou com você mesmo. Tenha paciência, procure manter a calma e contornar o problema com discrição e gentileza.

## Prepare-se para a aula virtual

Vista-se adequadamente e escolha na sua casa o local mais apropriado (se possível, separado de outras pessoas e das atividades que estiverem sendo realizadas por elas), para que haja o máximo de atenção na aula.

## Desative o microfone

Ao acionar seu aparelho, desative o microfone. Essa ação impedirá que, num momento de distração, você compartilhe uma fala ou ruídos indesejados. Seu celular deve ficar no silencioso. Evite também interromper a fala dos demais participantes e, pelo *chat*, peça a palavra ao professor quando quiser fazer algum comentário ou esclarecer alguma dúvida.



## Algoritmo SCM\_EE

**Entrada:**  $S = (s_1, s_2, \dots, s_n)$

**Saída:** uma SCM de  $S$  e o seu tamanho

maximo = -1

para  $i = 0$  até  $2^n - 1$

tamanho = tamanho\_subsequencia( $S, i$ )

se tamanho > maximo

maximo = tamanho, scm =  $i$

devolva scm e maximo // o bits de scm codificam uma SCM de  $S$

## Procedimento tamanho\_subsequencia

**Entrada:**  $S = (s_1, s_2, \dots, s_n)$  e um valor  $i$

**Saída:** se a subsequência de  $S$  codificada pelos bits de  $i$  for crescente devolve o tamanho dessa subsequência; caso contrário, devolve -1

tamanho = 0

para  $j = 1$  até  $n$

se  $i \% 2 = 1$

se tamanho > 0 e  $s_j < \text{anterior}$

devolva -1 e pare

tamanho++, anterior =  $s_j$

$i = i / 2$  // divisão inteira

devolva tamanho

O procedimento tamanho\_subsequencia requer tempo  $\Theta(n)$ .

O algoritmo SCM\_EE requer tempo  $\Theta(n2^n)$  😞



## Algoritmo PMB\_EE

**Entrada:**  $C, p_1, p_2, \dots, p_n, v_1, v_2, \dots, v_n$

**Saída:** uma solução ótima para a instância e o valor dessa solução

maximo = 0, sol = 0

para  $i = 1$  até  $2^n - 1$

    valor = valor\_mochila( $C, p_1, p_2, \dots, p_n, v_1, v_2, \dots, v_n, i$ )

    se valor > maximo

        maximo = valor, sol = i

devolva sol e maximo      // o bits de sol codificam uma solução da instância

## Procedimento valor\_mochila

**Entrada:**  $C, p_1, p_2, \dots, p_n, v_1, v_2, \dots, v_n$  e um valor  $i$

**Saída:** se a forma de preencher a mochila codificada pelos bits de  $i$  não exceder a capacidade da mochila devolve o valor obtido com essa solução; caso contrário, devolve -1

valor = 0, peso = 0

para  $j = 1$  até  $n$

    se  $i \% 2 = 1$

        se  $p_j + \text{peso} > C$

            devolva -1 e pare

        peso = peso +  $p_j$ , valor = valor +  $v_j$

$i = i / 2$       // divisão inteira

devolva valor

O procedimento valor\_mochila requer tempo  $O(n)$ .

O algoritmo PMB\_EE requer tempo  $\Theta(n2^n)$  😞





## Diversões para casa

- 1) Implemente e teste os algoritmos abordados na aula de hoje.
- 2) Escreva um algoritmo baseado em enumeração explícita que receba uma coleção de números e devolva uma partição dessa coleção em duas subcoleções tais que a soma dos números em cada subcoleção seja a mesma (*problema da 2-partição*). Informe a complexidade temporal do seu algoritmo.

