# Solução de sistema linear pelos métodos da Eliminação Gaussiana e decomposição LU
# +
# Obtenção da matriz inversa usando eliminação Gaussiana e LU

# Programa eli_gauss

```
function [A,b]=eli_gauss(A,b)  %entrada: matrizes A e
n=length(b);                   %saida: matrizes A e b
%ALGORITMO DA ELIMINAO GAUSSIANA
for k=1:n-1  %percorre todas as colunas da matriz A(i
    %TROCA DE LINHAS COM PIVOTAO PARCIAL, SE HOUVER Z
       if abs(A(k,k))<=  eps %Troca linhas, se A(j,j)
                 [amax,ind_i]=max(A(k+1:n,k))
          if(abs(amax) <= eps)
              error('Eliminao Gaussiana no pode pross
          end
              [A,b]=troca_linhas(A,b,k,ind_i+k,n);  %
      end
    for i=k+1:n  %(j fixo) percorre todas as linhas i
        m=A(i,k)/A(k,k);
        A(i,k)=0;  %zera coef da linha ij, j
        for j=k+1:n
            A(i,j)=A(i,j)-m*A(k,j);  %modifica os dem
        end
        b(i)= b(i)-m*b(k);  %modifica termo da linha
    end
end
```

# Programa troca_linhas

```
function [a,b]=troca_linhas(a,b,i,i_max,n)
    fprintf('k= %i. Trocou-se linha %i por linha %i',i,i,i_max)
at(1,:)=a(i,:); bt=b(i); %Uma linha de armazenamento temporario e

a(i,:)=a(i_max,:);  %troca linha i por linha i_max
a(i_max,:)=at(1,:);  %troca linha i_max por linha i

b(i)=b(i_max);
b(i_max)=bt;
```

# Programa retrosubstituição

```
function x=Resolve_sist_triangular_superior(U,b)
  n=length(b);
  x=zeros(n,1); %precisamos obrigar que vetor x s
  x(n)=b(n)/U(n,n);
  for i=n-1:-1:1
    x(i)=( b(i)-U(i,i+1:n)*x(i+1:n) )/U(i,i);
  end
end %fim da function
```

# Exemplo

```
clear all
A=[1 1 0 1;     2 1 -1 1;     -1 2 3 -1;      3 -1 -1 2];
b=[2; 1; 4; -3];

[A,b]=eli_gauss(A,b)

x=Resolve_sist_triangular_superior(A,b)
```

```
A =

     1     1     0     1
     0     1    -1     1
     0     0     3    -1
     0     0     0     2

b =

     2
    -3
    12
    -8

x =

    2.3333
    3.6667
    2.6667
   -4.0000
```

# Aplicação: Obtenção da matriz inversa usando eliminação Gaussiana

1. Sabendo que $AA^{-1} = I$, onde $A^{-1}$ é a matriz inversa da matriz $A$, e $I$ é a matriz identidade, e que $A$ e $I$ são matriz conhecidas, pode-se usar o método da eliminação gaussiana para achar as "n" colunas da matriz inversa $A^{-1}$.

   Por exemplo, para uma matriz $A_{3x3}$, temos que:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

   Assim, as 3 colunas incógnitas de $A^{-1}$ podem ser encontradas resolvendo 3 sistemas lineares $A\vec{x} = \vec{b}$, a saber:

1° sistema: $$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} \\ x_{21} \\ x_{31} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

2° sistema: $$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{12} \\ x_{22} \\ x_{32} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

## Sistemas lineares que precisam ser resolvidos

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$1^\circ \text{ sistema: } \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} \\ x_{21} \\ x_{31} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$2^\circ \text{ sistema: } \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{12} \\ x_{22} \\ x_{32} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$3^\circ \text{ sistema: } \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{13} \\ x_{23} \\ x_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Sabendo que $AA^{-1} = I$

**Montagem dos "n" sistemas lineares:**

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} \\ x_{21} \\ x_{31} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{12} \\ x_{22} \\ x_{32} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{13} \\ x_{23} \\ x_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Sabendo que $AA^{-1} = I$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} \\ x_{21} \\ x_{31} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{12} \\ x_{22} \\ x_{32} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{13} \\ x_{23} \\ x_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Sabendo que $AA^{-1} = I$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} \\ x_{21} \\ x_{31} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{12} \\ x_{22} \\ x_{32} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{13} \\ x_{23} \\ x_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Sabendo que $AA^{-1} = I$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} \\ x_{21} \\ x_{31} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{12} \\ x_{22} \\ x_{32} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{13} \\ x_{23} \\ x_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Sabendo que $AA^{-1} = I$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} \\ x_{21} \\ x_{31} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{12} \\ x_{22} \\ x_{32} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{13} \\ x_{23} \\ x_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

