Gabriel Costa de Moraes

EXERCÍCIO 01 (ZERO DE FUNÇÕES): Um servidor Cloud executa tarefas críticas e seu consumo de energia deve ser monitorado continuamente. A taxa de consumo energético, $E(u)$, depende da taxa de utilização da CPU, t, que varia de 0 (ocioso) a 1 (uso máximo). O responsável pela gestão do servidor - usando dados armazenados - modelou empiricamente por interpolação polinomial que a função de consumo é como:

$$E(t) = 162,5t^3 - 243,75t^2 + 130t + 16,25$$

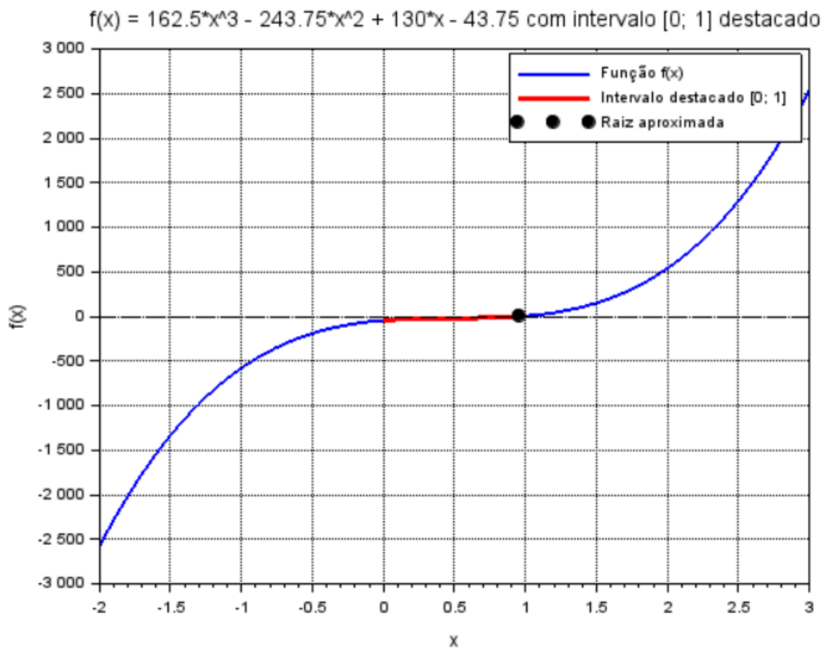
O engenheiro de computação do setor identificou que, um consumo máximo de $60w$ equilibra a eficiência e a temperatura do servidor em níveis seguros durante o período de pico. Então determine em que taxa de utilização da CPU o servidor atinge exatamente tal consumo.

Solução:

dada a função f(x)=162.5*x^3 - 243.75*x^2 + 130*x - 43.75
fazendo um estudo no intervalo [-4, 3] ⊂ ℝ

X	-4	-3	-2	-1	0	1	2	3
f(x)	-14863.75	-7015	-2578.75	-580	-43.75	5	541.25	

Como no subintervalo [0; 1] a função muda de sinal, possivelmente contém as raízes
Fazendo o gráfico da função no intervalo, temos que



O'Que corrobora que as eventuais raízes pertencem aos subintervalos indicados no item 1.
dada a derivada da função f(x) = f1(x) = 130 - 487.5*x + 487.5*x^2
Escolhendo o subintervalo [0, 1] para o estudo numérico, para mostrar a existência da raiz nesse intervalo deve-se mostrar que $\mathbf{f}(0) \cdot \mathbf{f}(1) < \mathbf{0}$, e para mostrar que é única neste subintervalo deve- se mostrar que $\mathbf{f1}(0) \cdot \mathbf{f1}(1) > \mathbf{0}$. Como $\mathbf{f}(1) = 5$ e $\mathbf{f}(0) = -43.75$ de modo que $\mathbf{f(0)} \cdot \mathbf{f}(1) = -43.75 \cdot (+5) = -218.75 < \mathbf{0}$. Analogamente, $f1(0) = 130$ e $\mathbf{f1}(1) = 130$, de modo que $\mathbf{f\ 1(0)} \cdot \mathbf{f\ 1(1)} = (130) \cdot (130) = 16900 > \mathbf{0}$. Assim existe uma única raiz no subintervalo escolhido.

