AULA 19

Prof. Mathias

Programação Dinâmica

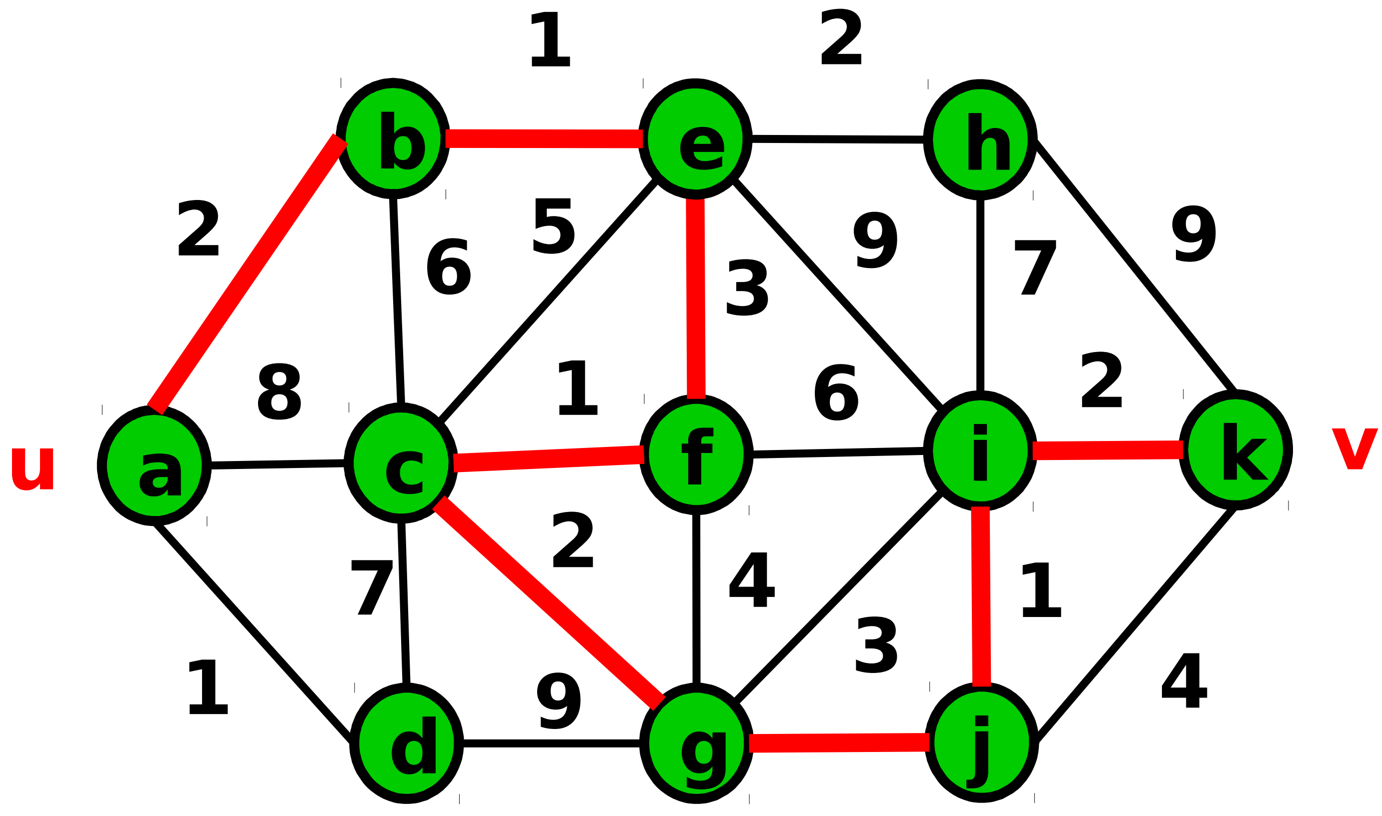
Análise de Algoritmos

Agenda

- Aula anterior
- Introdução
- Exemplo
- Exercícios
- Próxima aula

Aula Anterior

- Algoritmo de Huffman – Método Guloso



A, B, E, F, C, G, J, I, K = 13

Agenda

- Aula anterior
- Introdução

Introdução

- Método de dividir e conquistar, resolve problemas combinando as soluções para subproblemas.
- “Programação” refere-se a um método de tabular e não de codificar.
- Utilizado em problemas de otimização;
- Combina soluções de subproblemas;
- Subproblemas são dependente;
- A relação problema: geralmente expressada de forma recursiva.

