UNIVERSIDADE FEDERAL DE SÃO CARLOS
Centro de Ciências Exatas e de Tecnologia
Departamento de Física

Trabalho 1

por

Vitor de Souza Barboza/791446 - Curso: Engenharia Física

Departamento de Física
São Carlos - SP, 6 de janeiro de 2023

Ministrante:
Prof. Leonardo Kleber Castelano

1 Introdução

Este trabalho consiste na resolução do "Trabalho 1" da disciplina de Física Computacional 1. Todos os arquivos utilizados na prática podem ser acessados no repositório do GitHub [1]

2 Resolução

- **Questão 1**

1. (2,0) Uma versão simplificada de um receptor de rádio AM consiste de um circuito RLC ajustável contendo um resistor, um capacitor e um indutor conectados em série. O circuito RLC é conectado a uma antena e ao terra. O circuito ajustável permite que o rádio selecione uma estação específica dentre todas transmitidas na banda de frequências AM. O rádio recebe o sinal mais forte na chamada frequência de ressonância do circuito, que é dada aproximadamente por,

$$f_0 = \frac{1}{2\pi\sqrt{LC}},$$

onde L é a indutância em henrys (H) e C é a capacitância em farads (F). Escreva um programa que calcule e imprima no monitor a frequência de ressonância, quando especificados os valores de L e C fornecidos pelo usuário. Teste seu programa para frequência de ressonância para $L = 0,1$mH e $C = 0,25$nF.

O código para resolução da questão 1 pode ser visto abaixo:

```python
# Questão 1

import numpy as np

def frequencia(L,C):
    f0 = 1/(2*np.pi*np.sqrt(L*C*(1e-12)))

    return f0

L = float(input("Digite o valor da indutância L (em mH): "))
C = float(input("Digite o valor da capacitância C (em nF): "))
print(f"O valor da frequência é: {frequencia(L,C)}")
```

Para o teste do programa, foi utilizado os valores $L = 0,1\,mH$ e $C = 0,25\,nF$. O resultado está exibido abaixo:

```
O valor da frequência é: 1006584.2420897408
```

- **Questão 2**

2. (2,0) O produto de uma reação química, que varia de 0 a 100% em t segundos, segue a relação $100(1-e^{-kt})$, onde $k = e^{-q}$, $q = \frac{2000}{T+273{,}16}$ e T é a temperatura medida em °C. (a) Escreva um programa que permite ao usuário entrar com a temperatura e em seguida imprima no monitor e escreva em um arquivo o produto da reação a cada minuto até alcançar 95%. (b) Determine em quantos segundos levará para alcançar esse valor numa temperatura de 25°C. (c) Faça um gráfico a partir do arquivo do item (a) com o auxílio do módulo matplotlib.

Os códigos criados para resolver o exercício estão mostrados abaixo:

```
# Questão 2 a) e b)

import matplotlib as plt
import numpy as np

temp = int(input("Digite a temperatura (°C): "))

q = 2000/(temp + 273.16)
k = np.exp(-q)
t = -1
produto = 0
arquivo = open("produtos.txt", 'w')
arquivo.write("")
arquivo = open("produtos.txt", 'a')
arquivo.write("Tempo Produto \n")

while produto <= 95:
    t = t + 1
    minuto = int(t/60)
    produto = 100*(1-np.exp(-k*t))
    produto_formatado = "{:.2f}".format(produto)


    if t%60 == 0:
        arquivo.write(f'{minuto} {produto_formatado}\n')
        print(f"t = {minuto} e produto = {produto_formatado}")



print(f"O tempo (em segundos) levado para atingir 95% é {t} s.")

arquivo.close()
```

```
# Questão 2 c)

import matplotlib.pyplot as plt
import pandas as pd

arquivo = pd.read_csv("produtos.txt", sep = " ")

x = arquivo['Tempo']
y = arquivo['Produto']

plt.plot(x,y)
plt.xlabel("Tempo (minutos)")
plt.ylabel("Produto (%)")
plt.show
```

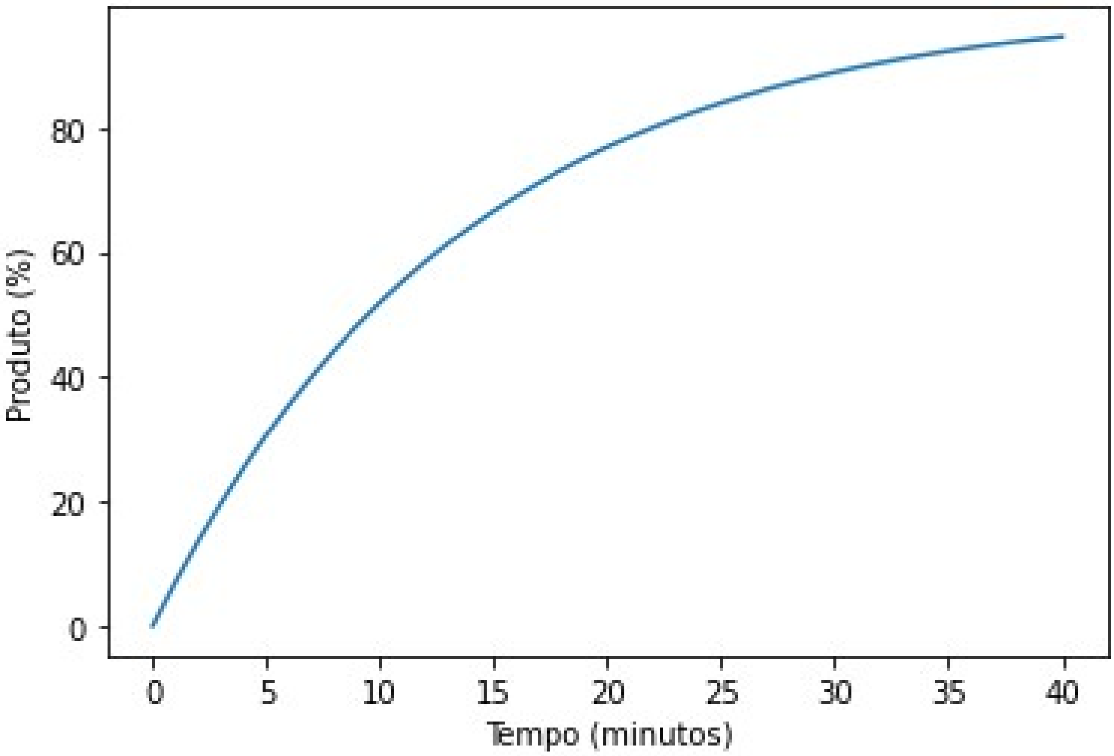
Como resposta para o item a) e b) o código imprime a porcentagem de produto por minuto, o tempo necessário para alcançar a temperatura informada e salva os pares de pontos de produto por minuto em um arquivo chamado "*produtos.txt*". A resposta do código pode ser mostrado abaixo (a temperatura utilizada foi 25 $°C$):

```
t = 0 e produto = 0.00
t = 1 e produto = 7.07
t = 2 e produto = 13.63
t = 3 e produto = 19.74
t = 4 e produto = 25.41
t = 5 e produto = 30.68
t = 6 e produto = 35.58
t = 7 e produto = 40.13
t = 8 e produto = 44.36
t = 9 e produto = 48.29
t = 10 e produto = 51.94
t = 11 e produto = 55.34
t = 12 e produto = 58.50
t = 13 e produto = 61.43
t = 14 e produto = 64.15
t = 15 e produto = 66.69
t = 16 e produto = 69.04
t = 17 e produto = 71.23
t = 18 e produto = 73.26
t = 19 e produto = 75.15
t = 20 e produto = 76.91
```

```
t = 20 e produto = 76.91
t = 21 e produto = 78.54
t = 22 e produto = 80.05
t = 23 e produto = 81.46
t = 24 e produto = 82.77
...
t = 38 e produto = 93.82
t = 39 e produto = 94.26
t = 40 e produto = 94.67
O tempo (em segundos) levado para atingir 95% é 2453 s.
```

O arquivo de texto "*produtos.txt*" gerado pela gravação dos pares de pontos está disponível no repostitório no GitHub [1].

Para responder o item c) foi construído o gráfico dos pontos obtidos através do arquivo de texto "*produtos.txt*":



- **Questão 3**

3. (2,0) Um gás ideal é aquele onde a colisão entre as moléculas são perfeitamente elásticas. É possível pensar nas moléculas num gás ideal como bolas de bilhar perfeitamente rígidas que colidem umas com as outras sem perder energia cinética. Tal gás pode ser caracterizado por três quantidade: pressão absoluta P, volume V e temperatura T. A relação entre essas quantidades é conhecida como a lei dos gases ideais,

$$PV = nRT,$$

onde n é o número de moléculas em unidades de mol (1mol= 6.02×10^{23}). Com a pressão dada em quilo-pascal (kPa), volume em litros (L) e temperatura em kelvins (K), a constante dos gases ideais vale $R = 8.314\text{L} \cdot \text{kPa/mol} \cdot \text{K}$. Considere que um mol de um gás ideal está a temperatura de 273K. (a) Escreva um programa em Python que grave em um arquivo o volume do gás em função de sua pressão para P variando de 1kPa até 1000kPa em passos de 50.5kPa. (b) Grave em um outro arquivo os mesmos dados, mas agora para $T = 300\text{K}$. (c) Faça um gráfico a partir dos arquivos gerados nos itens anteriores com o volume do gás em função da pressão para as duas temperaturas com o auxílio do matplotlib.

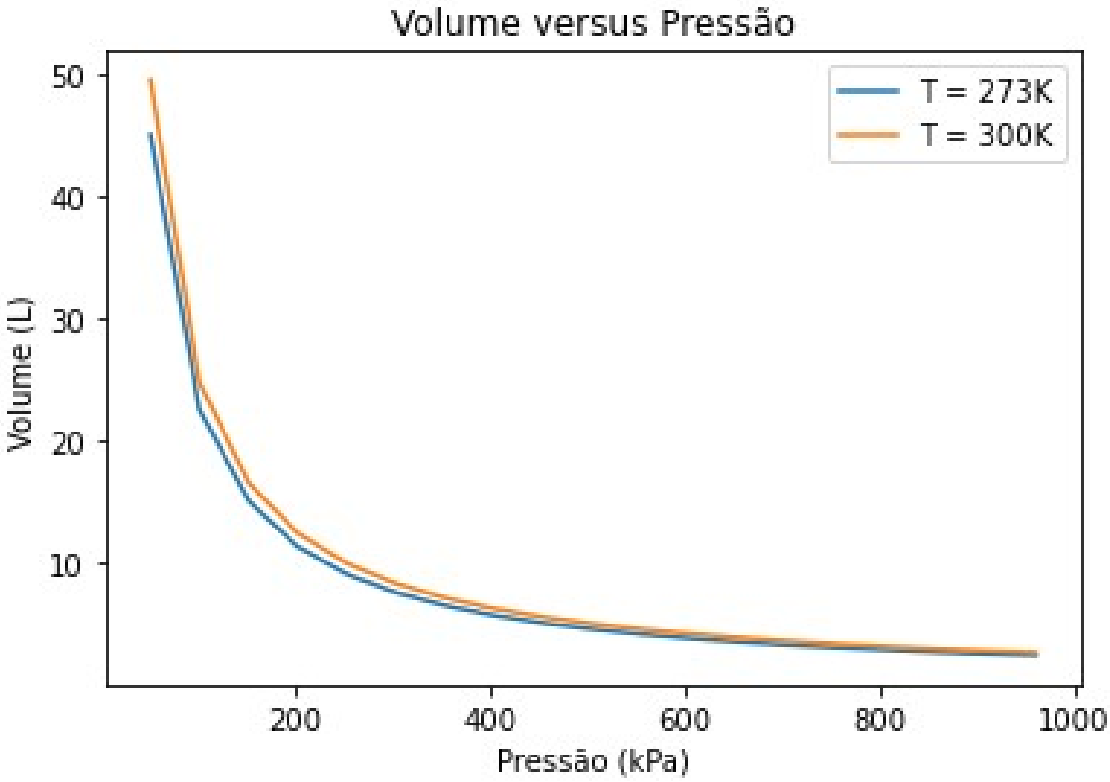
Os códigos criados para responder a questão 3 estão mostrados abaixo:

```python
# Questão 3 a) e b)

R = 8.314
T = float(input("Informe a Temperatura T(K): "))
i = 1
nome_arquivo = input("Digite o nome do arquivo a ser criado: ")
arquivo = open(f"{nome_arquivo}.txt", 'w')
arquivo.write("")
arquivo = open(f"{nome_arquivo}.txt", 'a')
arquivo.write("Pressao(kPa) Volume(L)\n")

for P in range(50,1000,50):
    P = P + 0.5 * i
    V = R * T / P
    arquivo.write(f"{P} {V}\n")
    i = i + 1


arquivo.close()
```

```python
# Questão 3 c)

import pandas as pd
import matplotlib.pyplot as plt

primeira = pd.read_csv("primeira.txt", sep=" ")
segunda = pd.read_csv("segunda.txt", sep=" ")

x = primeira["Pressao(kPa)"]
y1 = primeira["Volume(L)"]
y2 = segunda["Volume(L)"]

plt.plot(x, y1, y2)
plt.xlabel("Pressão (kPa)")
plt.ylabel("Volume (L)")
plt.legend(["T = 273K", "T = 300K"])
plt.title("Volume versus Pressão")
plt.show
```

Com isso foram gerados dois arquivos de texto nomeados ”*primeira.txt*” e ”*segunda.txt*” onde estão gravados os pares de pontos para a temperatura de 273 K e 300 K, respectivamente. Os arquivos podem ser encontrados no repositório do GitHub [1].

Para responder o item c), o gráfico construído a partir dos pontos contidos nos arquivos ”*primeira.txt*” e ”*segunda.txt*” pode ser visto abaixo:



- **Questão 4**

4. (2,0) Um CPF é uma sequência de 9 algarismos $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9$ seguidos de dois dígitos verificadores v_1, v_2. Por exemplo, no CPF:

$$4\,9\,3\,1\,7\,4\,1\,0\,0 - 2\,2$$

temos que $a_1 = 4$, $a_2 = 9$, $a_3 = 3$, ..., $a_9 = 0$, $v_1 = 2$, $v_2 = 2$. Os dígitos verificadores são determinados a partir dos outros como se segue. Primeiramente, calcula-se as quantidades S_1 e S_2,

$$S_1 = a_1 \times 1 + a_2 \times 2 + a_3 \times 3 + a_4 \times 4 + a_5 \times 5 + a_6 \times 6 + a_7 \times 7 + a_8 \times 8 + a_9 \times 9$$
$$S_2 = a_1 \times 9 + a_2 \times 8 + a_3 \times 7 + a_4 \times 6 + a_5 \times 5 + a_6 \times 4 + a_7 \times 3 + a_8 \times 2 + a_9 \times 1$$

Em seguida, calcula-se os dígitos verificadores tomando o resto da divisão de S_1 e S_2 por 11, ou seja,

$$v_1 = \text{mod}(S_1, 11) \;\; v_2 = \text{mod}(S_2, 11)$$

caso o resto da divisão seja 10, o dígito verificador é definido como zero. Um CPF é falso se as condições acima não forem satisfeitas.

(a) Escreva um programa para calcular os dígitos verificadores v_1, v_2 a partir de uma sequência de nove dígitos fornecida pelo usuário.

(b) Dentre os CPFs abaixo, marque os que são verdadeiros

( ) $9\,4\,7\,1\,1\,2\,9\,1\,1 - 9\,7$

( ) $1\,4\,9\,2\,1\,5\,2\,1\,9 - 4\,4$

( ) $9\,9\,1\,4\,4\,2\,1\,1\,7 - 2\,5$

( ) $7\,2\,1\,4\,5\,3\,2\,2\,2 - 0\,5$

(c) Quais os números verificadores de um CPF em que os 9 primeiros números são $1\,1\,1\,4\,4\,4\,7\,7\,7$?

O código para a questão 4 pode ser visualizado abaixo:

```python
# Questão 4

import numpy as np

seq = []
i = 1
s1 = 0
s2 = 0

while i < 10:
    num = int(input("Digite o número: "))
    seq.append(num)
    i += 1

for j in range(0,9):
    s1 = s1 + (seq[j] * (j+1))
    s2 = s2 + (seq[j] * (9-j))

v1 = s1 % 11
v2 = s2 % 11

if v1 == 10:
    v1 = 0

if v2 == 0:
    v2 = 0

print(f"Os dígitos verificadores são {v1} e {v2}.")
```

Para responder o item b):

- 947112911-97 $\longrightarrow Falso$
- 149215219-44 $\longrightarrow Falso$
- 991442117-25 $\longrightarrow Falso$
- 721453222-05 $\longrightarrow Verdadeiro$

Por fim, para o item c) os dígitos verificadores encontrados foram: 3 e 5.

- **Questão 5**

5. (2,0) A razão áurea é o número irracional dado por $\phi = \frac{1+\sqrt{5}}{2}$. Uma maneira de determinar ϕ é através da sequência de Fibonacci. Essa sequência é construída de modo que cada número da sequência a_n seja igual a soma dos dois números anteriores a_{n-1} e a_{n-2}, ou seja, $a_n = a_{n-1} + a_{n-2}$. A sequência inicia-se com os números 0 e 1. Os primeiros termos da sequência de Fibonacci são portanto

$$0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, \ldots$$

Kepler mostrou que a razão entre um número da sequência e o número imediatamente anterior tende para a razão áurea conforme o elemento da sequência cresce, i.e.,

$$\lim_{n\to\infty} \frac{a_n}{a_{n-1}} = \phi$$

Escreva um programa que calcula e imprime na tela a sequência de Fibonacci, utilizando como critério de parada da sequência que a diferença entre a razão de dois números consecutivos da sequência e a razão áurea ϕ seja menor em módulo que $\epsilon = 10^{-6}$. Determine o número de termos da sequência necessários para se chegar a essa condição.

O código criado para a questão 5 pode ser visto abaixo:

```
# Questão 5

import numpy as np

aurea = (1 + np.sqrt(5) )/2
E = aurea
i = 2
fib = [0, 1]
while E >= 1e-6:
    E = aurea
    termo = fib[i-1] + fib[i-2]
    fib.append(termo)
    E = np.abs(aurea - (fib[i]/fib[i-1]))
    i += 1

print(f"O número de termos foi {i} e a sequencia é {fib}")
```

A resposta do código:

```
O número de termos foi 18 e a sequencia é [0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597]
```

Bibliografia

[1] https://github.com/vitorsbarboza/FisComp1