Teste 2 - Algoritmos Numéricos- DI

NOME: ____.

Leia **atentamente** as questões até o fim. Está sendo usada a notação americana, ou seja, usa-se o ponto (.) para denotar a parte fracionária.
Todas as soluções (menos os códigos) devem ser escritas pelo aluno (com sua letra) e, em seguida, fotografadas. Se possível, coloque as soluções em único arquivo pdf. Escreva de forma bem legível. Use uma caneta preta ou lápis bem escuro para o contraste.

1. Questão (2.0)
   O algoritmo abaixo (em pseudo-linguagem) obtém a solução de um sistema linear $Ax = b$, de ordem n, não singular (det A $\neq 0$), via eliminação de Gauss SEM pivoteamento.

```
INICIO
LER (A, b, n)
Para  k de 1 ate (n-1)  % fazendo as etapas
{
      Para  i de (k+1) ate n
      {
           m =  A(i,k)/A(k,k)
           MOSTRAR (m)
           Para  j de (k+1) ate  n
           {
              A(i,j) = A(i,j) - m* A(k,j)
           }
           b(i) = b(i)- m*b(k)
       } % Fim do para i
       MOSTRAR (A)
} % Fim do para k (fim da etapa k)

FIM
```

   Obs: o simbolo % refere se a um comentário.
   Usando este algoritmo obtenha o sistema triangular superior equivalente ao sistema abaixo,

$$\begin{cases} 0.003x_1 + 0.007x_2 - x_3 = 0.0667 \\ 30x_1 + 2x_2 + 240x_3 = 310 \\ 501x_1 + 0.1x_2 + 84.0x_3 = 503 \end{cases}$$

   simulando uma máquina que opere com aritmética de ponto flutuante normalizada com $t = 3$ dígitos significativos na mantissa, base 10 e expoente (valor inteiro) em $I = [-99, 99]$. *OBS: (1) Os arredondamentos devem ser feitos por corte (chopping) (por falta) (2) Não é necessário ficar escrevendo os valores numéricos obtidos usando potências de 10 (mas se preferir pode empregá-las).*
   Ao resolver, mostre os valores dos multiplicadores para cada linha, a cada etapa, isto é, o valor de "m" onde há o comando " MOSTRAR (m)". Exiba também a matriz A, que está armazenada na memória após cada etapa da triangularização, com este algoritmo, isto é, mostre a matriz A após o comando " MOSTRAR (A)".

2. Questão (2.0)

Considere o sistema linear $Ax = b$ dado abaixo:

$$\begin{cases} 50.0x_1 & + & 12.0x_2 & - & 15.5x_3 & & & = & 46.5 \\ -6x_1 & + & 50.0x_2 & + & 12.0x_3 & - & 15.5x_4 & = & 40.5 \\ 15.0x_1 & - & 6.0x_2 & + & 50.0x_3 & + & 12.0x_4 & = & 71.0 \\ & & 15.0x_2 & - & 6.0x_3 & + & 50.0x_4 & = & 59.0 \end{cases}$$

(a)(0.5) Escreva as expressões da atualização genérica de cada componente do sistema pelo método iterativo de Gauss Seidel (ou seja, como é gerado cada x_i do vetor da iteração $(k+1)$ dado o vetor da iteração anterior, para **este** problema).

(b) (1.5) Faça as duas primeiras iterações do método de Gauss Seidel, isto é, gere os vetores $x^{(1)}$ e o $x^{(2)}$. Use como vetor inicial (chute) o vetor $x^{(0)} = (0, 0, 0, 0)^T$. Use a memória de sua calculadora/computador e anote os valores com, no mínimo, 4 dígitos depois do ponto (ou vírgula), com o arredondamento que quiser.

3. Questão (2.0)
   (a)(1.0) Qual é o objetivo de se realizar trocas de linhas, em cada etapa da fase de triangularização na eliminação de Gauss?

   (b)(1.0) Qual é o objetivo de se realizar trocas de linhas antes de se iniciar o processo iterativo no método iterativo de Gauss Jacobi?

4. Questão (2.5)
   Matrizes pentadigonais são aquelas cujos elementos não nulos estão dispostos apenas em cinco diagonais (a principal mais outras quarto diagonais)

   Um caso específico de matrizes pentadigonais é quando estas cinco diagonais se concentram na faixa central da matriz. Dito de outra forma, estas matrizes pentadigonais são matrizes tais que os únicos elementos não nulos são:

$$a_{i,j} \neq 0 \;\text{ se }\; |i - j| \leq 2$$

   Um sistema linear pentadiagonal que aparece, algumas vezes, em aplicações numéricas, é aquele onde os elementos das diagonais, tanto na diagonal principal, como nas demais subdiagonais adjacentes (superiores e inferiores à diagonal principal) são constantes. Este caso é descrito por,

$$\begin{aligned} a_{i,j} &= d_A \;\text{ se }\; (i - j) = 2 \\ a_{i,j} &= d_B \;\text{ se }\; (i - j) = 1 \\ a_{i,j} &= D_P \;\text{ se }\; i = j \\ a_{i,j} &= d_C \;\text{ se }\; (i - j) = (-2) \\ a_{i,j} &= d_D \;\text{ se }\; (i - j) = (-1) \end{aligned}$$