Dessa forma Para o subintervalo [0, 1] obtém-se, via métodos de refinamento, que:

*** APROXIMAÇÃO PELO MÉTODO DA BISSEÇÃO ***

PROBLEMA 1

k	xm	abs(bk-ak)	f(xm)
1	0.500000	1.000000	-19.375000
2	0.750000	0.500000	-14.804688
3	0.875000	0.250000	-7.758789
4	0.937500	0.125000	-2.212524
5	0.968750	0.062500	1.170578
6	0.953125	0.031250	-0.574903
7	0.960938	0.015625	0.284122
8	0.957031	0.007812	-0.148790
9	0.958984	0.003906	0.066812
10	0.958008	0.001953	-0.041202
11	0.958496	0.000977	0.012752
12	0.958252	0.000488	-0.014238
13	0.958374	0.000244	-0.000746
14	0.958435	0.000122	0.006002
15	0.958405	0.000061	0.002627
16	0.958389	0.000031	0.000940
17	0.958382	0.000015	0.000097
18	0.958378	0.000008	-0.000325
19	0.958380	0.000004	-0.000114
20	0.958381	0.000002	-0.000008

Aproximadamente: 0.958381 é a raiz, com 20 iterações

*** APROXIMAÇÃO PELO MÉTODO DA FALSA-POSIÇÃO ***

PROBLEMA 1

k	xk	abs(b-a)	f(xk)
01	0.897436	1.000000	-5.944554
02	0.953144	0.102564	-0.572872
03	0.957960	0.046856	-0.046432
04	0.958347	0.042040	-0.003708
05	0.958378	0.041653	-0.000296
06	0.958381	0.041622	-0.000024
07	0.958381	0.041619	-0.000002
08	0.958381	0.041619	-0.000000

Aproximação "0.958381" à raiz, com "08" iterações

*** APROXIMAÇÃO PELO MÉTODO DA SECANTE ***

PROBLEMA 1

k	xk	abs(xk-x1)	f(xk)
01	0.953144	0.055708	-0.572872
02	0.959085	0.005941	0.077948
03	0.958373	0.000712	-0.000829
04	0.958381	0.000007	-0.000001
05	0.958381	0.000000	0.000000

Aproximação "0.958381" à raiz, com "05" iterações

*** APROXIMAÇÃO PELO MÉTODO DE NEWTON-RAPHSON ***

PROBLEMA 1

k	xk	abs(xk-x0)	f(xk)	f1(xk)
01	2.884615	2.384615	2203.476331	2780.240385
02	2.092067	0.792549	649.308582	1243.779471
03	1.570022	0.522045	188.400553	566.286439
04	1.237327	0.332695	51.753672	273.154877
05	1.047861	0.189466	11.798028	154.448700
06	0.971473	0.076388	1.486026	116.489601
07	0.958716	0.012757	0.037066	110.704849
08	0.958381	0.000335	0.000025	110.555157
09	0.958381	0.000000	0.000000	110.555056

Aproximação "0.958381" à raiz, com "09" iterações

EXERCÍCIO 02 (ZERO DE FUNÇÕES): Durante testes de escalabilidade de uma API web, foi observado que a latência média de resposta $L(u)$ (em milissegundos, ms), depende da carga simultânea de usuários u (em centenas). Uma modelagem do comportamento empírico (que também foi obtida usando os dados e interpolação polinomial) gerou a função:

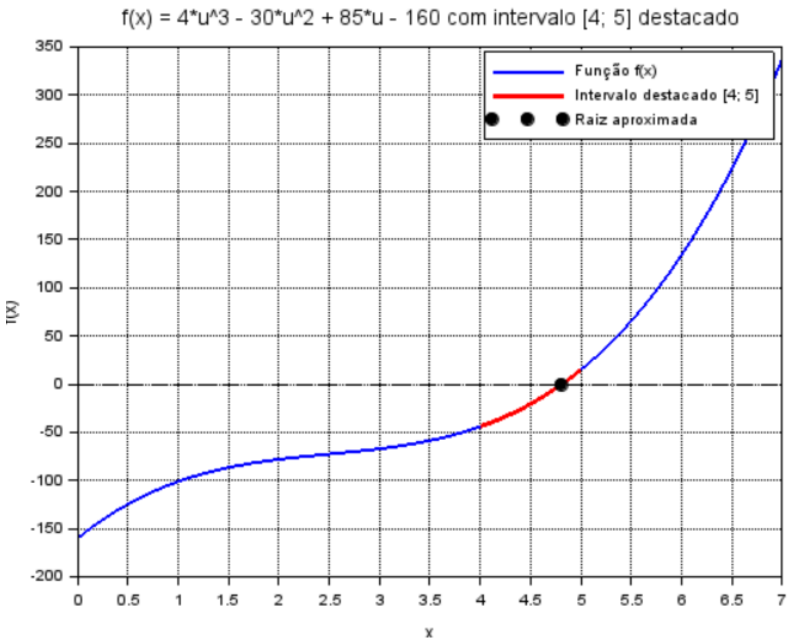
$$L(u) = 4c^3 - 30c^2 + 85c + 40$$

O time responsável pela rede estabeleceu que a latência aceitável não deve ultrapassar $200\ ms$ para manter a qualidade do serviço. Então determine a carga de usuários simultâneos (em centenas) em que a latência atinge esse limiar.

dada a função f(x)=4*u^3 - 30*u^2 + 85*u - 160
fazendo um estudo no intervalo [0, 7] ⊂ ℝ

X	0	1	2	3	4	5	6	7
f(x)	-160	-101	-78	-67	-44	15	134	337

Como no subintervalo [4; 5] a função muda de sinal, possivelmente contém as raízes
Fazendo o gráfico da função no intervalo, obtemos



O'Que corrobora que as eventuais raízes pertencem aos subintervalos indicados no item 1.
dada a derivada da função f(x) = f1(x) = 85 - 60*u + 12*u^2
Escolhendo o subintervalo [4, 5] para o estudo numérico, para mostrar a existência da raiz nesse intervalo deve-se mostrar que $\mathbf{f}(4) \cdot \mathbf{f}(5) < \mathbf{0}$, e para mostrar que é única neste subintervalo deve- se mostrar que $\mathbf{f1}(4) \cdot \mathbf{f1}(5) > \mathbf{0}$. Como $\mathbf{f}(4) = -44$ e $\mathbf{f}(5) = 15$ de modo que $\mathbf{f}(4) \cdot \mathbf{f}(5) = -44 \cdot (+15) = -660 < \mathbf{0}$. Analogamente, $\mathbf{f}1(4) = 37$ e $\mathbf{f1}(5) = 85$, de modo que $\mathbf{f\ 1}(4) \cdot \mathbf{f\ 1}(5) = (37) \cdot (45) = 3145 > \mathbf{0}$. Assim existe uma única raiz no subintervalo escolhido.

Dessa forma Para o subintervalo [4, 5] obtém-se, via métodos de refinamento, que:

```
*** APROXIMAÇÃO PELO MÉTODO DA FALSA-POSIÇÃO ***

PROBLEMA 2/n k  |      xk      | abs(b-a)    |     f(xk)     |
 01 |    4.745763 |   1.000000 |   -4.736804 |
 02 |    4.806779 |   0.254237 |   -0.332588 |
 03 |    4.810971 |   0.193221 |   -0.022557 |
 04 |    4.811254 |   0.189029 |   -0.001526 |
 05 |    4.811274 |   0.188746 |   -0.000103 |
 06 |    4.811275 |   0.188726 |   -0.000007 |
 07 |    4.811275 |   0.188725 |   -0.000000 |

Aproximação "4.811275" à raiz, com "07" iterações
```

```
*** APROXIMAÇÃO PELO MÉTODO DA BISSEÇÃO ***

PROBLEMA 2/n
  k |      xm      | abs(bk-ak) |   f(xm)      |
-----------------------------------------------
  1 |   4.500000 |   1.000000 | -20.500000   |
  2 |   4.750000 |   0.500000 |  -4.437500   |
  3 |   4.875000 |   0.250000 |   4.835938   |
  4 |   4.812500 |   0.125000 |   0.090820   |
  5 |   4.781250 |   0.062500 |  -2.200073   |
  6 |   4.796875 |   0.031250 |  -1.061356   |
  7 |   4.804688 |   0.015625 |  -0.486956   |
  8 |   4.808594 |   0.007812 |  -0.198490   |
  9 |   4.810547 |   0.003906 |  -0.053941   |
 10 |   4.811523 |   0.001953 |   0.018413   |
 11 |   4.811035 |   0.000977 |  -0.017770   |
 12 |   4.811279 |   0.000488 |   0.000320   |
 13 |   4.811157 |   0.000244 |  -0.008726   |
 14 |   4.811218 |   0.000122 |  -0.004203   |
 15 |   4.811249 |   0.000061 |  -0.001942   |
 16 |   4.811264 |   0.000031 |  -0.000811   |
 17 |   4.811272 |   0.000015 |  -0.000246   |
 18 |   4.811275 |   0.000008 |   0.000037   |
 19 |   4.811274 |   0.000004 |  -0.000104   |
 20 |   4.811275 |   0.000002 |  -0.000034   |
-----------------------------------------------
Aproximadamente: 4.811275 é a raiz, com 20 iterações
```

```
*** APROXIMAÇÃO PELO MÉTODO DA SECANTE ***

PROBLEMA 2 k  |      xk      | abs(xk-x1) |     f(xk)     |
 01 |    4.806779 |   0.061017 |   -0.332588 |
 02 |    4.811387 |   0.004608 |    0.008302 |
 03 |    4.811275 |   0.000112 |   -0.000014 |
 04 |    4.811275 |   0.000000 |   -0.000000 |

Aproximação "4.811275" à raiz, com "04" iterações
```

```
*** APROXIMAÇÃO PELO MÉTODO DE NEWTON-RAPHSON ***

PROBLEMA 2 k  |      xk      | abs(xk-x0) |     f(xk)     |    f1(xk)     |
 01 |    4.853448 |   0.353448 |   3.174835 |  76.464625 |
 02 |    4.811928 |   0.041520 |   0.048400 |  74.140131 |
 03 |    4.811275 |   0.000653 |   0.000012 |  74.103913 |
 04 |    4.811275 |   0.000000 |   0.000000 |  74.103904 |

Aproximação "4.811275" à raiz, com "04" iterações
```