Introdução

- Diferença entre o método guloso e programação dinâmica são as decisões geradas, pois para a dinâmica muitas decisões são geradas e para o guloso apenas uma decisão se produz.

Introdução

- Passos para produzir um algoritmo de programação dinâmica:
  - Caracterizar a estrutura de uma solução ótima
  - Definir recursivamente o valor de uma solução ótima
  - Calcular o valor de uma solução ótima de baixo para cima

Agenda

- Aula anterior
- Introdução
- Exemplo

Exemplo

- Como realizar multiplicação de matrizes realizando um número mínimo de operações?

- Multiplicar n matrizes, realizando o número mínimo de operações.

$$M = M_1 * M_2 * \ldots M_n$$

- As matrizes são multiplicadas aos pares, sabemos também que cada matriz M possui dimensões que devem ser respeitadas. Então esse é o problema:

  - Como determinar a melhor forma de realizar as multiplicações, ou seja, aquela que oferece o menor número de operações.

Exemplo

- Determinar com algoritmos de “força bruta” é impossível pois a complexidade é de 2^n.
- Pelo método de programação dinâmica alcançamos o resultado esperado.

Exemplo

- Exemplo:

$$M = M_1 * M_2 * M_3 * M_4$$

com

$$M = \{200, 2, 30, 20, 5\}$$

- Quantas multiplicações são realizadas nas sequências:

  - $M = (((M_1 * M_2) * M_3) * M_4)$
  - $M = (M_1 * ((M_2 * M_3) * M_4))$

Exemplo

- Exemplo:

$$M = M_1 * M_2 * M_3 * M_4$$

com

$$M = \{200, 2, 30, 20, 5\}$$

- Quantas multiplicações são realizadas nas sequências:

  - $M = (((M_1 * M_2) * M_3) * M_4)$ = 152.000 operações
  - $M = (M_1 * ((M_2 * M_3) * M_4))$ = 3400 operações

Exemplo

- Considere a seguinte fórmula:

Seja Mij o mínimo de operações na realização de um produto

Mi+Mi+1...*Mj*

Como calcular então Mij?

Exemplo

- Então:
  - Para resolver este problema, tudo que precisamos é saber qual o melhor índice k tal que:

$$M = (M_1 * M_2 *\dots M_4).*(M_{k+1}*M_{k+2}*\dots*M_{k+n})$$

onde k varia de 1 até $n-1$

Exemplo

- Precisamente:
  - Seja Mij o número mínimo de operações para realizar o produto:
    Mi * Mi+1* … Mj

  - Podemos calcular Mij.

$$M_{ij} = min(M_{ik} + M_{k+1j} + b_{i-1} b_k b_j),\ k=i,\ldots,j-1$$

Exemplo

- Então, tal expressão constitui uma solução dinâmica que propõe intuitivamente uma recorrência a ser implementada. Porém uma implementação botton-up seria mais interessamente devido a consulta apenas nos valores a serem calculados.