Ex:  $C = (9, 3, 2, 5, 1, 2, 3, 7)$

Uma solução:  $(9, 3, 1, 3)$  e  $(2, 5, 2, 7)$

Outra solução:  $(9, 7)$  e  $(3, 2, 5, 1, 2, 3)$



INSTITUTO FEDERAL  
Ceará

# Introdução à Análise de Algoritmos

Aula 18-06-2021

Enumeração Implícita para o Problema da Mochila

Prof: Glauber Cintra

[glauberfcintra@gmail.com](mailto:glauberfcintra@gmail.com)



# REGRAS GERAIS DE CONDUTA EM AULAS REMOTAS

A sala de aula virtual é uma extensão da sala de aula presencial e, portanto, o Regulamento da Organização Didática (ROD) é o documento que rege a sua dinâmica. Ao acessar a sala de aula virtual, você estará ciente de que a violação dessas regras é passível de medidas disciplinares, tanto no âmbito do IFCE como no âmbito civil e criminal. Para que possamos manter o ambiente harmônico, respeitoso e seguro entre todos, é necessário observar algumas regras de conduta, a saber:

## **Não compartilhe a gravação das aulas**

Você não deve copiar, distribuir, modificar, reproduzir, republicar, transmitir ou comercializar qualquer informação, texto e/ou documentos contidos nas aulas em qualquer meio eletrônico, nem criar qualquer trabalho utilizando imagens, textos ou documentos dessas aulas sem ter por escrito o prévio consentimento dos envolvidos na exposição.



## Tenha tolerância e paciência com possíveis falhas tecnológicas e eventuais limitações pessoais

Falhas técnicas poderão acontecer, seja com o professor, com colegas ou com você mesmo. Tenha paciência, procure manter a calma e contornar o problema com discrição e gentileza.

## Prepare-se para a aula virtual

Vista-se adequadamente e escolha na sua casa o local mais apropriado (se possível, separado de outras pessoas e das atividades que estiverem sendo realizadas por elas), para que haja o máximo de atenção na aula.

## Desative o microfone

Ao acionar seu aparelho, desative o microfone. Essa ação impedirá que, num momento de distração, você compartilhe uma fala ou ruídos indesejados. Seu celular deve ficar no silencioso. Evite também interromper a fala dos demais participantes e, pelo *chat*, peça a palavra ao professor quando quiser fazer algum comentário ou esclarecer alguma dúvida.



## Enumeração Implícita

Na *enumeração implícita* em vez de enumerarmos todos os candidatos a solução, enumeramos apenas os candidatos *promissores*, aqueles que têm chance de ser solução. Se o conjunto de candidatos promissores for bem menor que o conjunto de todos os candidatos, a enumeração implícita será muito mais eficiente do que a enumeração explícita.

Existem diversas técnicas de enumeração implícita: *branch-and-bound*, *branch-and-cut*, *branch-and-price*, *branch-and-cut-and-price* etc. Tais técnicas costumam ser bem sofisticadas. Veremos um exemplo de algoritmo de branch-and-bound (BB) para o PMB.

Seja  $X$  uma solução viável (que não excede a capacidade da mochila) para uma instância do PMB. Se a capacidade remanescente na mochila ao aplicar a solução  $X$  for maior do que o peso de algum dos itens que não foram colocados na mochila, dizemos que  $X$  é uma solução *insensata*, pois não pode ser uma solução ótima.

O algoritmo de BB que iremos apresentar enumera apenas soluções viáveis procurando evitar soluções insensatas. Além disso, se já enumeramos uma solução de valor  $V$ , devemos evitar enumerar soluções que tenham valor menor ou igual a  $V$ .

[illegible]

Essa árvore enumera todas as possíveis formas de preencher uma mochila para a instância descrita acima. Na árvore, após o nível da raiz, temos um nível para cada item. A ramificação à esquerda indica que o item não deve ser colocado na mochila e a ramificação à direita indica que o item deve ser colocado. Cada ramo da árvore, desde a raiz até uma folha, representa uma forma de preencher a mochila. O ramo destacado de azul representa a solução em que apenas itens 2 e 3 são colocados na mochila. Note que essa solução é ótima e tem valor 23. Observe ainda que os nós amarelos correspondem a soluções *insensatas* e os nós vermelhos correspondem a soluções *inviáveis*.



## Branch-and-bound para o PMB

Considere uma instância  $I = (C, p_1, p_2, \dots, p_n, v_1, v_2, \dots, v_n)$  do PMB sendo que os itens estão em ordem decrescente de *valor relativo* (valor/peso).

