

# **TAREFA - APLICAÇÕES COM PYTHON**

## **Prof. Methodio Godoy**

### **BIBLIOTECAS NO PYTHON**

A linguagem Python possui um vasto e diversificado conjunto de bibliotecas que expandem suas capacidades para diversas áreas de aplicação, como engenharia, matemática, estatística, ciência de dados, aprendizado de máquina, desenvolvimento web, automação, processamento de texto, entre outras.

A maioria das bibliotecas de Python pode ser instalada usando o pip, que é o gerenciador de pacotes padrão para Python. Por exemplo, para instalar a biblioteca numpy, você pode usar o seguinte comando no terminal ou prompt de comando:

**>pip install numpy**

Se estiver usando um ambiente de desenvolvimento integrado (IDE) como o Jupyter Notebook, você pode executar o comando diretamente em uma célula do notebook, precedido por !:

**>!pip install numpy**

Depois de instalar a biblioteca, você precisa importá-la em seu script ou ambiente de desenvolvimento. A importação pode ser feita usando o comando import.

**>import numpy as np**

**>import pandas as pd**

**>import matplotlib.pyplot as plt**

Assim importamos três bibliotecas mais usadas: numpy, pandas, e matplotlib. Também podemos usar alias (as np, as pd, as plt) para tornar a utilização dessas bibliotecas mais conveniente.

**>import numpy as np**

A biblioteca NumPy é fundamental para a computação científica com Python. Ela suporta arrays e matrizes multidimensionais, junto com uma coleção de funções matemáticas para operar com esses arrays. Após a importação, pode-se utilizar as funções e classes fornecidas pela biblioteca. A Figura 1 apresenta exemplo usando a biblioteca **numpy** para operações numéricas.

```
# Bibliotecas Básicas
import numpy as np

# Criação de um array
array = np.array([1, 2, 3, 4, 5])

# Operações com arrays
# Soma de todos os elementos: 15
print('\n Exemplos de Uso da Biblioteca numpy \n')
soma = np.sum(array)
print('Soma elementos da matriz %5.2f\n'%soma)
# Média dos elementos: 3.0
media = np.mean(array)
print('Média elementos da matriz %5.2f\n'%media)
```

Figura 1 - Uso da biblioteca numpy

```
Exemplos de Uso da Biblioteca numpy

Soma elementos da matriz 15.00

Média elementos da matriz  3.00
```

Figura 2 - Resultados do programa da Figura 1

A biblioteca matplotlib é adotada para criar gráficos de alta qualidade. É amplamente utilizada para visualização de dados. Na Figura 3 um exemplo de uso da biblioteca matplotlib para criar um gráfico (Figura 4).

```
# Bibliotecas Básicas

import matplotlib.pyplot as plt
import numpy as np

# Dados para o gráfico
x = np.linspace(0, 10, 100)
y = np.sin(x)

# Criação do gráfico
plt.plot(x, y)
plt.xlabel('x')
plt.ylabel('sin(x)')
plt.title('Uso da Biblioteca Matplotlib')
plt.show()
```

