UNIFAL-MG
Universidade Federal de Alfenas

Projeto e Análise de Algoritmos

Projeto de Algoritmos

Programação Dinâmica

Prof. Humberto Brandão
humberto@bcc.unifal-mg.edu.br

Universidade Federal de Alfenas
versão da aula: 0.3

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Programação Dinâmica não está relacionada com um programa de computador.

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Programação Dinâmica não está relacionada com um programa de computador.
  - A palavra está relacionada com um método de solução baseado em tabela.

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Programação Dinâmica não está relacionada com um programa de computador.
  - A palavra está **relacionada com um método** de solução **baseado em tabela**.
- Programação dinâmica (PD) × Divisão e conquista (DeC):
  - DeC quebra o problema em sub-problemas menores.
  - PD resolve todos os sub-problemas menores mas somente reusa as soluções ótimas.

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Dica:
  - Quando

$$\sum tamanho\ dos\ subproblemas = \mathrm{O}(n)$$

  - É provável que o algoritmo recursivo tenha complexidade polinomial;

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Dica:
  - Quando

$$\sum tamanho\ dos\ subproblemas = \mathrm{O}(n)$$

  - É provável que o algoritmo recursivo tenha complexidade polinomial;
  - Neste caso, nenhuma técnica mais elaborada é necessária;

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Dica:
  - Quando

$$\sum tamanho\ dos\ subproblemas = \mathrm{O}(n)$$

  - É provável que o algoritmo recursivo tenha complexidade polinomial;
  - Neste caso, nenhuma técnica mais elaborada é necessária;
  - Você precisa apenas da descrição recursiva da resolução;

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Dica 2:
  - Quando a divisão de um problema de tamanho *n* resulta em *n* subproblemas de tamanho *n-1* cada um
    - É provável que o algoritmo recursivo tenha complexidade exponencial;

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Dica 2:
  - Quando a divisão de um problema de tamanho *n* resulta em *n* subproblemas de tamanho *n-1* cada um
    - É provável que o algoritmo recursivo tenha complexidade exponencial;
  - Neste caso, a programação dinâmica *PODE* levar a um algoritmo mais eficiente;

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Dica 2:
  - Quando a divisão de um problema de tamanho *n* resulta em *n* subproblemas de tamanho *n-1* cada um
    - É provável que o algoritmo recursivo tenha complexidade exponencial;
  - Neste caso, a programação dinâmica *PODE* levar a um algoritmo mais eficiente;
  - A programação dinâmica calcula a solução para todos os sub-problemas, partindo dos sub-problemas menores para os maiores, armazenando os resultados em uma tabela.

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Ponto chave:
  - A vantagem é que uma vez que um sub-problema é resolvido, a resposta é armazenada em uma tabela e nunca mais é recalculado.

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Ponto chave:
  - A vantagem é que uma vez que um **sub-problema é resolvido, a resposta é armazenada em uma tabela e nunca mais é recalculado**.

- Ponto Crítico:
  - A solução do subproblema s' que compõe o problema maior s, deve ser ótima também para o problema *s*.
    - assim, pode-se aproveitá-la completamente;

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Exemplo:

$$M = M_2 \times M_3 \times M_4 \times \ldots \times M_n$$

- onde M_i é uma matriz com $i-1$ linhas e i colunas, 2 <= i <= n.

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Exemplo:

$$M = M_2 \times M_3 \times M_4 \times \ldots \times M_n$$

- onde M_i é uma matriz com ***i−1*** **linhas e** ***i*** **colunas**, 2 <= i <= n.

- Isso serve para dizer que o número de colunas da matriz *i* é igual ao número de linhas da matriz *i+1;*

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Exemplo:

$$M = M_2 \times M_3 \times M_4 \times \ldots \times M_n$$

- onde M_i é uma matriz com ***i−1* linhas e *i* colunas**, 2 <= i <= n.

- Isso serve para dizer que o número de colunas da matriz *i* é igual ao número de linhas da matriz *i+1;*

- A ordem da multiplicação pode ter um efeito enorme no número total de operações de adição e multiplicação necessárias para obter *M*.

