

# Matlab, Octave, algoritmos, grafos

**Leitura:** Slides 1 a 33 sobre Matlab/Octave do Prof. Afonso.

---

## Exercício 1

```
> a = [ 1, 2, 3 ];  
> b = [ 4; 2; 1 ];  
> a'*b'
```

Identifique a resposta que será obtida:

①  $\begin{matrix} 4 & 8 & 12 \\ 2 & 4 & 6 \\ 1 & 2 & 3 \end{matrix}$

② 4

③ 11

④  $\begin{matrix} 4 & 2 & 1 \\ 8 & 4 & 2 \\ 12 & 6 & 3 \end{matrix}$

⑤ *Error. Nonconformant arguments.*

## Exercício 2

Fazer em Matlab/Octave (randômico = uniforme em  $(0,1)$ )

- 1 Um programa que cria dois vetores randômicos e calcula o seu produto escalar.
- 2 Um programa que cria duas matrizes randômicas e calcula o seu produto.
- 3 Um programa que mede o tempo necessário para calcular o produto de duas matrizes randômicas de  $100 \times 100$ . Usar os comandos *tic* e *toc*.
- 4 Um programa que, para um valor dado de  $x$ , calcule numericamente a probabilidade  $P(x)$  de que uma matriz randômica de  $3 \times 3$  tenha determinante menor (em valor absoluto) que  $x$ . Desenhe a função  $P(x)$ .

## lista1.m

```
tic;  
nn=100;  
for i=1:10000  
    a=rand(nn); b=rand(nn);  
    c=a*b;  
endfor  
toc
```

Elapsed time is 3.94071 seconds.

0.394 milisegundos para multiplicar duas matrizes  $100 \times 100$ .

```

tic; n=1000; p=3;
i=1:n+1; dx=2.0/n; x=(i-1)*dx;
m=10000;
eve=zeros(1,n+1);
for k=1:m
    d=abs(det(rand(p))); j=floor(d/dx);
    if (j<=n)
        eve(j+1)=eve(j+1)+1;
    endif
endfor
toc

```

Complexidade  $O(m)$ , independente de  $n$ . Dependência com  $p$ :

|       |      |      |      |      |      |     |
|-------|------|------|------|------|------|-----|
| p     | 3    | 9    | 27   | 81   | 243  | 729 |
| tempo | 0.19 | 0.23 | 0.40 | 1.99 | 21.8 | 317 |

```
P(1,1)=0;
for k=2:n+1
    P(1,k)=P(1,k-1)+eve(k);
endfor
P = P/m;
```

Complexidade  $O(n)$ , independente de  $m$ . Total:  $O(n + m)$ .  
**Verifique se seu algoritmo tem a mesma complexidade.**

```
figure 1
plot(x,eve,"linewidth",2)
title("Eventos entre x e x + dx");
xlabel("x"); ylabel("#eve");
figure 2
plot(x,P,"linewidth",2)
title("Probabilidade de |det(A)|<x");
xlabel("x"); ylabel("P");
```

## Exercício 3

Acabar a leitura dos slides do Prof. Afonso sobre Matlab.

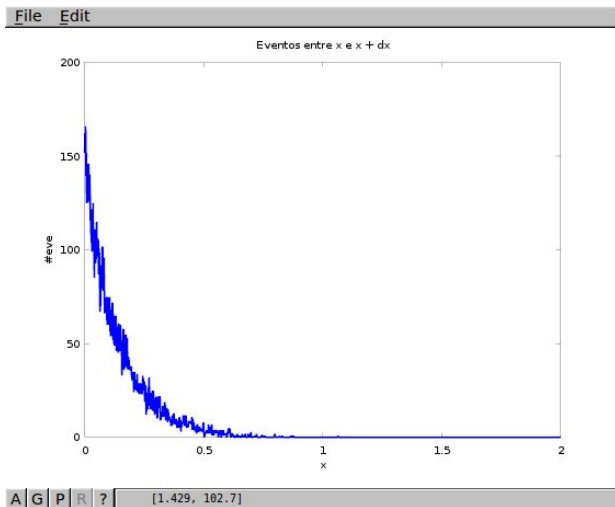


Figura 1.