Exemplo

- **Algoritmo necessário teria as seguintes características:**
  1. Os Mi,i são calculados, para 1 ≤ i ≥ n, onde claramente Mii = 0 para todo i;
  2. Os Mi,i+1 são calculados, para 1 ≤ i ≥ n-1
  3. Os Mi,i+2 são calculados, para 1 ≤ i ≥ n-2 Mi,i+1 são calculados, para 1 ≤ i ≥ n-1

Exemplo

- **Veja:**

M1	*	M2	*	M3	*	M4
200x2		2x30		30X20		20x5
b0*b1		b1*b2		b2*b3		b3*b4

- $M_{ij} = \min\{M_{i,k}+M_{k+1,j}+b_{i-1}b_k b_j\}$
  Para i = j temos
- $M_{1,1} = M_{2,2} = M_{3,3} = M_{4,4} = 0$
  Para j=i+1 e i++ até n-1
- $M_{1,2} = 12000$; $M_{2,3} = 1200$; $M_{3,4} = 3000$
  Para j++
- $M_{1,3} = \min\{(M_{1,1}+M_{2,3}+b_0b_1b_3), (M_{12}+M_{3,3}+b_0b_2b_3)\}$
  $= \min\{(0+1200+8000), (12000+0+120000)\}$
  $= 9200$ (M1*(M2*M3))

Exemplo

- **Veja:**

M1	*	M2	*	M3	*	M4
200x2		2x30		30X20		20x5
b0*b1		b1*b2		b2*b3		b3*b4

- $M_{ij} = \min\{M_{i,k}+M_{k+1,j}+b_{i-1}b_k b_j\}$, k =i, …, j-1
- $M_{2,4} = \min\{(M_{2,k}+M_{k+1,4}+b_1 b_k b_4)$, k=2,3
- $M_{2,4} = \min\{(M_{2,2}+M_{3,4}+b_1 b_2 b_4), (M_{2,3}+M_{4,4}+b_1 b_3 b_4)\}$

$= \min\{(0+3000+300), (1200+0+200)\}$

$= 1400$ ((M2*M3)*M4)

Exemplo

- **Veja:**

M1	*	M2	*	M3	*	M4
200x2		2x30		30X20		20x5
b0*b1		b1*b2		b2*b3		b3*b4

- $M_{ij} = \min\{M_{i,k}+M_{k+1,j}+b_{i-1}b_k b_j\}$, k =i, …, j-1
- $M_{1,4} = \min\{(M_{1,k}+M_{k+1,4}+b_0 b_k b_4)$, k=1,..,3
- $M_{1,4} = \min\{(M_{1,1}+M_{2,4}+b_0b_1b_4),(M_{1,2}+M_{3,4}+b_0b_2b_4),(M_{1,3}+M_{4,4}+b_0b_3b_4)\}$

  $= \min\{(0+1400+2000),(1200+3000+30000),(9200+0+20000)\}$

  = 3400 (M1*((M2*M3)*M4))

  A Resposta final é: (M1*((M2*M3)*M4))

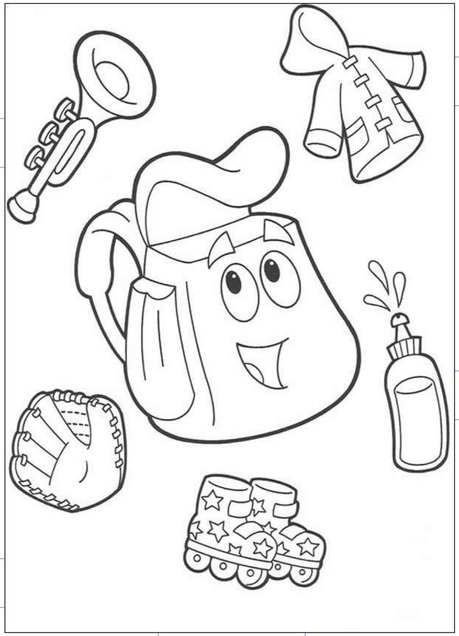
Exemplo 2

- **Problema da mochila**
  - **Conjunto de elementos a serem escolhidos para inserir em uma mochila, quais elementos devem ser escolhidos para obter maior valor na mochila?**
    - **Solução**
      - **Parte do pensamento:**
        - **Elemento n pode ou não estar na mochila.**

$$f(i,w) = \begin{cases} f(i-1,w) \; se \; pi > w \\ max\{f(i-1,w-pi)+vi, f(i-1,w)\} \; se \; pi \leq w \end{cases}$$