Seja  $X = (x_1, x_2, \dots, x_n)$  uma solução viável codificada por um vetor de bits onde  $x_i = 1$  indica que o item  $i$  deve ser colocado na mochila e  $x_i = 0$  indica que o item  $i$  não deve ser colocado na mochila.

Note que o valor dessa solução  $X$  é  $\sum_{j=1}^n x_j v_j$ . Além disso, a capacidade remanescente na mochila é  $C - \sum_{j=1}^n x_j p_j$ .

Suponha que uma solução  $X' = (x'_1, x'_2, \dots, x'_n)$  tenha  $x'_i = x_i$  ( $i = 1, 2, \dots, k-1$ ) e  $x'_k = 0$ . Observe que o valor dessa solução  $X'$  será no máximo:

$$M = \sum_{j=1}^{k-1} x_j v_j + (C - \sum_{j=1}^{k-1} x_j p_j) \frac{v_{k+1}}{p_{k+1}}$$

Se  $M$  for menor ou igual ao valor da melhor solução já encontrada então  $X'$  não precisa ser enumerada. Isso significa que podemos deixar de enumerar **todas** as soluções que começam com  $x_1, x_2, \dots, x_{k-1}$  e tenham  $x_k = 0$ . Temos assim uma *condição de poda*, ou seja, uma condição que pode ser usada para limitar a enumeração. 😊



## Algoritmo PMB\_BB

**Entrada:**  $C, p_1, p_2, \dots, p_n, v_1, v_2, \dots, v_n$

**Saída:** uma solução ótima  $x^*$

$M = 0, k = 0$

coloque os itens em ordem decrescente de valor relativo

faça

para  $i = k + 1$  até  $n$

se  $p_i \leq C - \sum_{j=1}^{i-1} x_j p_j$

$x_i = 1$

se não

$x_i = 0$

se  $\sum_{j=1}^n x_j v_j > M$

$M = \sum_{j=1}^n x_j v_j, x^* = x$

$k = \max(\{i \mid i < n, x_i = 1 \text{ e } \sum_{j=1}^{i-1} x_j v_j + (C - \sum_{j=1}^{i-1} x_j p_j) \frac{v_{i+1}}{p_{i+1}} > M\}, \{0\})$

se  $k > 0$

$x_k = 0$

enquanto  $k > 0$

devolva  $x^*$





Ex:  $C = 20$ ,  $n = 6$

$i$	1	2	3	4	5	6
$p_i$	5	3	7	6	10	8
$v_i$	12	7	16	13	20	14

$M = 0, 35, 39, 41, 42, 43$

$k = 0, 3, 4, 5, 2, 4, 3, 1, 4, 2, 0$

A melhor solução tem valor 43 e é obtida inserindo os itens 2, 3 e 5 na mochila.

**Obs:** Os números em vermelho indicam os casos em que a condição de poda foi usada para restringir a enumeração

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	valor	peso
1	1	1	0	0	0	35	15
1	1	0	1	0	0	32	14
1	1	0	0	1	0	39	18
1	1	0	0	0	1	33	16
1	0	1	1	0	0	41	18
1	0	1	0	0	1	42	20
1	0	0	1	0	1	39	19
0	1	1	1	0	0	36	16
0	1	1	0	1	0	43 x*	20
0	0	1	1	0	0	29	13



INSTITUTO FEDERAL  
Ceará

# Introdução à Análise de Algoritmos

Aula 02-07-2021

Estratégia Gulosa – Parte 1

Prof: Glauber Cintra

[glauberfcintra@gmail.com](mailto:glauberfcintra@gmail.com)



# REGRAS GERAIS DE CONDUTA EM AULAS REMOTAS

A sala de aula virtual é uma extensão da sala de aula presencial e, portanto, o Regulamento da Organização Didática (ROD) é o documento que rege a sua dinâmica. Ao acessar a sala de aula virtual, você estará ciente de que a violação dessas regras é passível de medidas disciplinares, tanto no âmbito do IFCE como no âmbito civil e criminal. Para que possamos manter o ambiente harmônico, respeitoso e seguro entre todos, é necessário observar algumas regras de conduta, a saber:

## **Não compartilhe a gravação das aulas**

Você não deve copiar, distribuir, modificar, reproduzir, republicar, transmitir ou comercializar qualquer informação, texto e/ou documentos contidos nas aulas em qualquer meio eletrônico, nem criar qualquer trabalho utilizando imagens, textos ou documentos dessas aulas sem ter por escrito o prévio consentimento dos envolvidos na exposição.