Figura 3 - Uso da biblioteca numpy e matplotlib

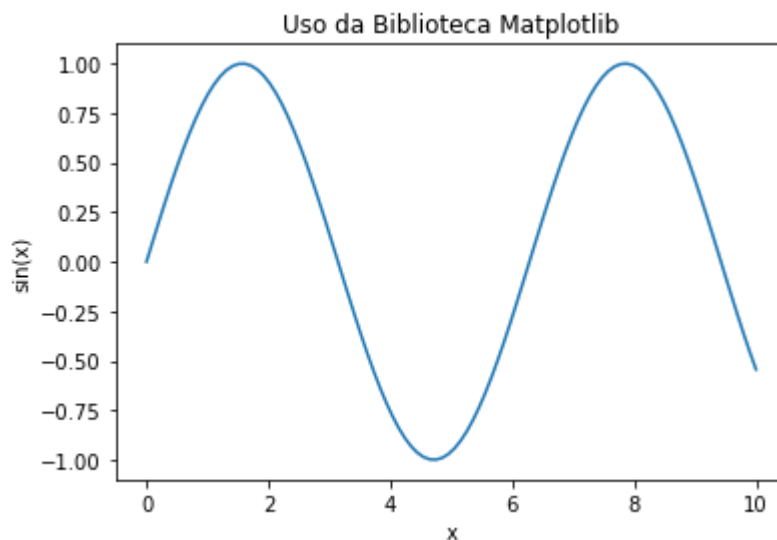


Figura 4 - Resultados do programa da Figura 3

A biblioteca **math** possui as funções matemáticas básicas para realizar operações comuns de matemática, incluindo funções trigonométricas, logarítmicas, exponenciais, entre outras. A **math** é uma biblioteca padrão, o que significa que ela vem incluída na instalação do Python, então você não precisa instalá-la separadamente. A biblioteca **cmath** possui as funções matemáticas básicas para realizar operações com número complexo.

A biblioteca **pandas** têm estruturas de dados e funções de análise de dados de alto desempenho e fáceis de usar. É essencial para manipulação e análise de dados, especialmente em tabelas de dados. Uma das estruturas de dados é a **Series** que é uma matriz unidimensional capaz de armazenar qualquer tipo de dados (inteiros, strings, floats, objetos Python, etc.). Outra estrutura de dados é o **DataFrame** que é uma estrutura de dados tabular bidimensional com colunas de tipos potencialmente diferentes. É similar a uma tabela em um banco de dados ou a uma planilha do Excel.

A linguagem Python é amplamente utilizada em sistemas elétricos de potência devido à sua flexibilidade, facilidade de uso e a disponibilidade de várias bibliotecas específicas para análise, simulação e modelagem de sistemas de energia. A biblioteca PyPSA (Python for Power System Analysis) foi desenvolvida para simulação e otimização de sistemas elétricos de potência, permitindo a análise de fluxos de potência, fluxos de potência ótimos, simulação de redes elétricas CA e CC.

**O objetivo desta Tarefa é fazer você desenvolver a Biblioteca em Python asp1 para te auxiliar no seu dia a dia. Cada questão deve ser feita desenvolvendo funções e usando-as para resolver os problemas. Para ilustrar tal desenvolvimento vamos apresentar a biblioteca methodio e algumas de suas funções.**

Na Figura 5 apresentamos a biblioteca methodio e destacamos algumas das funções para operar com números complexos e na Figura 6 um programa que usa esta biblioteca.

```
# Biblioteca methodio

import numpy as np
import matplotlib.pyplot as plt
from datetime import datetime

# Biblioteca de Números Complexos

def n_polar(F):
    modu=np.abs(F)
    ang=np.angle(F)
    angg=np.angle(F,deg='true')
    R=(modu,ang,angg)
    return R

def c_polar(m1,f1):
    # A fase do número complexo é em graus
    A = m1*np.cos(f1*np.pi/180.0)+1j*m1*np.sin(f1*np.pi/180.0)
    return A

def soma_polar(m1,f1,m2,f2):
    soma=c_polar(m1,f1)+c_polar(m2,f2)
    return soma

def subt_polar(m1,f1,m2,f2):
    subt=c_polar(m1,f1)-c_polar(m2,f2)
    return subt

def mult_polar(m1,f1,m2,f2):
    mult=c_polar(m1,f1)*c_polar(m2,f2)
    return mult

def div_polar(m1,f1,m2,f2):
    div=c_polar(m1,f1)/c_polar(m2,f2)
    return div
```

Figura 5 - Trecho da biblioteca methodio

```
# Bibliotecas Methodio
import numpy as np
import methodio as meth

print('\n Exemplos de Uso da Biblioteca methodio  \n')
R=3+2j+meth.c_polar(1,45)+6-4j
print('          Soma dos Complexos  \n')
print(' Módulo : %9.3f      Fase (gr) : %9.3f'%(abs(R),np.angle(R)))
print(' P Real : %9.3f      P Imag. : %9.3f'%(np.real(R),np.imag(R)))
```

Figura 6 - Exemplo de programa usando função da biblioteca

```
Exemplos de Uso da Biblioteca methodio

Soma dos Complexos

Módulo :      9.793      Fase (gr) :      -0.132
P Real  :      9.707      P Imag.   :      -1.293
```