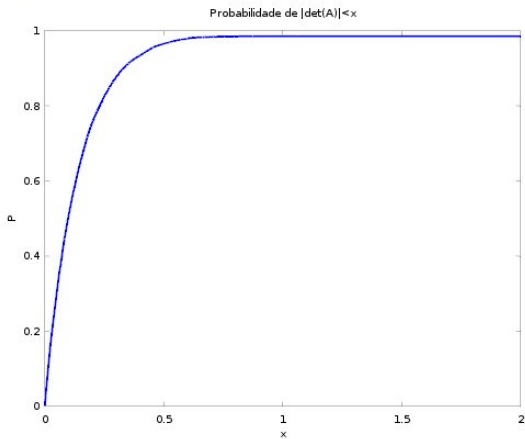


Figura 2.



## Exercício 4

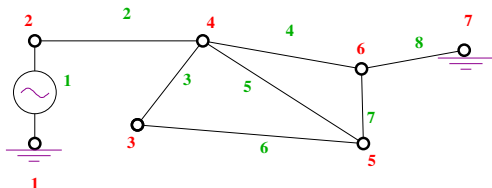
*Qual a resposta de*

```
x=[1:2:11]; i=[1:3 length(x):-1:length(x)-2]; x(i)
```

## Exercício 5

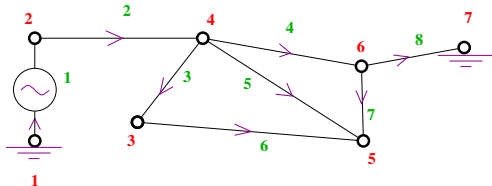
*Seja  $X$  uma matriz  $n \times m$ . Escreva os seguintes códigos:*

- 1 *Calcular uma matriz  $Y$  que consista de apenas as linhas e colunas pares de  $X$ .*
- 2 *Seja*  
 $i=[1\ 3\ 7]$ ;  $j=[2\ 4]$ ;  
*Calcular uma matriz  $Y$  que consista apenas das linhas de  $X$  que estão no vetor  $i$  e das colunas que estão no vetor  $j$ .*
- 3 *Calcular a matriz  $Y$  que é igual a  $X$  exceto a linha  $k$ . A linha  $k$  de  $Y$  é a linha  $k$  de  $X$  menos a primeira linha de  $X$  multiplicada por um fator  $a$ .*



A rede da figura está composta por  $m = 8$  **arestas** e  $n = 7$  **nós**. Uma vez numerados os componentes, resulta um **grafo** cuja representação pode ser feita por uma **matriz de incidência**  $J$  ( $m \times n$ ). Notar que se introduz uma **orientação** (ou não).

$$J = \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix}$$



Outra representação é a **lista de arestas**  $C$ , e outra é a **matriz de adjacência**  $A$ , na qual  $A_{ij} = 1$  se  $i$  é conectado a  $j$  (em geral não identifica orientação, pode se assumir  $i$  conectado com  $i$  ou não).

$$C = \begin{pmatrix} 1 & 2 \\ 2 & 4 \\ 4 & 3 \\ 4 & 6 \\ 4 & 5 \\ 3 & 5 \\ 6 & 5 \\ 6 & 7 \end{pmatrix}, \quad A = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

## Exercício 6

*Realizar um código Octave que, dada a lista de arestas de um grafo, construa a matriz de incidência. E viceversa.*

## Exercício 7

*Realizar um código Octave que, dada a lista de arestas de um grafo, construa a matriz de adjacência. E viceversa.*

## Exercício 8

*Dada  $C$ , calcular o número de vizinhos de cada nó do grafo (sem considerar orientação das arestas).*

### **Mais informação:**

[strategic.mit.edu/downloads.php?page=matlab\\_networks](http://strategic.mit.edu/downloads.php?page=matlab_networks)

[github.com/aeolianine/octave-networks-toolbox/blob/master/functions\\_manual.pdf](https://github.com/aeolianine/octave-networks-toolbox/blob/master/functions_manual.pdf)