## Tenha tolerância e paciência com possíveis falhas tecnológicas e eventuais limitações pessoais

Falhas técnicas poderão acontecer, seja com o professor, com colegas ou com você mesmo. Tenha paciência, procure manter a calma e contornar o problema com discrição e gentileza.

## Prepare-se para a aula virtual

Vista-se adequadamente e escolha na sua casa o local mais apropriado (se possível, separado de outras pessoas e das atividades que estiverem sendo realizadas por elas), para que haja o máximo de atenção na aula.

## Desative o microfone

Ao acionar seu aparelho, desative o microfone. Essa ação impedirá que, num momento de distração, você compartilhe uma fala ou ruídos indesejados. Seu celular deve ficar no silencioso. Evite também interromper a fala dos demais participantes e, pelo *chat*, peça a palavra ao professor quando quiser fazer algum comentário ou esclarecer alguma dúvida.



## Estratégia Gulosa

Alguns problemas podem ser resolvidos usando a técnica conhecida como *estratégia gulosa*. Tal estratégia tem algumas características.

A solução da instância a ser resolvida é calculada iterativamente, sendo que a cada iteração é feita uma *escolha gulosa*, ou seja, entre todas as decisões possíveis é escolhida aquela que é mais *apetitosa*, segundo algum critério. Tal escolha jamais precisará ser desfeita.

Um algoritmo baseado nessa estratégia é chamado de *algoritmo guloso*. Os algoritmos gulosos costumam ser simples e rápidos, mas a prova de sua corretude costuma ser complicada.

Um problema pode ser corretamente resolvido usando-se a estratégia gulosa se ele possui a *propriedade gulosa*. Em essência, essa propriedade estabelece que as escolhas gulosas levam necessariamente a uma solução correta, qualquer que seja a instância do problema.

Infelizmente, poucos problemas têm a propriedade gulosa. 🙄



Vejamos um exemplo de problema que possui a propriedade gulosa: dada uma coleção de números  $C = (c_1, c_2, \dots, c_n)$  e um número  $k$ , encontrar a maior subcoleção de  $C$  cuja soma dos elementos seja menor ou igual a  $k$ .

Ex:  $C = (6, 8, 10, 2, 4, 5, 7, 2, 8)$ ,  $k = 15$

Solução:  $(6, 2, 4, 2)$

Podemos resolver esse problema usando uma estratégia gulosa:

A cada iteração escolha o menor elemento de  $C$  que ainda não tiver sido escolhido.  
Repita esse procedimento até que não seja possível escolher mais nenhum elemento.  
Devolva a subcoleção formada pelos elementos escolhidos.

Seguindo essa estratégia para o exemplo acima chegaríamos à seguinte solução:

$(2, 2, 4, 5)$



Vejamos mais um exemplo: precisamos passar um troco de valor  $t$  usando a menor quantidade possível de moedas.

Podemos resolver esse problema usando a seguinte estratégia: use as moedas de maior valor.

Ex: Troco de R\$ 3,71

3 moedas de 1 real

1 moeda de 50 centavos

2 moedas de 10 centavos

1 moeda de 1 centavo

**Obs:** se tivéssemos moedas de 20 centavos a estratégia gulosa acima não funcionaria 😬





Em alguns casos, quando um problema não possui a propriedade gulosa, podemos usar a estratégia gulosa para encontrar uma “boa” solução do problema, mas não necessariamente a melhor solução.

Por exemplo, podemos aplicar a seguinte estratégia gulosa para o PMB: 😂

Coloque na mochila os itens em ordem decrescente de valor relativo, até que não seja possível colocar mais nenhum item.