# Programa inversa_via_elim_gauss

```
function  [tempo_gasto,inversa,A]=inversa_via_elim_gauss(n)
A=rand(n);
tinicio = tic;
for i=1:n
    b=zeros(n,1);  b(i)=1; %b=coluna "j" da matriz identidade
    [A1,b1]=eli_gauss(A,b);
    inversa(:,i)=Resolve_sist_triangular_superior(A1,b1);
end
tempo_gasto = toc(tinicio);
```

# Matriz inversa usando decomposição LU

- Para a obtenção da matriz inversa, conforme mostrado anteriormente, é necessário resolver “n” sistemas lineares, nos quais somente  muda o vetor b (a matriz A para todos os sistemas é a mesma) (veja slide seguinte...)
- Nestes casos, é ,mais eficiente usar o método da decomposição LU.

## Passo 1: Decomposição de A em LU

Exemplo: Usando a eliminação Gaussiana, para $A_{3x3}$, temos:

$$A = LU$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \cdot \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

# Programa LUdecomp

```
function [a]=LUdecomp(a)  %entrada: matrizes A
n=length(a(:,1));                          %saida: matrizes L e
%ALGORITMO DA ELIMINAÇÃO GAUSSIANA
for k=1:n-1  %percorre todas as colunas da matriz A(i,j)
    for i=k+1:n  %(j fixo) percorre todas as linhas ij ab
        a(i,k)=a(i,k)/a(k,k); %gravamos a matriz L na pro
        for j=k+1:n
            a(i,j)=a(i,j)-a(i,k)*a(k,j);  %modifica os de
        end
    end
end
```

Note que os multiplicadores $m_{ij}$, da matriz L foram armazenados na própria matriz A, na posição i,j.

# Programa Resolve_sist_triangular_inferior_Ly_b

```
function y=Resolve_sist_triangular_inferior_Ly_b(L,b)
%Este algoritmo resolve o sistema Ly=b,
%A diagonal de L é formada por 1s, quando L é obtida por
%gaussiana

n=length(b);
y=zeros(n,1); %precisamos obrigar que vetor x seja tipo
y(1)=b(1);
for i=2:n
    y(i)= b(i)- L(i,1:i-1)*y(1:i-1) ;
end
```

# Programa Resolve_sist_triangular_superior

```
function x=Resolve_sist_triangular_superior(U,b) %retr
  n=length(b);
  x=zeros(n,1); %precisamos obrigar que vetor x seja t
  x(n)=b(n)/U(n,n);
  for i=n-1:-1:1
    x(i)=( b(i)-U(i,i+1:n)*x(i+1:n) )/U(i,i);
  end
end %fim da function
```

# Exemplo

Digitando os comandos abaixo no Matlab:

```
A=[8 4 -1; -2 5 1; 2 -1 6]
b=[11; 4; 7];

[A]=LUdecomp(A)
y=Resolve_sist_triangular_inferior_Ly_b(A,b)
x=Resolve_sist_triangular_superior(A,y)
```

Obtemos:

```
A =

    8.0000    4.0000   -1.0000
   -0.2500    6.0000    0.7500
    0.2500   -0.3333    6.5000
```

U

L

```
y =

   11.0000
    6.7500
    6.5000
```

```
x =

     1
     1
     1
```

# Aplicação: Calculo da Matriz inversa usando LU decomposição

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \cdot \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{11} \\ x_{21} \\ x_{31} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \cdot \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{12} \\ x_{22} \\ x_{32} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \cdot \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix} \cdot \begin{bmatrix} x_{13} \\ x_{23} \\ x_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