EXERCÍCIO 03 (ZERO DE FUNÇÕES): Durante o projeto de cabos metálicos de alta precisão, que são utilizados em equipamento específico para a transmissão de sinais de alta frequência, é essencial garantir que certos fios suportem adequadamente a pressão mecânica resultante de vibrações termomecânicas. Um engenheiro de computação, em parceria com a equipe de hardware, modelou que a pressão máxima suportada, $\mathbf{p(d)}$, (em $\mathbf{kg/mm^2}$) em função do diâmetro, $\mathbf{d}$, do condutor metálico (em milímetros, $\mathbf{mm}$), é como na expressão empírica:

$$\mathbf{p(d) = 25d^2 + \ln(d)}$$

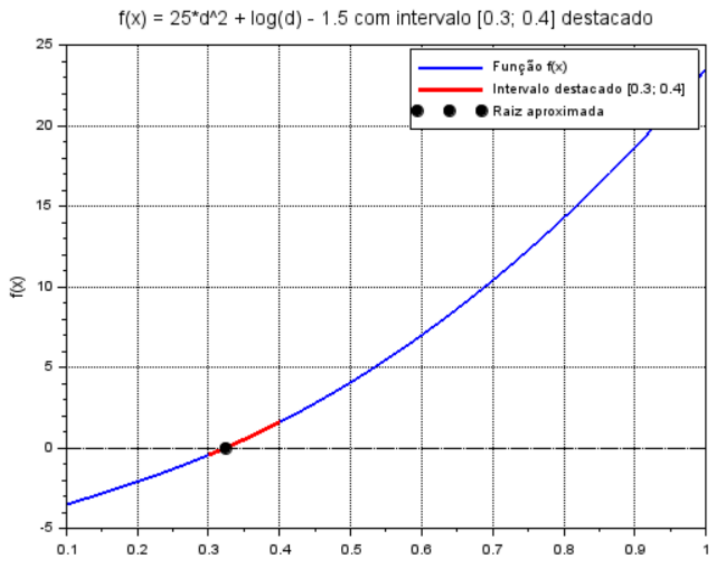
O diâmetro do cabo deve estar na faixa de valores $\mathbf{0,2 \leq d \leq 0,3}$ mm . Então determine uma aproximação para seu valor que garante suportar uma pressão de $\mathbf{p = 1,5\ kg/mm^2}$.