Ex:  $C = 20$ ,  $n = 6$

$i$	1	2	3	4	5	6
$p_i$	5	3	7	6	10	8
$v_i$	12	7	16	13	20	14
valor relativo	2,4	2,333	2,286	2,167	2	1,75

Solução gulosa: colocar na mochila os itens 1, 2 e 3. Valor: 35

Solução ótima: colocar na mochila os itens 2, 3 e 5. Valor: 43





INSTITUTO FEDERAL  
Ceará

# Introdução à Análise de Algoritmos

Aula 05-07-2021

Estratégia Gulosa – Parte 2

Prof: Glauber Cintra

[glauberfcintra@gmail.com](mailto:glauberfcintra@gmail.com)



# REGRAS GERAIS DE CONDUTA EM AULAS REMOTAS

A sala de aula virtual é uma extensão da sala de aula presencial e, portanto, o Regulamento da Organização Didática (ROD) é o documento que rege a sua dinâmica. Ao acessar a sala de aula virtual, você estará ciente de que a violação dessas regras é passível de medidas disciplinares, tanto no âmbito do IFCE como no âmbito civil e criminal. Para que possamos manter o ambiente harmônico, respeitoso e seguro entre todos, é necessário observar algumas regras de conduta, a saber:

## **Não compartilhe a gravação das aulas**

Você não deve copiar, distribuir, modificar, reproduzir, republicar, transmitir ou comercializar qualquer informação, texto e/ou documentos contidos nas aulas em qualquer meio eletrônico, nem criar qualquer trabalho utilizando imagens, textos ou documentos dessas aulas sem ter por escrito o prévio consentimento dos envolvidos na exposição.



## Tenha tolerância e paciência com possíveis falhas tecnológicas e eventuais limitações pessoais

Falhas técnicas poderão acontecer, seja com o professor, com colegas ou com você mesmo. Tenha paciência, procure manter a calma e contornar o problema com discrição e gentileza.

## Prepare-se para a aula virtual

Vista-se adequadamente e escolha na sua casa o local mais apropriado (se possível, separado de outras pessoas e das atividades que estiverem sendo realizadas por elas), para que haja o máximo de atenção na aula.

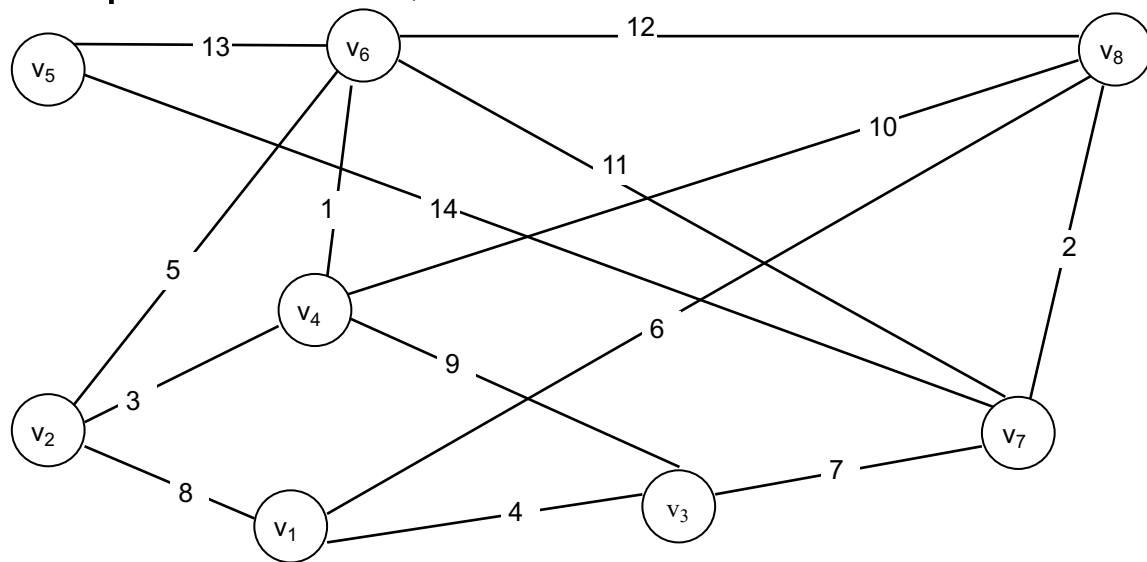
## Desative o microfone

Ao acionar seu aparelho, desative o microfone. Essa ação impedirá que, num momento de distração, você compartilhe uma fala ou ruídos indesejados. Seu celular deve ficar no silencioso. Evite também interromper a fala dos demais participantes e, pelo *chat*, peça a palavra ao professor quando quiser fazer algum comentário ou esclarecer alguma dúvida.