# Programa inversa_via_LU

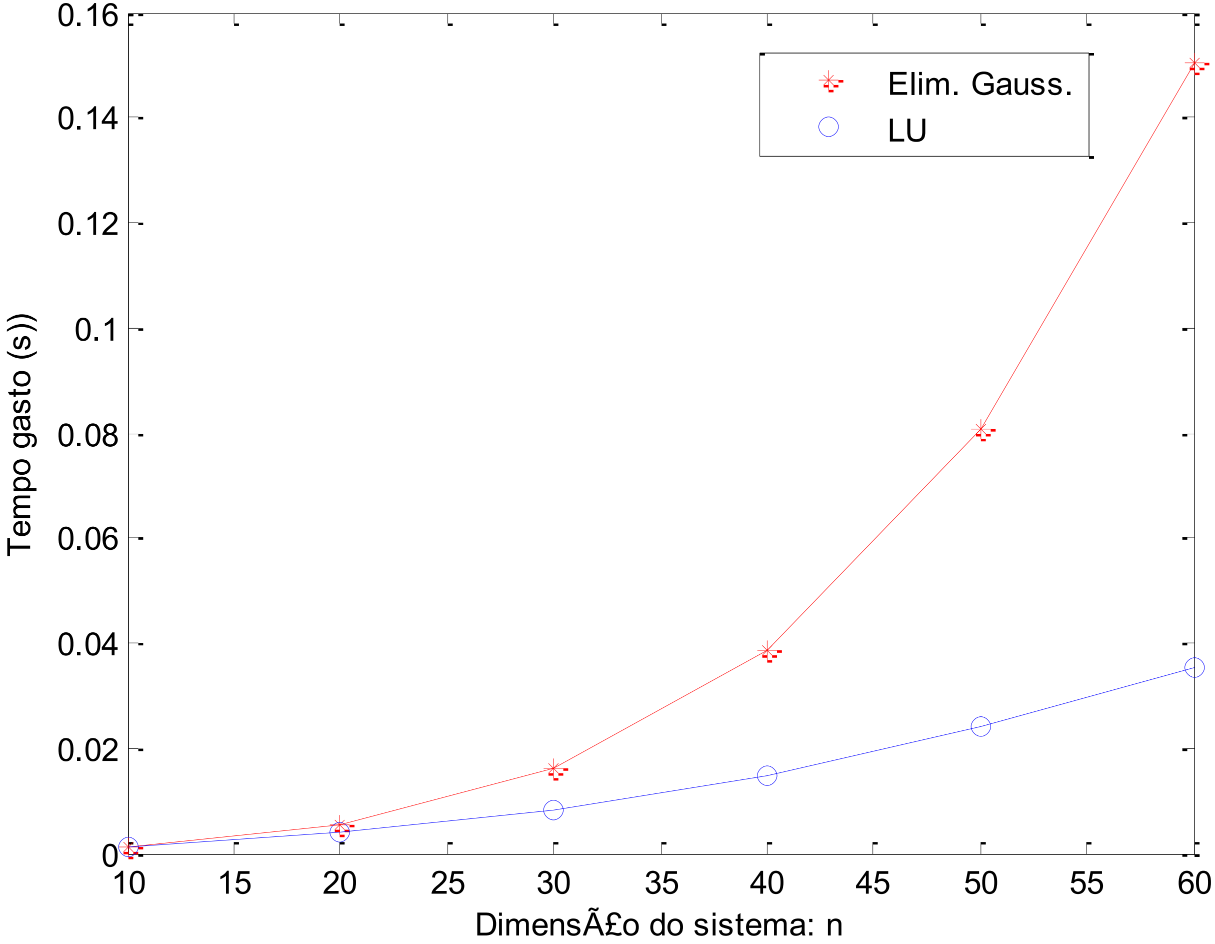
```
function  [tempo_gasto,inversa,A]=inversa_via_LU(n)
A=rand(n);  %A=[8 4 -1; -2 5 1; 2 -1 6] matriz para teste
A0=A;
tinicio = tic;
%Decompomos A=LU, mas armazenamos L e U na propria matriz A
[A]=LUdecomp(A);
for i=1:n
    b=zeros(n,1);  b(i)=1; %b=coluna "j" da matriz identidade
    %resolvemos sistema Ly=b
    y=Resolve_sist_triangular_inferior_Ly_b(A,b); %L está salva
    %resolvemos sistema Ux=y
%     x=Resolve_sist_triangular_superior(A,y)
    inversa(:,i)=Resolve_sist_triangular_superior(A,y);
end
%teste_ident=inversa*A0
tempo_gasto = toc(tinicio);
```