Solução:

dada a função f(d)= y = 25*d^2 + log(d) - 1.5
fazendo um estudo no intervalo [0.1; 1] ⊂ ℝ
temos que aumentando o intervalo dado no enunciado em **uma casa decimal** de 0,2 < d < 0,3 para o novo intervalo usado para 0,3 < d < 0,3

X	0.1	0.2	0.3	0.4	0.5	0.6	0.7	1
f(x)	−3.55258	−2.10943	−0.45397	1.58370	4.05685	6.98917	10.3933	23.5

Com o novo no subintervalo [0.3; 0.4] a função muda de sinal, possivelmente contém as raízes. Fazendo o gráfico da função no intervalo, obtemos



O'Que corrobora que as eventuais raízes pertencem aos subintervalos indicados no item 1.
dada a derivada da função f(x) = f1(x) = 1/d + 50*d
Escolhendo o subintervalo [0.3; 0.4] para o estudo numérico, para mostrar a existência da raiz, nesse intervalo deve-se mostrar que $\mathbf{f}(0.3) \cdot \mathbf{f}(0.4) < \mathbf{0}$, e para mostrar que é única neste subintervalo deve- se mostrar que $\mathbf{f1}(0.3) \cdot \mathbf{f1}(0.4) > \mathbf{0}$. Como $\mathbf{f}(0.3) = -0.45397$ e $\mathbf{f}(0.4) = 1.58370$ de modo que $\mathbf{f}(0.3) \cdot \mathbf{f}(0.4) = -0.45397 \cdot (+1.58370) = -0.71895 < \mathbf{0}$. Analogamente, $\mathbf{f1}(0.3) = 18.33333$ e $\mathbf{f1}(0.4) = 22.5$, de modo que $\mathbf{f\,1}(0.3) \cdot \mathbf{f\,1}(0.4) = (18.33333) \cdot (22.5) = 412.49992 > \mathbf{0}$. Assim existe uma única raiz no subintervalo escolhido.

Dessa forma Para o subintervalo [0.3, 0.4] obtém-se, via métodos de refinamento, que:

*** APROXIMAÇÃO PELO MÉTODO DA FALSA-POSIÇÃO ***

PROBLEMA 3

k	xk	abs(b-a)	f(xk)
01	0.322279	0.100000	-0.035746
02	0.323994	0.077721	-0.002720
03	0.324125	0.076006	-0.000206
04	0.324135	0.075875	-0.000016
05	0.324135	0.075865	-0.000001
06	0.324135	0.075865	-0.000000

Aproximação "0.324135" à raiz, com "06" iterações

*** APROXIMAÇÃO PELO MÉTODO DA SECANTE ***

PROBLEMA 3

k	xk	abs(xk-x1)	f(xk)
01	0.323994	0.001716	-0.002720
02	0.324136	0.000141	0.000005
03	0.324135	0.000000	-0.000000

Aproximação "0.324135" à raiz, com "03" iterações

*** APROXIMAÇÃO PELO MÉTODO DA BISSEÇÃO ***

PROBLEMA 3

k	xm	abs(bk-ak)	f(xm)
1	0.350000	0.100000	0.512678
2	0.325000	0.050000	0.016695
3	0.312500	0.025000	-0.221745
4	0.318750	0.012500	-0.103309
5	0.321875	0.006250	-0.043504
6	0.323437	0.003125	-0.013454
7	0.324219	0.001563	0.001608
8	0.323828	0.000781	-0.005926
9	0.324023	0.000391	-0.002160
10	0.324121	0.000195	-0.000276
11	0.324170	0.000098	0.000666
12	0.324146	0.000049	0.000195
13	0.324133	0.000024	-0.000041
14	0.324139	0.000012	0.000077
15	0.324136	0.000006	0.000018
16	0.324135	0.000003	-0.000011
17	0.324136	0.000002	0.000004

Aproximadamente: 0.324136 é a raiz, com 17 iterações

*** APROXIMAÇÃO PELO MÉTODO DE NEWTON-RAPHSON ***

PROBLEMA 3

k	xk	abs(xk-x0)	f(xk)	f1(xk)
01	0.324816	0.025184	0.013136	19.319459
02	0.324136	0.000680	0.000009	19.291920
03	0.324135	0.000000	0.000000	19.291900

Aproximação "0.324135" à raiz, com "03" iterações

EXERCÍCIO 05 (RESOLUÇÃO DE SISTEMAS LINEARES): **Você é responsável por abastecer uma loja de informática e precisa verificar o estoque dos quatro principais componentes vendidos: HDs (h), pendrives (p), SSDs (s) e memórias RAM (r). O almoxarife, que era aluno do Curso de Matemática, deixou indicações como deve ser abastecido o estoque. Ele indicou que:**

