

ESDC5: Estruturas de Dados II

Aula 04: Programação Dinâmica: Teoria e Prática **(Implementações)**

Lista de Exercício – Listex 03

Breno Lisi Romano

<http://sites.google.com/site/blromano>

Instituto Federal de São Paulo – IFSP São João da Boa Vista
Bacharelado em Ciência da Computação – 3º Semestre



**INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA**
SÃO PAULO
Campus São João da Boa Vista



Trabalhos para Casa (1)

- **Exercício 01 – Aplicar o Algoritmo Problema da Mochila 0-1 para resolução do seguinte problema:**
 - $W = 10, n = 4, w = [8, 1, 5, 4]$ e $c = [500, 1000, 300, 210]$
 - Apresentar a solução no formato da Matriz para formulação por PD e a resposta exatamente como elaborado nos slides



Trabalhos para Casa (2)

- **Exercício 02 – Aplicar o Algoritmo para determinar a Subcadeia Comum Máxima para resolução do seguinte problema:**
 - $X = \text{abcbdad}$ e $Y = \text{bdcaba}$
 - Apresentar a solução no formato da Matriz para formulação por PD e a resposta exatamente como elaborado nos slides



Trabalhos para Casa (3)

- **Exercício 03 – Aplicar o Algoritmo “Coin Change” para as seguintes condições:**
 - $V = 7$; $n = 4$ e denominação = $\{1, 3, 4, 5\}$
 - Apresentar a solução no formato da Tabela para formulação por PD e a resposta exatamente como elaborado nos slides



Trabalhos para Casa (4)

- **Exercício 04 – Aplicar o Algoritmo de Kadane para encontrar o intervalo de maior soma da sequência:**
 - $A = [4, -5, 4, -3, 4, 4, -4, 4, -5]$ e $n = 9$
 - Apresentar a solução no formato da Tabela para formulação por PD e a resposta exatamente como elaborado nos slides



Trabalhos para Casa (5)

■ Exercício 05 – Implementar o Problema da Mochila 0-1, utilizando-se a técnica de Programação Dinâmica.

- Deve-se imprimir a matriz auxiliar para aplicação da programação dinâmica
- Imprimir também o Custo total dos itens selecionados e os itens selecionados
- Sugestão, utilizar a linguagem C
- Devem ser manipuladas as seguintes informações:
 - n: número de itens que podem ser adicionados na mochila
 - W: peso máximo da mochila
 - c[]: array de custo dos itens a serem adicionados na mochila
 - w[]: array de peso dos itens a serem adicionados na mochila
 - x[]: array auxiliar para identificar os itens selecionados
 - z[][]: matriz utilizada para aplicação da programação dinâmica
 - n: número de itens que podem ser adicionados na mochila

```
"D:\Google Drive\AULAS-BRENO\IFSP\Acadêmico\Disciplinas\ESDC5 (Su...  -  □  ×

Resolucao do problema da Mochila 0-1:
Forneça a capacidade maxima da mochila (W): 7

Quantos itens distintos e unicos existem: 4
Peso (w) do item 1: 2
Custo (c) do item 1: 10

Peso (w) do item 2: 1
Custo (c) do item 2: 7

Peso (w) do item 3: 6
Custo (c) do item 3: 25

Peso (w) do item 4: 5
Custo (c) do item 4: 24

Possiveis Itens a serem adicionados:
Item 2 - w: 2 / c: 10
Item 3 - w: 1 / c: 7
Item 4 - w: 6 / c: 25
Item 5 - w: 5 / c: 24

Matriz - Programacao Dinamica:
  0   0   0   0   0   0   0
  0   0  10  10  10  10  10
  0   7  10  17  17  17  17
  0   7  10  17  17  17  25
  0   7  10  17  17  24  31  34

Solucao do problema da Mochila 0-1:
Custo Total dos Itens Selecionados: 34

Itens Selecionados para a Mochila:
Item 1 - w: 2 / c: 10
Item 4 - w: 5 / c: 24
Process returned 0 (0x0)   execution time : 43.175 s
Press any key to continue.
```



Trabalhos para Casa (6)

- **Exercício 06 – Implementar o Problema do Subarranjo máximo, utilizando-se a técnica de Programação Dinâmica, com foco no Algoritmo de Kadane.**
 - Deve-se solicitar o tamanho do array a ser analisado e cada um dos elementos a serem inseridos
 - Imprimir o Array a Ser Analisado
 - Imprimir o Valor do Subarranjo Máximo utilizando-se o Algoritmo de Kadane
 - Sugestão, utilizar a linguagem C

```
"D:\Google Drive\AULAS-BRENO\IFSP\Acadêmico\Disciplinas\ES...  — □ ×  
  
Resolucao do problema do Subarranjo Maximo - Kadane:  
Forneca o tamanho do array (n): 9  
Entre com o 1 elemento do array: -2  
Entre com o 2 elemento do array: 1  
Entre com o 3 elemento do array: -3  
Entre com o 4 elemento do array: 4  
Entre com o 5 elemento do array: -1  
Entre com o 6 elemento do array: 2  
Entre com o 7 elemento do array: 1  
Entre com o 8 elemento do array: -5  
Entre com o 9 elemento do array: 4  
  
Imprimindo o Array Analisado: -2 1 -3 4 -1 2 1 -5 4  
O Valor do Subarranjo Maximo eh: 6  
  
Process returned 0 (0x0)   execution time : 28.188 s  
Press any key to continue.
```