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Considere o produto de uma matriz de dimensão $l_1 \times c_1$ por outra matriz de dimensão $l_2 \times c_2$ cujo algoritmo requer $O(l_1.c_1.c_2)$ operações.
  - Lembre-se do algoritmo de multiplicação de matrizes (3 loops aninhados);

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Considere o produto de uma matriz de dimensão $l_1 \times c_1$ por outra matriz de dimensão $l_2 \times c_2$ cujo algoritmo requer $O(l_1.c_1.c_2)$ operações.
  - Lembre-se do algoritmo de multiplicação de matrizes (3 loops aninhados);
- Lembrando: $c_1 = l_2$
- Considere o produto

$$M = M1_{[10,20]} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

  - onde as dimensões de cada matriz aparecem entre colchetes.

Programação Dinâmica

$$M = M1_{[10,20]} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Considere duas formas possíveis de multiplicar as matrizes:
  - Operações = $O(l_1.c_1.c_2)$

$M = M_1 \times (M_2 \times \underbrace{(M_3 \times M_4)})$

$50 \times 1 \times 100 = 5\,000$ operações

$M = M_1 \times \underbrace{(M_2 \times M_a)}$, sendo $M_a[50, 100]$

$20 \times 50 \times 100 = 100\,000$ operações

$M = \underbrace{M_1 \times M_b}$, sendo $M_b[20, 100]$

$10 \times 20 \times 100 = 20\,000$ operações

Total $= 125\,000$ operações

Programação Dinâmica

$$M = M1_{[10,20]} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Considere duas formas possíveis de multiplicar as matrizes:
  - Operações = $O(l_1.c_1.c_2)$

$M = M_1 \times (M_2 \times \underbrace{(M_3 \times M_4)}_{50 \times 1 \times 100 = 5\,000 \text{ operações}})$

$M = \underbrace{M_1 \times M_a}_{20 \times 50 \times 100 = 100\,000 \text{ operações}}$, sendo $M_a[50, 100]$

$M = \underbrace{M_1 \times M_b}_{10 \times 20 \times 100 = 20\,000 \text{ operações}}$, sendo $M_b[20, 100]$

Total = 125 000 operações

$M = (M_1 \times \underbrace{(M_2 \times M_3)}_{20 \times 50 \times 1 = 1\,000 \text{ operações}}) \times M_4$

$M = \underbrace{(M_1 \times M_a)}_{10 \times 20 \times 1 = 200 \text{ operações}} \times M_4$, sendo $M_a[20, 1]$

$M = \underbrace{M_b \times M_4}_{10 \times 1 \times 100 = 1\,000 \text{ operações}}$, sendo $M_b[10, 1]$

Total = 2 200 operações

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Tentar todas as ordens possíveis para minimizar o número de operações *f(n)* é de ordem exponencial em *n*, onde *f(n)* >= 2^{n-2};

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Tentar todas as ordens possíveis para minimizar o número de operações $f(n)$ é de ordem exponencial em n, onde $f(n) >= 2^{n-2}$;

- Este problema é o de tentar todas as possíveis possibilidades na colocação de parênteses;

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Tentar todas as ordens possíveis para minimizar o número de operações *f(n)* é de ordem exponencial em *n*, onde *f(n)* >= 2^{n-2};

- Este problema é o de tentar todas as possíveis possibilidades na colocação de parênteses;

- Vamos analisar no quadro, as diversas possibilidades na colocação de parênteses ao multiplicar 4 matrizes, por exemplo;

$$M = M_1 \times M_2 \times M_3 \times M_4$$

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Usando programação dinâmica é possível obter um algoritmo $O(n^3)$.

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

- Usando programação dinâmica é possível obter um algoritmo $O(n^3)$.

- Vamos voltar a matriz:

$$M = M1_{[10,20]} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Definiremos uma tabela que mostra o custo m_{ij} de obter a matriz que é resultante da multiplicação da matriz M_i até a matriz Mj; $i<=j$.

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

$O(l_1.c_1.c_2)$

$$M = M1_{[10,20]} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando $i=j$, $M_{ij} = 0$, obviamente.

$m_{11} = 0$ (10,20)	$m_{22} = 0$ (20,50)	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

$O(l_1.c_1.c_2)$

$$M = M1_{(10,20)} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando $i=j$, $M_{ij} = 0$, obviamente.

10.000=0+(10x20x50)

$m_{11} = 0$ (10,20)	$m_{22} = 0$ (20,50)	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)
$m_{12} = 10.000$ $M_1 \times M_2$ (10,50)			

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

$O(l_1.c_1.c_2)$

$$M = M1_{(10,20)} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando $i=j$, $M_{ij} = 0$, obviamente.