- **Três vezes o número de HDs, mais duas vezes o número de pendrives, menos o número de SSDs, mais o número de memórias RAM resulta em 9 componentes.**
- **Duas vezes o número de HDs, menos duas vezes o número de pendrives, mais quatro vezes o número de SSDs, menos três vezes o número de memórias RAM resulta em 11 componentes.**
- **O número de HDs, mais o número de pendrives, mais o número de SSDs, menos o número de memórias RAM totaliza 8 unidades.**
- **Duas vezes o número de HDs, mais três vezes o número de pendrives, mais o número de SSDs, mais quatro vezes o número de memórias RAM somam 21 componentes.**

SOLUÇÃO:

Queremos determinar quantos componentes de cada tipo (HDs, pendrives, SSDs, memórias RAM) devem ser estocados, de forma que sejam satisfeitas as condições do sistema do enunciado:

temos o sistema:

$3h + 2p - s + r = 9$

$2h - 2p + 4s - 3r = 11$

$h + p + s - r = 8$

$2h + 3p + s + 4r = 21$

Resolvendo por Eliminação Gaussiana sem pivoteamento, temos os seguintes resultados:

```
PROBLEMA 1
*** MÉTODO DIRETO: GAUSS (ELIMINAÇÃO GAUSSIANA) SEM PIVOTEAMENTO ***

Entrada - Matriz A (original):

   3.   2.  -1.   1.
   2.  -2.   4.  -3.
   1.   1.   1.  -1.
   2.   3.   1.   4.
Entrada - Vetor B (original):

   9.
   11.
   8.
   21.


---------------------------
****Dimensão de n: 4 variáveis
****-----------------------------
****Matriz A triangularizada:****
   3.   2.          -1.          1.
   0.  -3.3333333   4.6666667  -3.6666667
   0.   0.          1.8        -1.7
   0.   0.          0.          5.2777778
****Vetor B escalonado:****
   9.
   5.
   5.5
   5.2777778

Solução X do Sistema:
 x(1)  = 2.000000
 x(2)  = 3.000000
 x(3)  = 4.000000
 x(4)  = 1.000000
```

```
Verificação dos resultados (AX = B):
 (3*2.000000) +  (2*3.000000) +  (-1*4.000000) +  (1*1.000000) =  9.000000
 (2*2.000000) +  (-2*3.000000) +  (4*4.000000) +  (-3*1.000000) =  11.000000
 (1*2.000000) +  (1*3.000000) +  (1*4.000000) +  (-1*1.000000) =  8.000000
 (2*2.000000) +  (3*3.000000) +  (1*4.000000) +  (4*1.000000) =  21.000000

Erro absoluto (AX - B):

   0.
   0.
   0.
   0.

 ********** ELIMINAÇÃO GAUSSIANA FINALIZADA **********
```

Resolvendo por Fatoração LU , temos os seguintes resultados:

```
PROBLEMA 1
***** MÉTODO DIRETO: FATORAÇÃO LU por CROUT *****
Entrada - Matriz A (original):
   3.   2.  -1.   1.
   2.  -2.   4.  -3.
   1.   1.   1.  -1.
   2.   3.   1.   4.

Entrada - Vetor B (original):
   9.
   11.
   8.
   21.

---------------------------
****Dimensão de n: 4 variáveis****
-----------------------------
*****FATOR L:*****
   3.   0.          0.    0.
   2.  -3.3333333   0.    0.
   1.   0.3333333   1.8   0.
   2.   1.6666667   4.    5.2777778

*****FATOR U:*****
   1.   0.6666667  -0.3333333   0.3333333
   0.   1.         -1.4         1.1
   0.   0.          1.         -0.9444444
   0.   0.          0.          1.

Solução Y de LY=B:
   3.
  -1.5000000
   3.0555556
   1.

Solução X (UX = Y):
 x(1) = 2.000000
 x(2) = 3.000000
 x(3) = 4.000000
 x(4) = 1.000000

Verificação dos resultados (A*X ≈ B):
(3*2.000000) + (2*3.000000) + (-1*4.000000) + (1*1.000000) = 9.000000
(2*2.000000) + (-2*3.000000) + (4*4.000000) + (-3*1.000000) = 11.000000
(1*2.000000) + (1*3.000000) + (1*4.000000) + (-1*1.000000) = 8.000000
(2*2.000000) + (3*3.000000) + (1*4.000000) + (4*1.000000) = 21.000000

Erro absoluto (AX - B):

   0.
   0.
   0.
   0.

 ********** FATORAÇÃO LU FINALIZADA **********
```

Resolvendo com os métodos **Gauss-Jacobi e Gauss-Seidel**
A = [3, 2, -1, 1;   2, -2, 4, -3;   1, 1, 1, -1;   2, 3, 1, 4];
B = [9; 11; 8; 21];

não podem ser aplicados, pois não existem permutações que transformem a matriz em diagonal dominante, para aplicar o Critério das Linhas.