Exemplo 2

- **Problema da mochila – Solução**



R$: 4P:2

R$:1 P:1

R$:2 P:2

R$:2 P:1

R$:3 P:4

$$f(i,w)=\begin{cases} f(i-1,w)\ se\ pi>w \\ max\{f(i-1,w-pi)+vi, f(i-1,w)\}\ se\ pi\le w \end{cases}$$

						w0	w1	w2	w3	w4	w5
						0	1	2	3	4	5
vi	0	pi	0	i=	0	0	0	0	0	0	0
vi	4	pi	2	i=	1	0					
vi	2	pi	1	i=	2	0					
vi	3	pi	4	i=	3	0					
vi	2	pi	2	i=	4	0					
vi	1	pi	1	i=	5	0					

Mathias Talevi Betim

Exemplo 2

$$f(i,w)=\begin{cases} f(i-1,w)\ se\ pi>w \\ max\{f(i-1,w-pi)+vi, f(i-1,w)\}\ se\ pi\leq w \end{cases}$$

- **Problema da mochila – Solução**

						w0	w1	w2	w3	w4	w5
						0	1	2	3	4	5
vi	0	pi	0	i=	0	0	0	0	0	0	0
vi	4	pi	2	i=	1	0	0	4	4	4	4
vi	2	pi	1	i=	2	0					
vi	3	pi	4	i=	3	0					
vi	2	pi	2	i=	4	0					
vi	1	pi	1	i=	5	0					

vi=4	pi=2	i=1	w=1	pi>w	f(i-1, w)	f(0, 1)	0
vi=4	pi=2	i=1	w=2	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(0,0)+4, f(0, 2)	4
vi=4	pi=2	i=1	w=3	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(0,1)+4, f(0, 3)	4
vi=4	pi=2	i=1	w=4	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(0,2)+4, f(0, 4)	4
vi=4	pi=2	i=1	w=5	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(0,3)+4, f(0, 5)	4

Exemplo 2

$$f(i,w)=\begin{Bmatrix} f(i-1,w)\ se\ pi>w \\ max\{f(i-1,w-pi)+vi, f(i-1,w)\}\ se\ pi\leq w \end{Bmatrix}$$

- **Problema da mochila – Solução**

						w0	w1	w2	w3	w4	w5
						0	1	2	3	4	5
vi	0	pi	0	i=	0	0	0	0	0	0	0
vi	4	pi	2	i=	1	0	0	4	4	4	4
vi	2	pi	1	i=	2	0	2	4	6	6	6
vi	3	pi	4	i=	3	0					
vi	2	pi	2	i=	4	0					
vi	1	pi	1	i=	5	0					

vi=2	pi=1	i=2	w=1	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(1,0)+2, f(1, 1)	2
vi=2	pi=1	i=2	w=2	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(1,1)+2, f(1, 2)	4
vi=2	pi=1	i=2	w=3	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(1,2)+2, f(1, 3)	6
vi=2	pi=1	i=2	w=4	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(1,3)+2, f(1, 4)	6
vi=2	pi=1	i=2	w=5	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(1,4)+2, f(1, 5)	6

Exemplo 2

$$f(i,w)=\begin{cases} f(i-1,w)\ se\ pi>w \\ max\{f(i-1,w-pi)+vi, f(i-1,w)\}\ se\ pi\leq w \end{cases}$$

