AULA PRÁTICA 3
MÉTODOS ITERATIVOS PARA CÁLCULO DE AUTOVALORES E AUTOVETORES

1) MÉTODO DA POTÊNCIA - versão 1

Escreva uma função Scilab

function [lambda,x1,k,n_erro] = Metodo_potencia(A,x0,epsilon,M)

que implementa o Método da Potência para determinar o autovalor dominante (lambda) de A.

Variáveis de entrada:
A: matriz real n x n, diagonalizável, com autovalor dominante (lambda);
x0: vetor, não nulo, a ser utilizado como aproximação inicial do autovetor dominante.
epsilon: precisão a ser usada no critério de parada.
M: número máximo de iterações.

Variáveis de saída:
lambda: autovalor dominante de A;
x1: autovetor unitário (norma infinito) correspondente a lambda;
k: número de iterações necessárias para a convergência;
n_erro: norma infinito do erro

Critério de parada: sendo erro=x1 – x0 (diferença entre dois iterados consecutivos), parar quando n_erro < épsilon ou k>M.

ALGORITMO – versão 1

```
k=0
x0=x0/(coordenada de maior módulo de x0)
x1 = A*x0     (aproximação do autovetor dominante)
n_erro = épsilon + 1     (obriga a entrar no loop)
Enquanto k<=M e n_erro >= épsilon
        lambda = coord. de maior módulo de x1    (aproximação autovalor dominante)
        x1=x1/lambda
        n_erro = norma infinito de x1 – x0
        x0 = x1
        x1=A*x0
        k=k+1
Fim Enquanto
Mensagem e retorna
```

ALGORITMO – versão 2
k=0
x0=x0/(norma_2 de x0)
x1=A*x0 (aproximação do autovetor dominante)
n_erro = épsilon + 1 (Obriga a entrar no loop)
Enquanto k<=M e n_erro >= épsilon
    lambda = $x1^T$*x0 (Quociente de Rayleigh; x0 é unitário)
    Se lambda<0 então x1 = − x1 (Mantém x1 com mesmo sentido de x0)
    x1 = x1/norma_2 de x1
    n_erro = norma_2 de x1 – x0
    x0 = x1
    x1=A*x0
    k=k+1
Fim Enquanto
Mensagem e retorna

2) MÉTODO DA POTÊNCIA DESLOCADA com ITERAÇÃO INVERSA

Escreva uma função Scilab
function [lambda1,x1,k,n_erro] = Potencia_deslocada_inversa (A,x0,epsilon,alfa,M)
que implementa o Método da Potência Deslocada com Iteração Inversa para determinar o **autovalor de A mais próximo de "alfa".**

Variáveis de entrada:
A: matriz real n x n, diagonalizável;
x0: vetor, não nulo, a ser utilizado como aproximação inicial do autovetor dominante.
epsilon: precisão a ser usada no critério de parada.
alfa: valor do qual se deseja achar o autovalor de A mais próximo;
M: número máximo de iterações.

Variáveis de saída:
lambda1: autovalor de A mais próximo de alfa;
x1: autovetor unitário (norma_2) correspondente a lambda;
k: número de iterações necessárias para a convergência
n_erro: norma_2 do erro

Critério de parada: sendo erro = x1 – x0 (diferença entre dois iterados consecutivos), parar quando a n_erro < épsilon ou k>M.

```
ALGORITMO Potência Deslocada com Iteração Inversa
k=0
x0=x0/(norma_2 de x0)
n_erro = épsilon + 1      (Obriga a entrar no loop)
Enquanto k<=M e n_erro >= épsilon
        Resolva o sistema (A – alfa*I)*x1 = x0 para achar x1
        x1 = x1/(norma_2 de x1)
        lambda = x1^T*A*x1                (Quociente de Rayleigh; x1 é unitário)
        Se x1^T*x0 < 0 então x1 = − x1    (Mantém x1 com mesmo sentido de x0)
        n_erro = norma_2 de x1 – x0
        x0 = x1
        k=k+1
Fim Enquanto
lambda1 = ...
Mensagem e retorna
```

3) Teste suas duas primeiras funções para várias matrizes A, com ordens diferentes e também variando as demais variáveis de entrada de cada função. Use matrizes com autovalores reais (por exemplo, matrizes simétricas ou matrizes das quais você saiba os autovalores). Teste a mesma matriz com os dois primeiros algoritmos, comparando os números de iterações necessárias para convergência e os tempos de execução. Teste com uma matriz em que o autovalor dominante é negativo. Alguma coisa deu errada? Se for o caso, corrija o algoritmo (e a função) correspondente.
4) Construa uma matriz simétrica e use os Discos de Gerschgorin para estimar os autovalores. Use essas estimativas e o Método da Potência Deslocada com Iteração Inversa para calcular os autovalores.
5) Faça outros testes que achar convenientes ou interessantes!!! ☺