Figura 7 - Resultados do programa da Figura 6

1. Desenvolva uma função **imp\_serie(Z)** que calcule a impedância equivalente ZEQ das n impedâncias em série dadas no vetor Z para a biblioteca com seu nome. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.
2. Desenvolva uma função **imp\_par(Z)** que calcule a impedância equivalente ZEQ das n impedâncias em paralelo dadas no vetor Z para a biblioteca com seu nome. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.
3. Desenvolva uma função **pot\_comp1f(V,I)** que calcule a potência complexa S para a biblioteca com seu nome. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.
4. Desenvolva uma função **pot\_comp3f(Va,la,Vb,lb,Vc,lc)** que calcule a potência complexa S para a biblioteca com seu nome. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.
5. Desenvolva uma função **cor\_carga(V,N,FP)** que calcule a corrente de carga I para a biblioteca com seu nome. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.
6. Desenvolva uma função **oper\_comp(z1,form1,op,z2,form2)** para a biblioteca com seu nome, onde z1 é o número complexo 1, form1 é o formato de z1 (se é retangular ou polar), op é a operação que pode ser soma, subtração, multiplicação e divisão, z2 é o número complexo 2, form2 é o formato de z2. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.
7. Desenvolva uma função **Qcor\_pot(P, FPA, FPD)** que calcule a potência reativa mínima para corrigir o fator de potência e uma carga de potência ativa P do fator de

potência antigo (FPA) para o fator de potência novo (FPN) para a biblioteca com seu nome. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.

8. Desenvolva uma função **cte\_gener(Zpi,Ya,Yb)** que calcule  $R=(A,B,C,D)$  onde A, B, C e D são as constantes generalizadas da matriz de transmissão de quadripolo obtido a partir do circuito PI para a biblioteca com seu nome. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.(Figura 8)

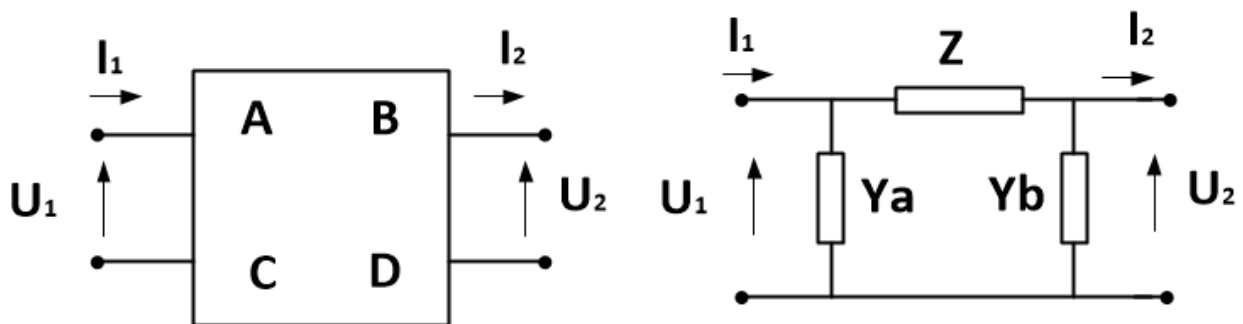


Figura 8 - Quadripolo e circuito PI

9. Desenvolva uma função **quad\_casc(A1,B1,C1,D1,A2,B2,C2,D2)** que calcule  $R=(Aeq,Beq,Ceq,Deq)$  com os parâmetros do quadripolo equivalente da cascata de dois quadripolos cujos parâmetros são A1,B1,C1,D1 e A2,B2,C2, D2 para a biblioteca com seu nome. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**. (Figura 9)