Resolvendo com TDMA

O algoritmo TDMA (Thomas Algorithm) **não pode ser aplicado**, pois a matriz de coeficientes do sistema não é tridiagonal.

EXERCÍCIO 06 (RESOLUÇÃO DE SISTEMAS LINEARES): Um problema de condução de calor (muito simplificado mas utilizado em alguns equipamentos de informática) é o definido em uma barra unidimensional (1D) isolada termicamente nas laterais, mas não nas extremidades. A equação diferencial para condução de calor 1D (temperatura $\mathbf{T}\ {}^{\mathbf{o}}\mathbf{C}$) em estado estacionário com sua respectiva discretização em diferenças finitas são, respectivamente, como:

$$\frac{d^2T}{dx^2} = 0 \Rightarrow T_{i-1} - 2\,T_i + T_{i+1} = 0$$

Considere que as temperatura nas extremidades da barra são conhecidas, sendo $b_0 = 100$ e $b_6 = 200$. As temperaturas em todos os elementos discretizado com $n = 5$ pontos são obtidas com a solução de um sistema tridiagonal. Resolva o sistema para mostrar as temperaturas em cada ponto. Inicie a solução com $i = 1$ e considere que $T_0 = 0 = T_6$.
Solução:

precisamos determinar as temperaturas T_1, T_2, T_3, T_4, T_5 em cinco pontos internos de uma barra unidimensional, onde a equação de condução de calor em estado estacionário é dada por T(i−1) −2Ti +T (i+1) =0
então complementando a matriz temos:
T_1, T_2, T_3, T_4, T_5
T_1, T_2, T_3, T_4, T_5
T_1, T_2, T_3, T_4, T_5
T_1, T_2, T_3, T_4, T_5
T_1, T_2, T_3, T_4, T_5

-2T1 + T2 = -100
T1 - 2T2 + t3 = 0
T2 - 2T3 + T4 = 0
T3 - 2T4 + T5 = 0
T4 -2T5 = -200

A = [-2, 1, 0, 0, 0;
  1, -2, 1, 0, 0;
  0, 1, -2, 1, 0;
  0, 0, 1, -2, 1;
  0, 0, 0, 1, -2];

B = [-100; 0; 0; 0; -200];

Resolvendo por Eliminação Gaussiana sem pivoteamento

```
PROBLEMA 2
*** MÉTODO DIRETO: GAUSS (ELIMINAÇÃO GAUSSIANA) SEM PIVOTEAMENTO ***

Entrada - Matriz A (original):

  -2.   1.   0.   0.   0.
   1.  -2.   1.   0.   0.
   0.   1.  -2.   1.   0.
   0.   0.   1.  -2.   1.
   0.   0.   0.   1.  -2.
Entrada - Vetor B (original):

  -100.
   0.
   0.
   0.
  -200.

---------------------------
****Dimensão de n: 5 variáveis
****-----------------------------
****Matriz A triangularizada:****
  -2.   1.    0.          0.     0.
   0.  -1.5   1.          0.     0.
   0.   0.   -1.3333333   1.     0.
   0.   0.    0.         -1.25   1.
   0.   0.    0.          0.    -1.2
****Vetor B escalonado:****
  -100.
  -50.
  -33.333333
  -25.000000
  -220.
```

```
Solução X do Sistema:
 x(1) = 116.666667
 x(2) = 133.333333
 x(3) = 150.000000
 x(4) = 166.666667
 x(5) = 183.333333

Verificação dos resultados (AX = B):
 (-2*116.666667) +  (1*133.333333) +  (0*150.000000) +  (0*166.666667) +  (0*183.333333) =  -100.000000
 (1*116.666667) +  (-2*133.333333) +  (1*150.000000) +  (0*166.666667) +  (0*183.333333) =  0.000000
 (0*116.666667) +  (1*133.333333) +  (-2*150.000000) +  (1*166.666667) +  (0*183.333333) =  0.000000
 (0*116.666667) +  (0*133.333333) +  (1*150.000000) +  (-2*166.666667) +  (1*183.333333) =  -0.000000
 (0*116.666667) +  (0*133.333333) +  (0*150.000000) +  (1*166.666667) +  (-2*183.333333) =  -200.000000

Erro absoluto (AX - B):

   0.
   0.
   5.684D-14
  -5.684D-14
   0.

 ********** ELIMINAÇÃO GAUSSIANA FINALIZADA **********
```