1.000=0+(20x50x1)

$m_{11} = 0$ (10,20)	$m_{22} = 0$ (20,50)	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)
$m_{12} = 10.000$ $M_1 \times M_2$ (10,50)	$m_{23} = 1.000$ $M_2 \times M_3$ (20,1)		

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

$O(l_1.c_1.c_2)$

$$M = M1_{(10,20)} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando *i* 5.000=0+(50x1x100) nte.

$m_{11} = 0$ (10,20)	$m_{22} = 0$ (20,50)	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)
$m_{12} = 10.000$ $M_1 \times M_2$ (10,50)	$m_{23} = 1.000$ $M_2 \times M_3$ (20,1)	$m_{34} = 5.000$ $M_3 \times M_4$ (50,100)	

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

$O(l_1.c_1.c_2)$

$$M = M1_{(10,20)} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando $i=j$, $M_{ij} = 0$, obviamente.

$m_{11} = 0$ (10,20)	$m_{22} = 0$ (20,50)	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)
$m_{12} = 10.000$ $M_1 \times M_2$ (10,50)	$m_{23} = 1.000$ $M_2 \times M_3$ (20,1)	$m_{34} = 5.000$ $M_3 \times M_4$ (50,100)	
m_{13}			

(M1x(M2xM3) = ?
((M1xM2)xM3) = ?

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

$O(l_1.c_1.c_2)$

$$M = M1_{(10,20)} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando $i=j$, $M_{ij} = 0$, obviamente.

$m_{11} = 0$ (10,20)	$m_{22} = 0$ (20,50)	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)
$m_{12} = 10.000$ (10,50) $M_1 \times M_2$	$m_{23} = 1.000$ (20,1) $M_2 \times M_3$	$m_{34} = 5.000$ (50,100) $M_3 \times M_4$	
m_{13}			

(M1x(M2xM3) = 1000+(10*20*1)=1.200
((M1xM2)xM3) = ?

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

$O(l_1.c_1.c_2)$

$$M = M1_{(10,20)} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando $i=j$, $M_{ij} = 0$, obviamente.

$m_{11} = 0$ (10,20)	$m_{22} = 0$ (20,50)	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)
$m_{12} = 10.000$ $M_1 \times M_2$ (10,50)	$m_{23} = 1.000$ $M_2 \times M_3$ (20,1)	$m_{34} = 5.000$ $M_3 \times M_4$ (50,100)	
m_{13}			

(M1x(M2xM3) = 1.000+(10*20*1)=1.200
((M1xM2)xM3) = 10.000+(10*50*1) =10.500

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

$O(l_1.c_1.c_2)$

$$M = M1_{(10,20)} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando $i=j$, $M_{ij} = 0$, obviamente.

$m_{11} = 0$ (10,20)	$m_{22} = 0$ (20,50)	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)
$m_{12} = 10.000$ $M_1 \times M_2$ (10,50)	$m_{23} = 1.000$ $M_2 \times M_3$ (20,1)	$m_{34} = 5.000$ $M_3 \times M_4$ (50,100)	
m_{13}			

Para multiplicar de M1 até M3, é melhor que seja efetuada a operação (M_2xM_3)=M_{23} e depois M_1X(M_{23}).

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

$O(l_1.c_1.c_2)$

$$M = M1_{(10,20)} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando $i=j$, $M_{ij} = 0$, obviam

1200=1000+(10x20x1)

$m_{11} = 0$ (10,20)	$m_{22} = 0$	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)
$m_{12} = 10.000$ $M_1 \times M_2$ (10,50)	$m_{23} = 1.000$ $M_2 \times M_3$ (20,1)	$m_{34} = 5.000$ $M_3 \times M_4$ (50,100)	
$m_{13} = 1.200$ $M_1 \times (M_2 \times M_3)$ (10,1)			

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

$O(l_1.c_1.c_2)$

$$M = M1_{(10,20)} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando $i=j$, $M_{ij} = 0$, obviamente.

$m_{11} = 0$ (10,20)	$m_{22} = 0$ (20,50)	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)
$m_{12} = 10.000$ $M_1 \times M_2$ (10,50)	$m_{23} = 1.000$ $M_2 \times M_3$ (20,1)	$m_{34} = 5.000$ $M_3 \times M_4$ (50,100)	
$m_{13} = 1.200$ $M_1 \times (M_2 \times M_3)$ (10,1)	m_{24}		

Para multiplicar de M2 até M4, é melhor que seja efetuada a operação (M_2xM_3)=M_{23} e depois (M_{23})xM_4.