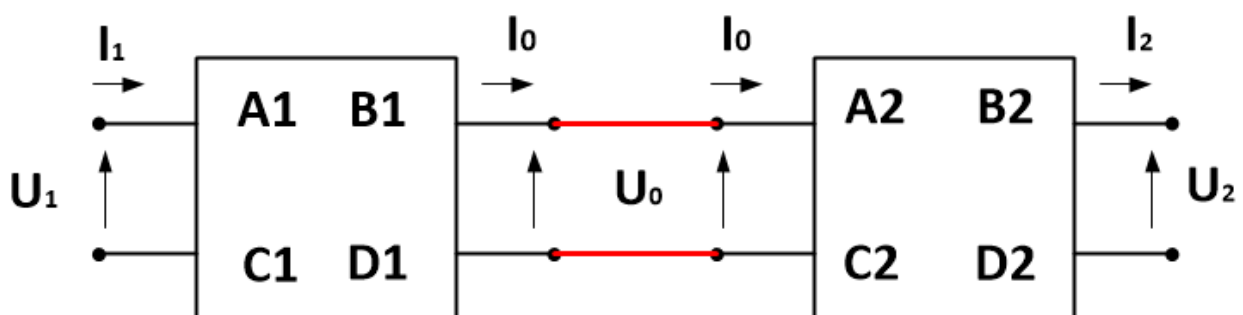


Figura 6 - Quadripolo em cascata

10. Desenvolva uma função **quad\_par(A1,B1,C1,D1,A2,B2,C2,D2)** que calcule  $R=(A,B,C,D)$  com os parâmetros do quadripolo equivalente em paralelo de dois

quadripolos cujos parâmetros são  $A_1, B_1, C_1, D_1$  e  $A_2, B_2, C_2, D_2$  para a biblioteca com seu nome. Apresente um exemplo e envie o arquivo teste e a biblioteca asp1.

11. Desenvolver uma função **cirpi** (**V2, I2, Z, Ya, Yb, nomearq**) com Python onde o vetor resposta  $R = (V_1, I_1)$  são a tensão  $V_1$  e a corrente  $I_1$  no terminal transmissor de um circuito pi conhecendo a tensão e a corrente no terminal receptor e a impedância  $Z$  e as admitâncias  $Y_a$  e  $Y_b$ . A variável nomearq é o nome do arquivo tipo texto que deve ser gerado onde devem ser apresentados todos os dados de entrada e saída, para a biblioteca com seu nome. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.

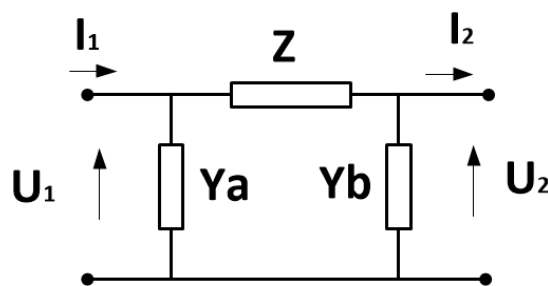


Figura 7 - Circuito pi

12. Desenvolver uma função **cirpir** (**V1, I1, Z, Ya, Yb, nomearq**) com Python onde são obtidas a tensão  $V_2$  e a corrente  $I_2$  no terminal receptor de um circuito pi conhecendo a tensão e a corrente no terminal transmissor e a impedância  $Z$  e as admitâncias  $Y_a$  e  $Y_b$ . O resultado  $R=(V_2, I_2)$  apresenta a tensão e a corrente obtidas no terminal receptor. A variável nomearq é o nome do arquivo tipo texto onde devem ser apresentados todos os dados de entrada e saída. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.
13. Desenvolver uma função **estrela\_delta**(**zab, zbc, zca, nomearq**) que tenha por objetivo obter as impedâncias de um circuito em triângulo ou delta ( $z_{ab}$ ,  $z_{bc}$  e  $z_{ca}$ ) equivalente a um circuito em estrela ( $z_a$ ,  $z_b$  e  $z_c$ ). O resultado  $R$  é ( $z_{ab}$ ,  $z_{bc}$ ,  $z_{ca}$ ) é apresentado na janela de comandos ou via arquivo. A variável nomearq é o nome do arquivo tipo texto onde devem ser apresentados todos os dados de entrada e saída. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.



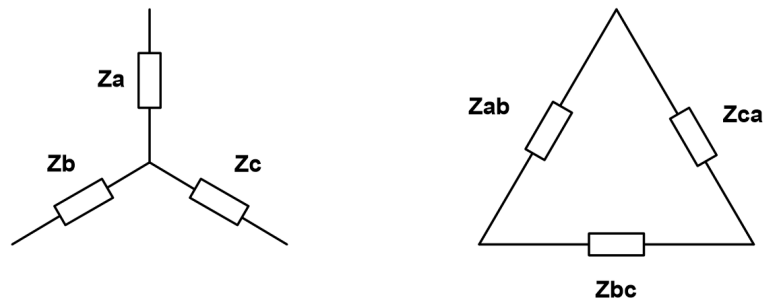


Figura 8 - Transformação Estrela - Triângulo

14. Desenvolver uma função **delta\_estrela(za, zb, zc, nomearq)** no Python que tenha por objetivo obter as impedâncias de um circuito em estrela ou Y (za, zb e zc) equivalente a um circuito em delta (zab, zbc e zca). O resultado R é (za, zb, zc) é apresentado na janela de comandos ou via arquivo. A variável nomearq é o nome do arquivo tipo texto onde devem ser apresentados todos os dados de entrada e saída. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.
15. Elabore uma função **queda1f (lc, DVc, Lc, Vfn, nomearq)** no Python que determina a seção de um circuito monofásico em cobre para um condutor por onde circula uma corrente  $I_c$ , submetido a uma queda de tensão DVc em % num condutor de cobre de comprimento Lc em metros num circuito cuja tensão fase-neutro é Vfn. A variável nomearq é o nome do arquivo tipo texto onde devem ser apresentados todos os dados de entrada e saída. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.

$\rho$  – resistividade do material condutor (cobre) :  $1/56 \Omega \cdot \text{mm}^2/\text{m}$

$L_c$  – comprimento do circuito, em m;

$I_c$  – corrente total do circuito, em A;

$\Delta V\%$  – queda de tensão máxima admitida em projeto, em %;

$V_{fn}$  – tensão entre fase e neutro, em V.

$$S_c = \frac{200 \times \rho \times \Sigma (L_c \times I_c)}{\Delta V\% \times V_{fn}} \text{ (mm}^2 \text{)}$$

16. Elabore em Python uma função **queda3f (lc, DVc, Lc, Vfn, nomearq)** que determina a seção de um circuito trifásico em cobre para um condutor por onde circula uma corrente  $I_c$ , submetido a uma queda de tensão DVc em % num condutor de cobre de comprimento Lc em metros num circuito cuja tensão

fase-neutro é Vfn. A variável nomearq é o nome do arquivo tipo texto onde devem ser apresentados todos os dados de entrada e saída. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.

$$S_c = \frac{173,2 \times \rho \times \Sigma (L_c \times I_c)}{\Delta V\% \times V_{ff}} \text{ (mm}^2\text{)}$$

$\rho$  – resistividade do material condutor (cobre) : 1/56  $\Omega \cdot \text{mm}^2/\text{m}$   
 $L_c$  – comprimento do circuito, em m;  
 $I_c$  – corrente total do circuito, em A;  
 $\Delta V\%$  – queda de tensão máxima admitida em projeto, em %;  
 $V_{ff}$  – tensão entre fase , em V.

17. Elabore usando o Python uma função **vfonte(Vc, Z, Sc, nomearq)** que calcula a tensão na fonte para alimentar uma carga  $S_c$  ( $P_c + jQ_c$ ) na tensão  $V_c$  (módulo e fase) conhecendo a impedância do alimentador  $Z$  ( $R + jX$ ). Além de no arquivo de saída apresentar a tensão na fonte é necessário também determinar a corrente no alimentador. A variável nomearq é o nome do arquivo tipo texto onde devem ser apresentados todos os dados de entrada e saída. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.
18. Elabore usando o Python uma função **vfontepi(Vc, Z, Ya, Yb, Sc, nomearq)** que calcula a tensão na fonte para alimentar uma carga  $S_c$  ( $P_c + jQ_c$ ) na tensão  $V_c$  (módulo e fase) conhecendo os parâmetros do circuito pi a impedância  $Z$  ( $R + jX$ ) e as admitâncias  $Y_a$  do lado da fonte e  $Y_b$  do lado da carga. Além de no arquivo de saída apresentar a tensão na fonte é necessário também determinar a corrente no alimentador. A variável nomearq é o nome do arquivo tipo texto onde devem ser apresentados todos os dados de entrada e saída. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.
19. Elabore usando o Python uma função **impcabo(secao\_bt, nomearq)** cujo objetivo é apresentar a impedância de um cabo de cobre com isolamento em PVC. A variável nomearq é o nome do arquivo tipo texto onde devem ser apresentados todos os dados de entrada e saída. Apresente um exemplo e envie o arquivo teste e a biblioteca **asp1**.

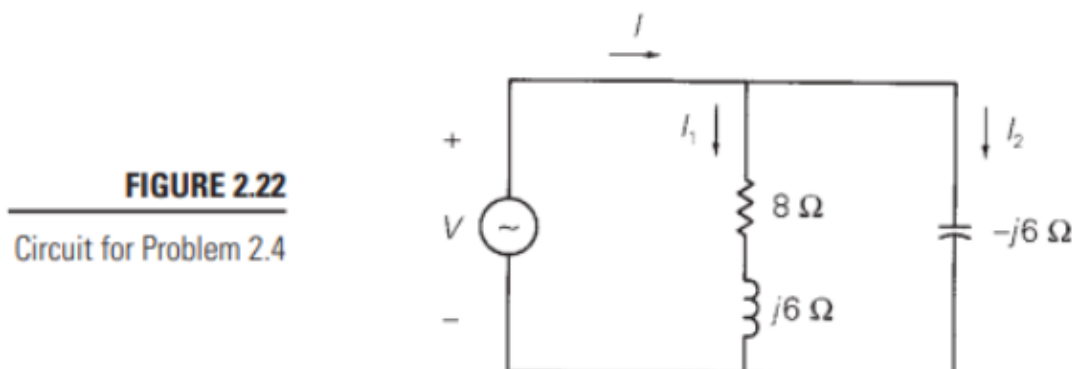
Seção nominal (mm <sup>2</sup> )	Resistência (mΩ/m)	Reatância (mΩ/m)	Seção nominal (mm <sup>2</sup> )	Resistência (mΩ/m)	Reatância (mΩ/m)
1	22,1	0,176	70	0,328	0,0965
1,5	14,8	0,168	95	0,236	0,0975
2,5	8,91	0,155	120	0,188	0,0939
4	5,57	0,143	150	0,153	0,0928
6	3,71	0,135	185	0,123	0,0908
10	2,24	0,119	240	0,0943	0,0902
16	1,41	0,112	300	0,0761	0,0895
25	0,880	0,106	400	0,0607	0,0876
35	0,841	0,101	500	0,0496	0,0867
50	0,473	0,101	630	0,0402	0,0865

20. Responda às seguintes questões:

- 2.1** Given the complex numbers  $A_1 = 6/\underline{30}$  and  $A_2 = 4 + j5$ , (a) convert  $A_1$  to rectangular form; (b) convert  $A_2$  to polar and exponential form; (c) calculate  $A_3 = (A_1 + A_2)$ , giving your answer in polar form; (d) calculate  $A_4 = A_1 A_2$ , giving your answer in rectangular form; (e) calculate  $A_5 = A_1/(A_2^*)$ , giving your answer in exponential form.
- 2.2** Convert the following instantaneous currents to phasors, using  $\cos(\omega t)$  as the reference. Give your answers in both rectangular and polar form.
- (a)  $i(t) = 500\sqrt{2} \cos(\omega t - 30)$
- (b)  $i(t) = 4 \sin(\omega t + 30)$
- (c)  $i(t) = 5 \cos(\omega t - 15) + 4\sqrt{2} \sin(\omega t + 30)$

21. Responda a seguinte questão:

- 2.4** For the single-phase circuit shown in Figure 2.22,  $I = 10/\underline{0^\circ}\text{A}$ . (a) Compute the phasors  $I_1$ ,  $I_2$ , and  $V$ . (b) Draw a phasor diagram showing  $I$ ,  $I_1$ ,  $I_2$ , and  $V$ .

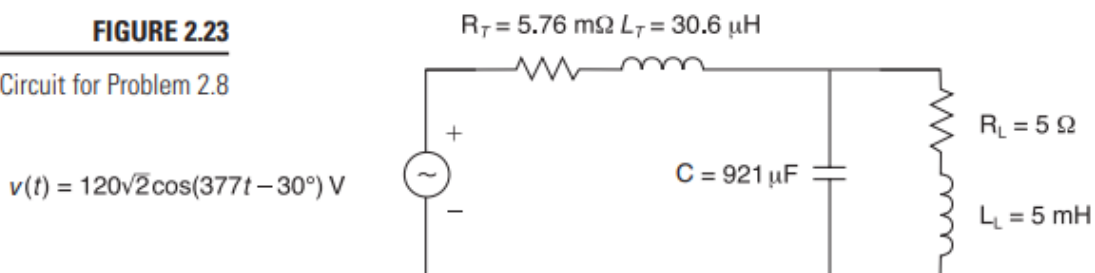


22. Responda às seguintes questões:

- 2.7** Let a 100-V sinusoidal source be connected to a series combination of a 3- $\Omega$  resistor, an 8- $\Omega$  inductor, and a 4- $\Omega$  capacitor. (a) Draw the circuit diagram. (b) Compute the series impedance. (c) Determine the current  $I$  delivered by the source. Is the current lagging or leading the source voltage? What is the power factor of this circuit?
- 2.8** Consider the circuit shown in Figure 2.23 in time domain. Convert the entire circuit into phasor domain.

**FIGURE 2.23**

Circuit for Problem 2.8

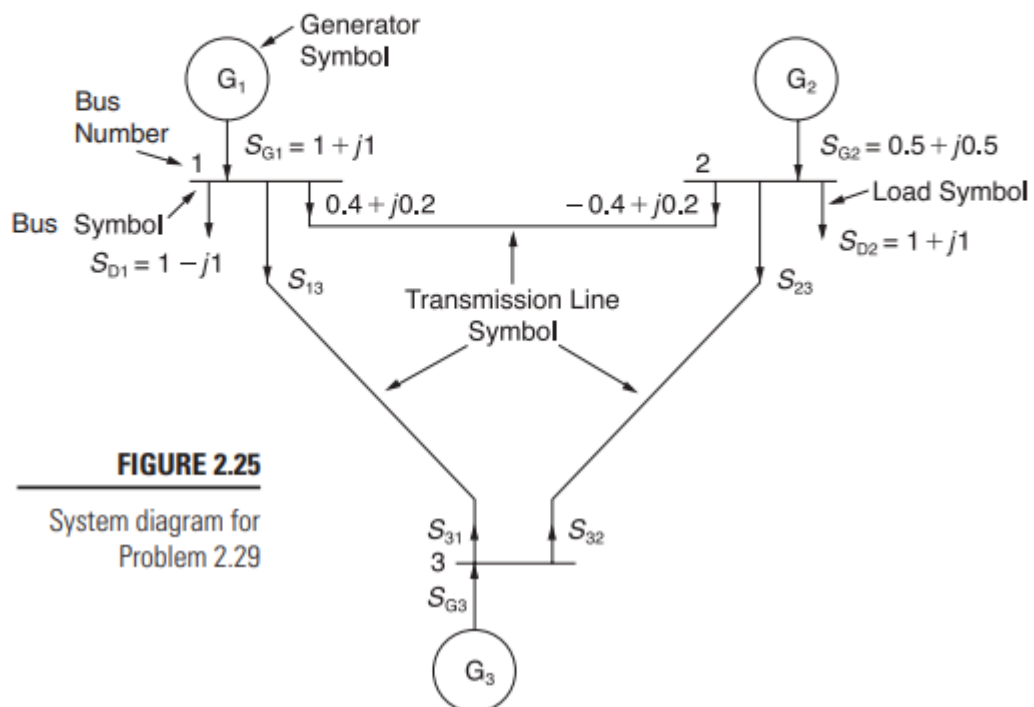


23. Responda às seguintes questões:

- 2.26** A small manufacturing plant is located 2 km down a transmission line, which has a series reactance of 0.5  $\Omega/\text{km}$ . The line resistance is negligible. The line voltage at the plant is 480  $\angle 0^\circ$  V (rms), and the plant consumes 120 kW at 0.85 power factor lagging. Determine the voltage and power factor at the sending end of the transmission line by using (a) a complex power approach and (b) a circuit analysis approach.
- 2.27** An industrial load consisting of a bank of induction motors consumes 50 kW at a power factor of 0.8 lagging from a 220-V, 60-Hz, single-phase source. By placing a bank of capacitors in parallel with the load, the resultant power factor is to be raised to 0.95 lagging. Find the net capacitance of the capacitor bank in  $\mu\text{F}$  that is required.
- 2.28** Three loads are connected in parallel across a single-phase source voltage of 240 V (RMS).  
 Load 1 absorbs 15 kW and 6.667 kvar;  
 Load 2 absorbs 3 kVA at 0.96PF leading;  
 Load 3 absorbs 15 kW at unity power factor.  
 Calculate the equivalent impedance,  $Z$ , for the three parallel loads, for two cases:  
 (i) Series combination of  $R$  and  $X$ , and (ii) parallel combination of  $R$  and  $X$ .

24. Responda às seguintes questões:

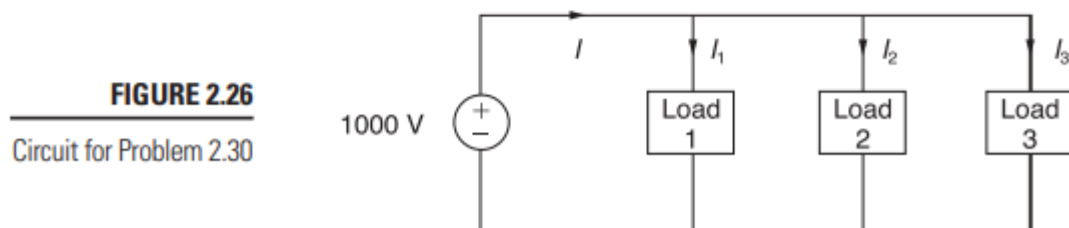
- 2.29** Modeling the transmission lines as inductors, with  $S_{ij} = S_{ji}^*$ , Compute  $S_{13}$ ,  $S_{31}$ ,  $S_{23}$ ,  $S_{32}$ , and  $S_{G3}$  in Figure 2.25. (Hint: complex power balance holds good at each bus, satisfying KCL.)



**FIGURE 2.25**

System diagram for  
Problem 2.29

- 2.30** Figure 2.26 shows three loads connected in parallel across a 1000-V (RMS), 60-Hz single-phase source.
- Load 1: Inductive load, 125 kVA, 0.28PF lagging.
  - Load 2: Capacitive load, 10 kW, 40 kvar.
  - Load 3: Resistive load, 15 kW.
- (a) Determine the total kW, kvar, kva, and supply power factor.
- (b) In order to improve the power factor to 0.8 lagging, a capacitor of negligible resistance is connected in parallel with the above loads. Find the kvar rating of that capacitor and the capacitance in  $\mu F$ .
- Comment on the magnitude of the supply current after adding the capacitor.



**FIGURE 2.26**

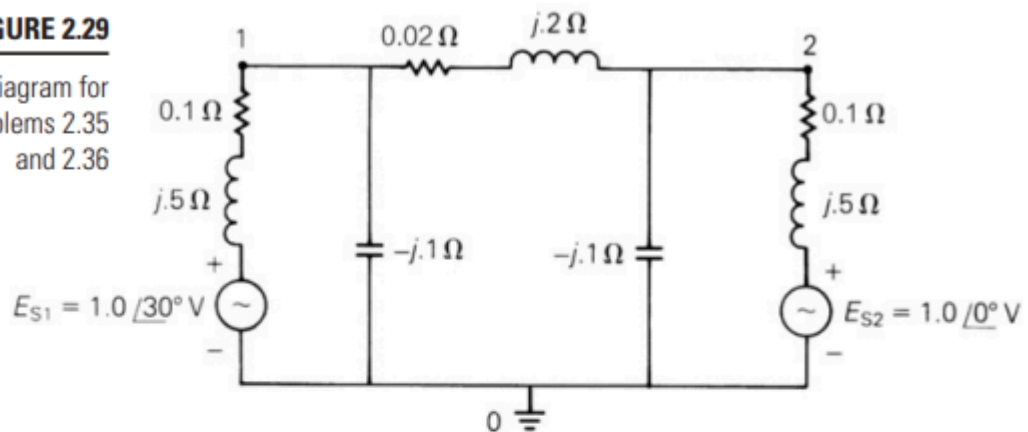
Circuit for Problem 2.30

25. Responda às seguintes questões:

- 2.35** For the circuit shown in Figure 2.29, convert the voltage sources to equivalent current sources and write nodal equations in matrix format using bus 0 as the reference bus. Do not solve the equations.

**FIGURE 2.29**

Circuit diagram for  
Problems 2.35  
and 2.36



- 2.36** For the circuit shown in Figure 2.29, (a) determine the  $2 \times 2$  bus admittance matrix  $\mathbf{Y}_{\text{bus}}$ , (b) convert the voltage sources to current sources and determine the vector of source currents into buses 1 and 2.
- 2.37** Determine the  $4 \times 4$  bus admittance matrix  $\mathbf{Y}_{\text{bus}}$  and write nodal equations in matrix format for the circuit shown in Figure 2.30. Do not solve the equations.

**FIGURE 2.30**

Circuit for Problem 2.37