Resolvendo por Fatoração LU , temos os seguintes resultados:

```
PROBLEMA 2
***** MÉTODO DIRETO: FATORAÇÃO LU por CROUT *****
Entrada - Matriz A (original):
  -2.   1.   0.   0.   0.
   1.  -2.   1.   0.   0.
   0.   1.  -2.   1.   0.
   0.   0.   1.  -2.   1.
   0.   0.   0.   1.  -2.

Entrada - Vetor B (original):
  -100.
   0.
   0.
   0.
  -200.

---------------------------
****Dimensão de n: 5 variáveis****
-----------------------------
*****FATOR L:*****
  -2.   0.    0.           0.      0.
   1.  -1.5   0.           0.      0.
   0.   1.   -1.3333333    0.      0.
   0.   0.    1.          -1.25    0.
   0.   0.    0.           1.     -1.2

*****FATOR U:*****
   1.  -0.5   0.           0.      0.
   0.   1.   -0.6666667    0.      0.
   0.   0.    1.          -0.75    0.
   0.   0.    0.           1.     -0.8
   0.   0.    0.           0.      1.

Solução Y de LY=B:
   50.
   33.333333
   25.
   20.
   183.33333
```

```
Solução X (UX = Y):
 x(1) = 116.666667
 x(2) = 133.333333
 x(3) = 150.000000
 x(4) = 166.666667
 x(5) = 183.333333

Verificação dos resultados (A*X ≈ B):
(-2*116.666667) + (1*133.333333) + (0*150.000000) + (0*166.666667) + (0*183.333333) = -100.000000
(1*116.666667) + (-2*133.333333) + (1*150.000000) + (0*166.666667) + (0*183.333333) = 0.000000
(0*116.666667) + (1*133.333333) + (-2*150.000000) + (1*166.666667) + (0*183.333333) = 0.000000
(0*116.666667) + (0*133.333333) + (1*150.000000) + (-2*166.666667) + (1*183.333333) = -0.000000
(0*116.666667) + (0*133.333333) + (0*150.000000) + (1*166.666667) + (-2*183.333333) = -200.000000

Erro absoluto (AX - B):

   0.
   0.
   2.842D-14
  -2.842D-14
   0.

 ********** FATORAÇÃO LU FINALIZADA **********
```

gauss_jacobi_guloso não pode ser aplicado pois o sistema não é diagonal dominante

gauss_seidel_guloso não pode ser aplicado pois o sistema não é diagonal dominante

TDMA

```
PROBLEMA 2
*** MÉTODO DIRETO: THOMAS (TDMA) - SISTEMAS TRIDIAGONAIS ***

 Vetor a^T:
   0.   1.   1.   1.   1.

 Vetor b^T:
  -2.  -2.  -2.  -2.  -2.

 Vetor c^T:
   1.   1.   1.   1.   0.

 Vetor d^T:
  -100.   0.   0.   0.  -200.

O sistema possui 5 variáveis (dimensão da raiz x).

 Solução X do sistema:
  x(1) = 116.666667
  x(2) = 133.333333
  x(3) = 150.000000
  x(4) = 166.666667
  x(5) = 183.333333

Verificação dos resultados (A*X ≈ d):
 (-2.0*116.666667) + (1.0*133.333333) + (0.0*150.000000) + (0.0*166.666667) + (0.0*183.333333) = -100.000000
 (1.0*116.666667) + (-2.0*133.333333) + (1.0*150.000000) + (0.0*166.666667) + (0.0*183.333333) = 0.000000
 (0.0*116.666667) + (1.0*133.333333) + (-2.0*150.000000) + (1.0*166.666667) + (0.0*183.333333) = 0.000000
 (0.0*116.666667) + (0.0*133.333333) + (1.0*150.000000) + (-2.0*166.666667) + (1.0*183.333333) = -0.000000
 (0.0*116.666667) + (0.0*133.333333) + (0.0*150.000000) + (1.0*166.666667) + (-2.0*183.333333) = -200.000000

 Erro absoluto (A*X - d):

   0.
   0.
   2.842D-14
  -2.842D-14
   0.
```