- **Problema da mochila – Solução**

						w0	w1	w2	w3	w4	w5
						0	1	2	3	4	5
vi	0	pi	0	i=	0	0	0	0	0	0	0
vi	4	pi	2	i=	1	0	0	4	4	4	4
vi	2	pi	1	i=	2	0	2	4	6	6	6
vi	3	pi	4	i=	3	0	2	4	6	6	6
vi	2	pi	2	i=	4	0					
vi	1	pi	1	i=	5	0					

vi=3	pi=4	i=3	w=1	pi>w	f(i-1, w)	f(2, 1)	2
vi=3	pi=4	i=3	w=2	pi>w	f(i-1, w)	f(2, 2)	4
vi=3	pi=4	i=3	w=3	pi>w	f(i-1, w)	f(2, 3)	6
vi=3	pi=4	i=3	w=4	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(2,0)+3, f(2, 4)	6
vi=3	pi=4	i=3	w=5	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(2,1)+3, f(2, 5)	6

Exemplo 2

$$f(i,w)=\begin{cases} f(i-1,w)\ se\ pi>w \\ max\{f(i-1,w-pi)+vi, f(i-1,w)\}\ se\ pi\leq w \end{cases}$$

- **Problema da mochila – Solução**

						w0	w1	w2	w3	w4	w5
						0	1	2	3	4	5
vi	0	pi	0	i=	0	0	0	0	0	0	0
vi	4	pi	2	i=	1	0	0	4	4	4	4
vi	2	pi	1	i=	2	0	2	4	6	6	6
vi	3	pi	4	i=	3	0	2	4	6	6	6
vi	2	pi	2	i=	4	0	2	4	6	6	8
vi	1	pi	1	i=	5	0					

vi=2	pi=2	i=4	w=1	pi>w	f(i-1, w)	f(3, 1)	2
vi=2	pi=2	i=4	w=2	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(3,0)+2, f(3, 2)	4
vi=2	pi=2	i=4	w=3	pi>w	f(i-1, w-pi)+v1, f(i-1, w)	f(3,1)+2, f(3, 3)	6
vi=2	pi=2	i=4	w=4	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(3,2)+2, f(3, 4)	6
vi=2	pi=2	i=4	w=5	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(3,3)+2, f(3, 5)	8

Exemplo 2

$$f(i,w)=\begin{cases} f(i-1,w)\ se\ pi>w \\ max\{f(i-1,w-pi)+vi, f(i-1,w)\}\ se\ pi\leq w \end{cases}$$

- **Problema da mochila – Solução**

						w0	w1	w2	w3	w4	w5
						0	1	2	3	4	5
vi	0	pi	0	i=	0	0	0	0	0	0	0
vi	4	pi	2	i=	1	0	0	4	4	4	4
vi	2	pi	1	i=	2	0	2	4	6	6	6
vi	3	pi	4	i=	3	0	2	4	6	6	6
vi	2	pi	2	i=	4	0	2	4	6	6	8
vi	1	pi	1	i=	5	0	2	4	6	7	8

vi=1	pi=1	i=5	w=1	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(4,0)+1, f(4, 1)	2
vi=1	pi=1	i=5	w=2	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(4,1)+1, f(4, 2)	4
vi=1	pi=1	i=5	w=3	pi>w	f(i-1, w-pi)+v1, f(i-1, w)	f(4,2)+1, f(4, 3)	6
vi=1	pi=1	i=5	w=4	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(4,3)+1, f(4, 4)	7
vi=1	pi=1	i=5	w=5	pi≤w	f(i-1, w-pi)+v1, f(i-1, w)	f(4,4)+1, f(4, 5)	8

Exemplo 2

$$f(i,w)=\begin{cases} f(i-1,w)\ se\ pi>w \\ max\{f(i-1,w-pi)+vi, f(i-1,w)\}\ se\ pi\leq w \end{cases}$$

- **Problema da mochila – Solução**

						w0	w1	w2	w3	w4	w5
						0	1	2	3	4	5
vi	0	pi	0	i=	0	0	0	0	0	0	0
vi	4	pi	2	i=	1	0	0	4	4	4	4
vi	2	pi	1	i=	2	0	2	4	6	6	6
vi	3	pi	4	i=	3	0	2	4	6	6	6
vi	2	pi	2	i=	4	0	2	4	6	6	8
vi	1	pi	1	i=	5	0	2	4	6	7	8

Resposta: A soma dos valores é 8; os valores que pertencem a mochila são 2,2,4.