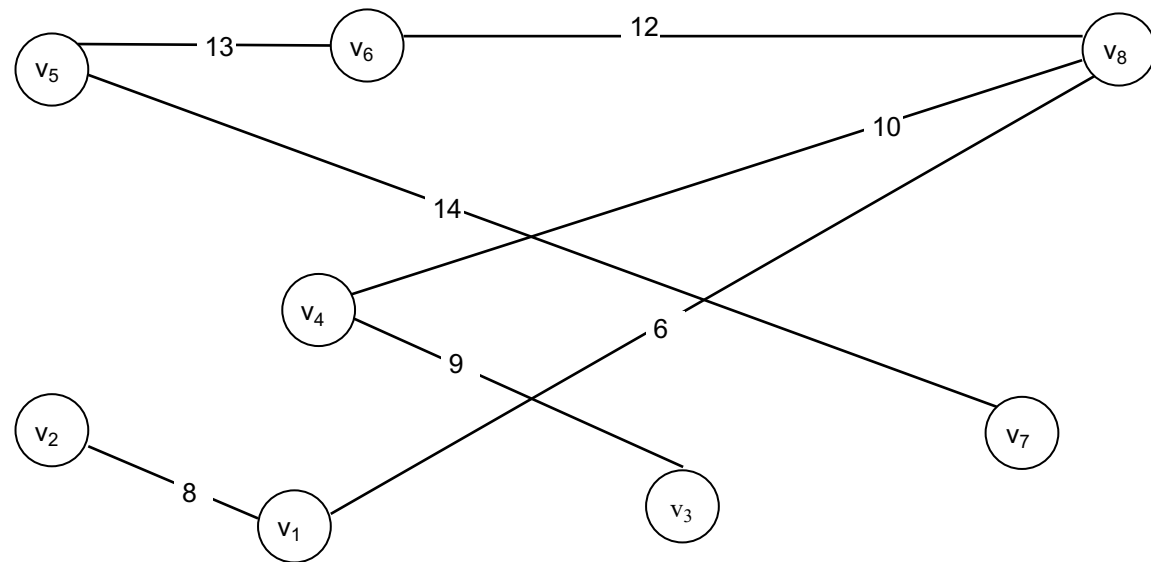


# Árvore Geradora Mínima

Seja  $G$  um grafo conexo no qual cada aresta tem um custo. Uma árvore geradora de  $G$  é um subgrafo de  $G$  que é conexo, acíclico e contém todos os vértices de  $G$ .



$G$



Árvore geradora de  $G$

O custo de uma árvore geradora é a soma dos custos das arestas que a compõem. O Problema da Árvore Geradora Mínima (AGM) consiste em, dado um grafo conexo no qual cada aresta tem um custo, encontrar uma AGM do grafo.