   Um exemplo, de matriz pentadiagonal, de dimensão $n = 8$, que satisfaz às condições descritas acima, é mostrada abaixo. Neste caso, $d_A = 0.5$, $d_B = -2.0$, $D_P = 4.0$, $d_C = -2.4$ e $d_D = 0.8$.

$$A = \begin{bmatrix} 4.0 & -2.4 & 0.8 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ -2.0 & 4.0 & -2.4 & 0.8 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.5 & -2.0 & 4.0 & -2.4 & 0.8 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.5 & -2.0 & 4.0 & -2.4 & 0.8 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.5 & -2.0 & 4.0 & -2.4 & 0.8 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.5 & -2.0 & 4.0 & -2.4 & 0.8 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.5 & -2.0 & 4.0 & -2.4 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.5 & -2.0 & 4.0 \end{bmatrix}$$

A aplicação do método de eliminação de Gauss em sistemas lineares cuja estrutura da matriz é particular pode ser feita levando-se em consideração esta especificidade.

No caso pentadiagonal, à medida as etapas vão sendo realizadas, há posições, inicialmente nulas, que se tornam não nulas. Ou seja, à medida que se efetua a triangularização, algumas "novas diagonais" (adjacentes às já existentes) passam a ter elementos não nulos. Este preenchimento é conhecido como *fill-in*. É possível prever que posições serão preenchidas.

Um algoritmo bem projetado implicará em uma redução de operações em relação ao caso onde a matriz é cheia. Quando n é grande, a redução no número de operações, em relação ao caso geral (matriz cheia), é bastante significativa.

Escreva um código em octave para implementar o método de eliminação de Gauss para sistemas deste tipo, levando em considerção que se trata de uma matriz pentadiagonal, ou seja, deve ser construído para operar/acessar **apenas** os elementos que são **não nulos** ou que permanecem não nulos ao longo da triangularização.

Crie uma função chamada resolvePenta que tenha como dados de entrada a matriz A e o vetor b e retorne o vetor solução do sistema. Uma chamada seria:

```
>> [x]= resolvePenta(A,b)
```

Está disponível, o script (do tipo function) em octave "geraexemploPenta.m ", que gera um sistema linear, de dimensão n, do tipo pentadiagonal com valores específicos, isto é, onde a matriz tem elementos não nulos apenas 5 diagonais centrais. Neste caso, o código gera uma matriz Pentadiagonal com valores fixos para os elementos da diagonal principal e nas demais subdiagonais (veja os dados colocados no corpo do código). O código gera também um vetor b (ver detalhes no código).

Por exemplo, ao chamar a função, na janela de comandos, para $n = 6$, tal como abaixo:

```
>> [A,b]=geraexemploPenta(6)
```

A matriz é

```
  4.00000   12.00000  -15.50000    0.00000    0.00000    0.00000
 -6.00000    4.00000   12.00000  -15.50000    0.00000    0.00000
 15.00000   -6.00000    4.00000   12.00000  -15.50000    0.00000
  0.00000   15.00000   -6.00000    4.00000   12.00000  -15.50000
  0.00000    0.00000   15.00000   -6.00000    4.00000   12.00000
  0.00000    0.00000    0.00000   15.00000   -6.00000    4.00000
```

Rode o seu código para obter a solução de um sistema para dimensão n=10. Mostre a matriz após a triangularização e a solução.

5. Questão (1.5)
(a)(0.5) Dado um sistema linear $Ax = b$, um vetor chute inicial $x^{(0)}=(0, ..., 0)^T$, uma dada tolerância *tol* e um valor NumMax. Escreva um codigo para obter a solução de um sistema, com precisão *tol*, pelo método iterativo de Gauss Seidel. Considere a possibilidade de não haver convergência, sendo assim, nestas situações, o código deve parar quando fizer NumMax iterações (valor definido pelo usuário).
O critério de parada deve ser : $DifRel = \frac{||x^{k+1} - x^k||_{max}}{||x^{k+1}||_{max}} < tol$.

(b)(0.5) Está disponível um script chamado "geraexemploRANDDiagDom.m ", que gera um sistema linear, de dimensão n, com valores específicos (veja os detalhes no corpo do código).

Por exemplo, ao chamar a função, para $n = 3$, tal como abaixo:

```
>> [[A,b]=geraexemploRANDDiagDom(3)
```

A saida eh

```
A =
   1.38903   0.46781   0.74424
   0.64795   1.11086   0.18098
   0.84961   0.95718   2.41047

b =
   2.6011
   1.9398
   4.2173
```

Usando os códigos "GaussJacobi.m" (já disponibilizado) e o seu código implementado na letra (a) (para o Gauss Seidel), obtenha a solução de um sistema $Ax = b$, de dimensão $n = 10$, pelo dois métodos iterativos, com A e b gerados pela função "geraexemploRANDDiagDom.m".
Registre a quantidade de iterações que cada um realizou. Use tolerância $tol = 10^{-12}$ e um valor $NumMax = 500$

(c)(0.5) Rode 5 vezes, o mesmo experimento feito em (b), isto é, serão 5 sistemas $Ax = b$ (com A e b gerados pela função "geraexemploRANDDiagDom.m", com $n = 10$). Anote em uma tabela (ou em um gráfico) a quantidade de iterações que cada método "gastou" para atingir a solução com $tol = 10^{-12}$.