Exemplo 3

- **Programação de linha de montagem**

  A Empresa x Computadores possuem duas linhas de montagens, uma placa mae entra em uma linha de montagem tem as peças adicionadas a ela em uma série de estações e um computador sai pronto no final da linha.

  Cada linha de montagem possui n estações, numeradas com j=1,2..n. Denotamos o j-ésima estação na linha i (onde i é 1 ou 2) por $S_{i,j}$. A j-ésima estação da na linha 1 ($S_{1,j}$) executa a mesma ação que a j-ésima ação da linha 2 ($S_{2,j}$). Porém as estações foram construídas em épocas diferentes e com tecnologias diferentes, de forma que o tempo exigido em cada estação varia, até mesmo entre as estações na mesma posição nas duas linhas.

Exemplo 3

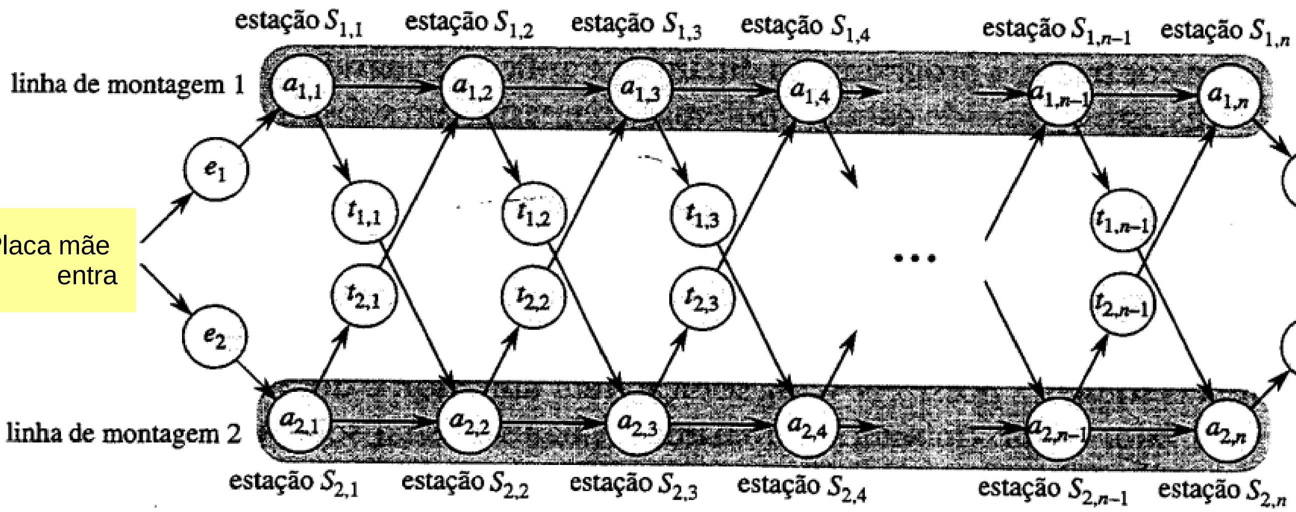
- **Programação de linha de montagem**

  Denotamos o tempo de montagem exigido na estação $S_{i,j}$ por a_{ij}. Conforme no exemplo um computador entra na estação 1 de uma das duas linhas de montagem e avança de cada estação para a seguinte. Também há um tempo de entrada e_i para a placa mãe entrar na linha de montagem $_i$, e um tempo de saída x_i, para o computador concluído sair da linha de montagem $_i$.

Exemplo 3

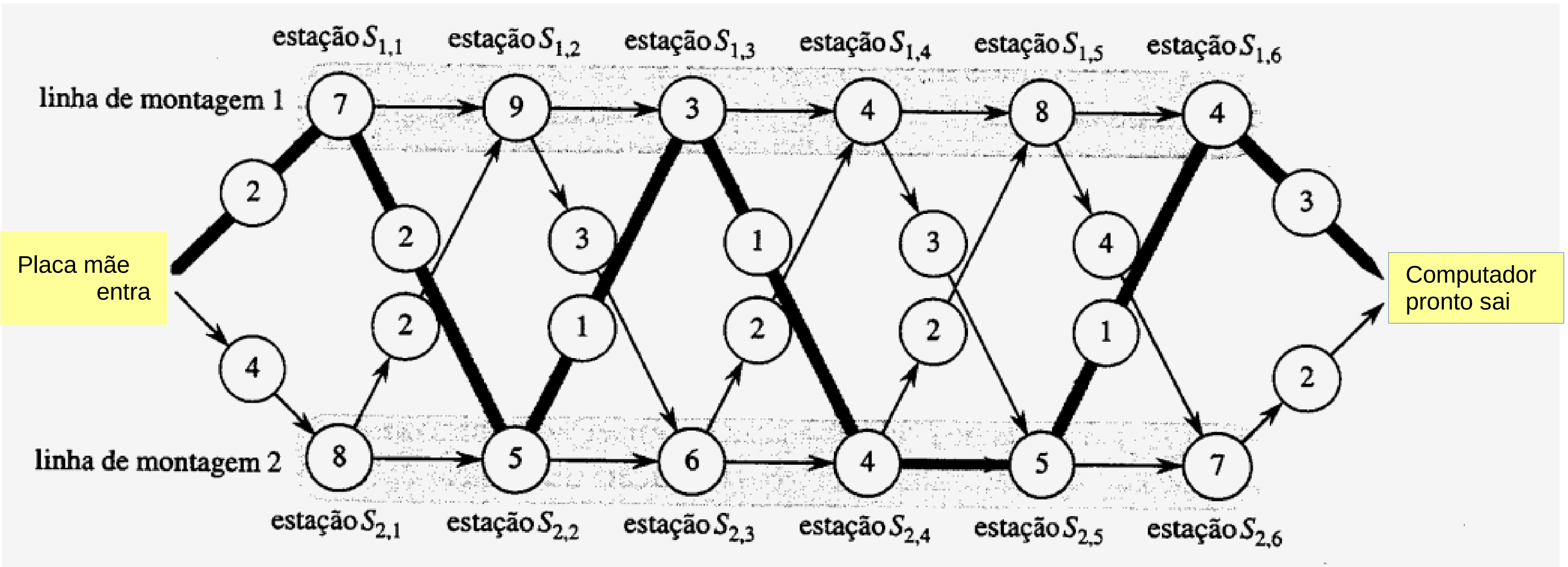
- **Programação de linha de montagem**

Placa mãe entra



Computador pronto sai

Exemplo 3



s	1	2	3	4	5	6
1	7	9	3	4	8	4
2	8	5	6	4	5	7

e	
1	2
2	4

x	
1	3
2	2

t	1	2	3	4	5
1	2	3	1	3	4
2	2	1	2	2	1

Agenda

- Aula anterior
- Introdução
- Exemplo
- Exercícios

Exercícios

- Aplique o algoritmo para multiplicar as 5 matrizes, onde b={30,35,15,5,10,20}

- Considerar a seguinte solução gulosa: a cada passo selecione o produto que requer um número mínimo de operações, como ficaria?

- Se precisar o número máximo de operações como seria esse algoritmo (solução)?

Exercícios

- Um motoboy precisa entregar alguns produtos e sua caixa tem capacidade de 10 kg, considere os produtos abaixo e determine quais produtos ele deve carregar para obter maior valor agregado na sua viagem.

Produtos	p1	p2	p3	p4	p5
Valores	4	6	4	2	2
Pesos	3	2	4	6	3

Exercícios - Produção

- Considere os exercícios citado nas aulas.

Agenda

- Aula anterior
- Introdução
- Exemplo
- Exercícios
- Próxima aula

Próxima aula

- **Classes de problemas (p, np, np-completo)**

AULA 19

Prof. Mathias