Programação Dinâmica

UNIFAL-MG
Universidade Federal de Alfenas

$O(l_1.c_1.c_2)$

$$M = M1_{(10,20)} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando $i=j$, $M_{ij} = 0$, obviam

3000=1000+(20x1x100)

$m_{11} = 0$ (10,20)	$m_{22} = 0$ (20,5	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)
$m_{12} = 10.000$ $M_1 \times M_2$ (10,50)	$m_{23} = 1.00$ $M_2 \times M_3$ (20,1)	$m_{34} = 5.000$ $M_3 \times M_4$ (50,100)	
$m_{13} = 1.200$ $M_1 \times (M_2 \times M_3)$ (10,1)	$m_{24} = 3.000$ $(M_2 \times M_3) \times M_4$ (20,100)		

Programação Dinâmica

$O(l_1.c_1.c_2)$

$$M = M1_{(10,20)} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando $i=j$, $M_{ij} = 0$, obviamente.

$m_{11} = 0$ (10,20)	$m_{22} = 0$ (20,50)	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)
$m_{12} = 10.000$ $M_1 \times M_2$ (10,50)	$m_{23} = 1.000$ $M_2 \times M_3$ (20,1)	$m_{34} = 5.000$ $M_3 \times M_4$ (50,100)	
$m_{13} = 1.200$ $M_1 \times (M_2 \times M_3)$ (10,1)	$m_{24} = 3.000$ $(M_2 \times M_3) \times M_4$ (20,100)		
m_{14}			

Para multiplicar de M1 até M4, é melhor que seja efetuada a operação (M1x(M_2xM_3))=M_{13} e depois (M_{13})xM_4.

Programação Dinâmica

$O(l_1.c_1.c_2)$

$$M = M1_{(10,20)} \times M2_{[20,50]} \times M3_{[50,1]} \times M4_{[1,100]}$$

- Quando $i=j$, $M_{ij} = 0$, obviamente.

$m_{11} = 0$ (10,20)	$m_{22} = 0$ (20,50)	$m_{33} = 0$ (50,1)	$m_{44} = 0$ (1,100)
$m_{12} = 10.000$ $M_1 \times M_2$ (10,50)	$m_{23} = 1.000$ $M_2 \times M_3$ (20,1)	$m_{34} = 5.000$ $M_3 \times M_4$ (50,100)	
$m_{13} = 1.200$ $M_1 \times (M_2 \times M_3)$ (10,1)	$m_{24} = 3.000$ $(M_2 \times M_3) \times M_4$ (20,100)		
$m_{14} = 2.200$ $(M_1 \times (M_2 \times M_3)) \times M_4$ (10,100)			

2200=1200+(10x1x1000)

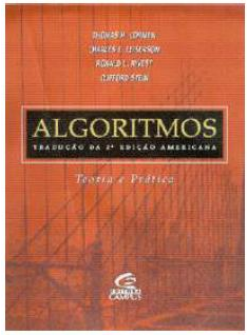
Programação Dinâmica

- Função de Fibonacci:

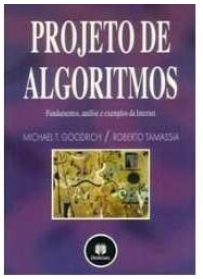
$$fib(n) = \begin{cases} 1, se\ n = 0; \\ 1, se\ n = 1; \\ fib(n-1) + fib(n-2), caso\ contrário \end{cases}$$

Bibliografia

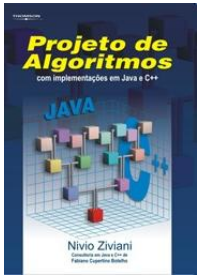
- CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; (2002). Algoritmos – Teoria e Prática. Tradução da 2ª edição americana. Rio de Janeiro. Editora Campus.



- TAMASSIA, ROBERTO; GOODRICH, MICHAEL T. (2004). Projeto de Algoritmos - Fundamentos, Análise e Exemplos da Internet.



- ZIVIANI, N. (2007). Projeto e Algoritmos com implementações em Java e C++. São Paulo. Editora Thomson;



- Material de aulas do Professor Loureiro (DCC-UFMG)
  - http://www.dcc.ufmg.br/~loureiro/paa/