# Script para calcular e plotar os tempos tG e tLU

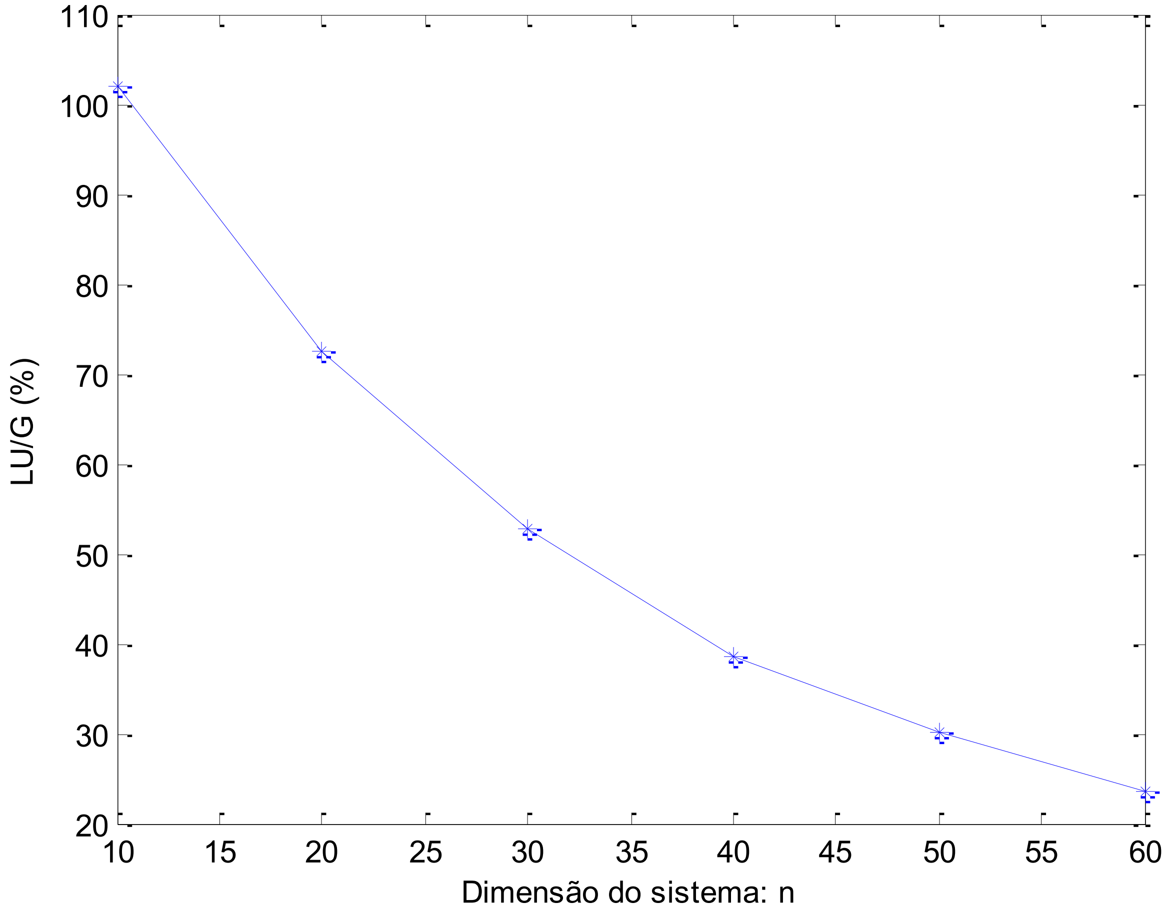
```
close
i=0;
for k=10:10:60
    i=i+1
     x(i)=k;
      tG(i)=inversa_via_elim_gauss(k);
       tLU(i)=inversa_via_LU(k);
end

plot(x,tG,'--r*', x,tLU,'--bo')
```

# Comparação do tempo de processamento (usando desktop)



# Comparação do tempo de processamento (%) (usando desktop)



# Conclusão

- Em situações onde é necessário resolver um conjunto de sistemas lineares, para os quais a matriz A permanece constante (apenas o vetor b muda), o método da decomposição LU é mais eficiente (menor tempo de processamento).

- Se for um único sistema, o método da eliminação gaussiana com retrosubstituição é mais adequado (levemente mais eficiente).

FIM !