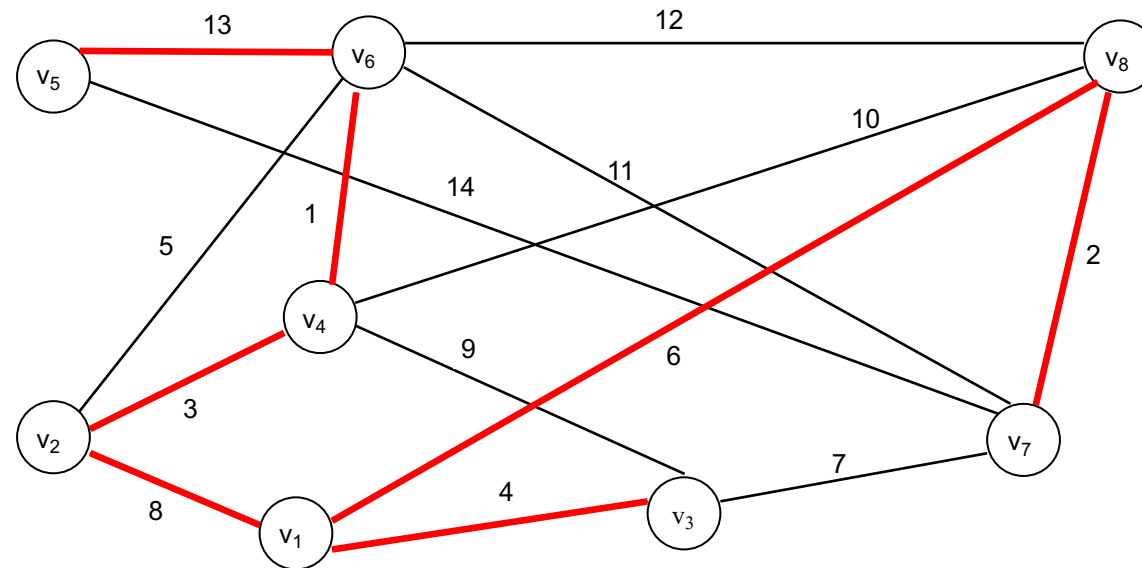
Esse problema possui a propriedade gulosa e portanto podemos resolvê-lo usando algoritmos gulosos.



# Algoritmo de Kruskal

O Algoritmo de Kruskal utiliza a seguinte estratégia gulosa: construa a AGM escolhendo as arestas em ordem crescente de custo, sem permitir a formação de ciclos.

Ex:



Custo: 37

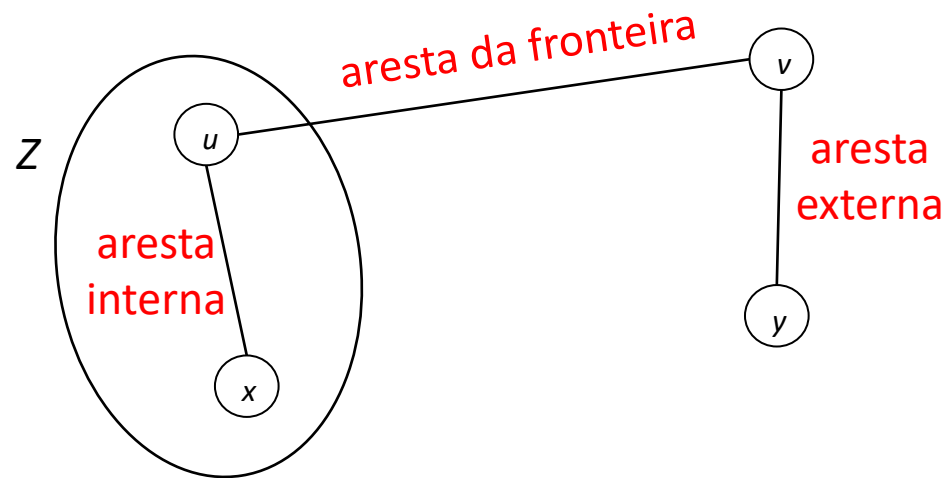


# Algoritmo de Prim

Podemos resolver o problema da AGM usando o *Algoritmo de Prim*. Tal algoritmo também é guloso, simples e rápido. Antes de descrevê-lo, precisamos de uma definição.

Seja  $Z$  um conjunto de vértices. Dizemos que uma aresta  $uv$  está na fronteira de  $Z$  se  $u \in Z$  e  $v \notin Z$  ou  $v \in Z$  e  $u \notin Z$ .

Ex:



Iniciamos o Algoritmo de Prim com um conjunto  $Z$  de vértices contendo um único vértice. A cada iteração escolhemos uma aresta da fronteira de  $Z$  que possua o menor custo para fazer parte da AGM. Seja  $uv$  a aresta escolhida, com  $u \in Z$  e  $v \notin Z$ . Incluimos o vértice  $v$  em  $Z$  e iniciamos uma nova iteração.

Ex:

