

ANO
2025



**ATIVIDADE PRÁTICA
MÓDULO C
LÓGICA DE PROGRAMAÇÃO
E ALGORITMOS**

Simei Pereira Moura. RU: 5489190

Prof. Me. Bruno Kostiuk

QUESTÃO 1 de 4 – Conteúdos até Aula 3

Enunciado: Imagina-se que você é um dos programadores responsáveis pela construção de app para uma empresa X que vende Planos de Saúde. Uma das estratégias dessa empresa X é cobrar um valor diferente com base na idade do cliente, conforme a **listagem abaixo**:

- Se a idade for **maior ou igual que 0 e menor que 19**, o valor será de **100%** do **valor base** do plano (100 / 100);
- Se a idade for **maior ou igual que 19 e menor que 29**, o valor será de **150%** do **valor base** do plano (150 / 100);
- Se a idade for **maior ou igual que 29 e menor que 39**, o valor será de **225%** do **valor base** do plano (225 / 100);
- Se a idade for **maior ou igual que 39 e menor que 49**, o valor será de **240%** do **valor base** do plano (240 / 100);
- Se a idade for **maior ou igual que 49 e menor que 59**, o valor será de **350%** do **valor base** do plano (350 / 100);
- Se a idade for **maior ou igual que 59**, o valor será de **600%** do **valor base** do plano (600 / 100);

O valor mensal do plano é calculado da seguinte maneira:

$$\text{valorMensal} = \text{valorBase} * \text{porcentagem}$$

Exemplo: Se o **valorBase** informado for **100.00** e a **idade** for **45** anos (**240%** segundo a tabela acima)

$$\text{valorMensal} = 100.00 * \left(\frac{240}{100}\right) = R\$ 240.00$$

Elabore um programa em Python que:

- Deve-se implementar o print com uma mensagem de boas-vindas que apareça o seu **nome e sobrenome** (somente print, não usar input aqui). **[EXIGÊNCIA DE CÓDIGO 1 de 6]**;
- Deve-se implementar o input do **valorBase** do plano e da **idade** do cliente **[EXIGÊNCIA DE CÓDIGO 2 de 6]**;
- Deve-se implementar as regras de valores **conforme a enunciado acima** (obs.: atente-se as condições de menor, igual e maior) **[EXIGÊNCIA DE CÓDIGO 3 de 6]**;
- Deve-se implementar o **valorMensal** **[EXIGÊNCIA DE CÓDIGO 4 de 6]**;
- Deve-se implementar as estruturas **if, elif e else (todas elas)** **[EXIGÊNCIA DE CÓDIGO 5 de 6]**;
- Deve-se inserir comentários relevantes no código **[EXIGÊNCIA DE CÓDIGO 6 de 6]**;

Teste seu código atendendo as seguintes exigências:

- Deve-se apresentar na saída de console uma mensagem de boas-vindas com seu **nome e sobrenome** **[EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 2]**;
- Deve-se apresentar na saída de console a utilização do sistema informando uma **idade maior ou igual a 29 anos**, apresentando na saída de console o **valorMensal** do plano **[EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 2]**;

EXEMPLO DE SAÍDA DE CONSOLE:



Bem vindo ao Sistema do Bruno Kostiuk
Informe o valor Base do plano: R\$ 134.05
Informe a idade do cliente: 34
O valor mensal do plano é de: R\$ 301.61

Figura 1: Exemplo de saída de console que o aluno deve fazer. Em que se perguntar o valorBase do plano (pode ser qualquer valor) e a idade (maior ou igual a 29 anos [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 2]), e é apresentado o valorMensal.

Apresentação de Código da Questão 1:

```
print()
print('Bem vindo ao Sistema do Simei Pereira Moura')
valorBase = float(input('Informe o valor Base do plano: R$ '))
idade = int(input('Informe a idade do cliente: '))
#estrutura de decisão abaixo, informando a idade e calculo do valor conforme idade
if (idade >= 0 and idade < 19):
    valorMensal = valorBase * 100 / 100

elif (idade >= 19 and idade < 29):
    valorMensal = valorBase * 150 / 100

elif (idade >= 29 and idade < 39):
    valorMensal = valorBase * 225 / 100

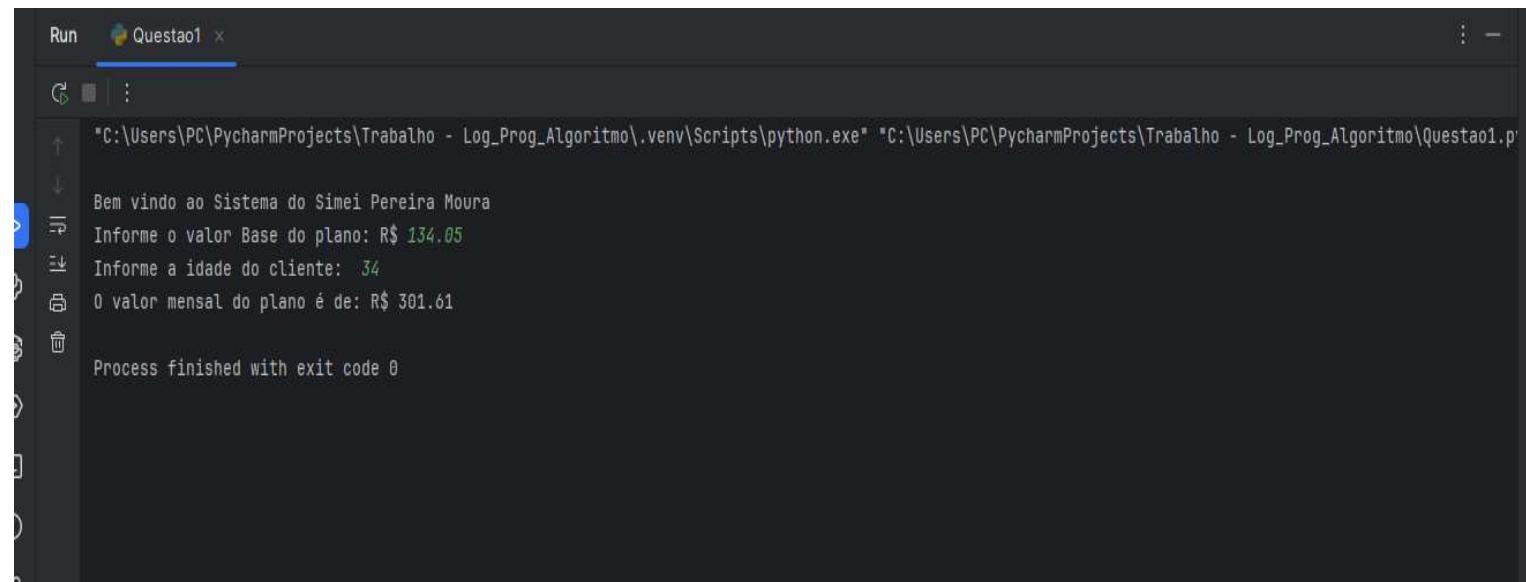
elif (idade >= 39 and idade < 49):
    valorMensal = valorBase * 240 / 100

elif (idade >= 49 and idade < 59):
    valorMensal = valorBase * 350 / 100

else:
    valorMensal = valorBase * 600 / 100

print(f'O valor mensal do plano é de: R$ {valorMensal:.2f}') #informa o valor mensal do plano
```

Apresentação de Saída do Console da Questão 1:



The screenshot shows the PyCharm interface with the 'Run' tab selected. The console tab displays the following output:

```
*C:\Users\PC\PycharmProjects\Trabalho - Log_Prog_Algoritmo\.venv\Scripts\python.exe* "C:\Users\PC\PycharmProjects\Trabalho - Log_Prog_Algoritmo\Questao1.py"
Bem vindo ao Sistema do Simei Pereira Moura
Informe o valor Base do plano: R$ 134.05
Informe a idade do cliente: 34
O valor mensal do plano é de: R$ 301.61
Process finished with exit code 0
```

QUESTÃO 2 de 4 - Conteúdo até aula 04

Enunciado: Você e sua equipe de programadores foram contratados para desenvolver um app de vendas para uma Pizzaria que vende sabores de Pizzas Doces e Pizzas Salgadas. Você ficou com a parte de desenvolver a interface do cliente para retirada do produto.

A Loja possui seguinte relação:

- Tamanho **P**: Pizza Salgada (**PS**) custa 30 reais e a Pizza Doce (**PD**) custa 34 reais;
- Tamanho **M**: Pizza Salgada (**PS**) custa 45 reais e a Pizza Doce (**PD**) custa 48 reais;
- Tamanho **G**: Pizza Salgada (**PS**) custa 60 reais e a Pizza Doce (**PD**) custa 66 reais;

Elabore um programa em Python que:

- Deve-se implementar o print com uma mensagem de boas-vindas que apareça o seu **nome e sobrenome** (somente print, não usar input aqui). Além do seu nome completo, deve-se implementar um **print com um Menu** para o cliente. [EXIGÊNCIA DE CÓDIGO 1 de 8];
- Deve-se implementar o input do **sabor** (PS/PD) e o print "Sabor inválido. Tente novamente" se o usuário entra com valor diferente de PS e PD [EXIGÊNCIA DE CÓDIGO 2 de 8];
- Deve-se implementar o input do **tamanho** (P/M/G) e o print "Tamanho inválido. Tente novamente" se o usuário com entra valor diferente de P, M ou G [EXIGÊNCIA DE CÓDIGO 3 de 8];
- Deve-se implementar **if, elif e/ou else**, utilizando o modelo **aninhado** (aula 3 – Tema 4) com cada uma das combinações de **sabor e tamanho** [EXIGÊNCIA DE CÓDIGO 4 de 8];
- Deve-se implementar um **acumulador** para somar os valores dos pedidos (valor total do pedido) [EXIGÊNCIA DE CÓDIGO 5 de 8];
- Deve-se implementar o input com a pergunta: "Deseja pedir mais alguma coisa?". Se sim **repetir a partir do item B**, senão encerrar o programa executar o print do **acumulador** [EXIGÊNCIA DE CÓDIGO 6 de 8];
- Deve-se implementar as estruturas de **while, break, continue (todas elas)** [EXIGÊNCIA DE CÓDIGO 7 de 8];
- Deve-se inserir comentários relevantes no código [EXIGÊNCIA DE CÓDIGO 8 de 8];

Teste seu código atendendo as seguintes exigências:

- Deve-se apresentar na saída de console uma mensagem de boas-vindas com o seu **nome e sobrenome** e o menu para o cliente conhecer as opções [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 4];
- Deve-se apresentar na saída de console um pedido em que o usuário errou o **sabor** [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 4];
- Deve-se apresentar na saída de console um pedido em que o usuário errou o **tamanho** [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 4];
- Deve-se apresentar na saída de console um pedido com **duas opções sabores diferentes e com tamanhos diferentes** [EXIGÊNCIA DE SAÍDA DE CONSOLE 4 de 4];

EXEMPLO DE SAÍDA DE CONSOLE:

----- Bem-vindo a Pizzaria do Bruno Kostiuk -----
 ----- Cardápio -----

Tamanho	Pizza Salgada(PS)	Pizza Doce(PD)	
P	R\$ 30.00	R\$ 34.00	
M	R\$ 45.00	R\$ 48.00	
G	R\$ 60.00	R\$ 66.00	

Entre com o sabor desejado (PS/PD): PG Usuário errou o sabor

Sabor inválido. Tente novamente

Entre com o sabor desejado (PS/PD): PS

Entre com o tamanho desejado (P/M/G): EXGG Usuário errou o tamanho

Tamanho inválido. Tente novamente

Entre com o sabor desejado (PS/PD): PS

Entre com o tamanho desejado (P/M/G): G

Você pediu uma Pizza Salgada no tamanho G: R\$ 60.00

Deseja mais alguma coisa? (S/N): S

Entre com o sabor desejado (PS/PD): PD

Entre com o tamanho desejado (P/M/G): M

Você pediu um Filé de Frango no tamanho M: R\$ 48.00

Deseja mais alguma coisa? (S/N): N

O valor total a ser pago: R\$ 108.00

**Mensagem com seu nome completo
e Menu de opções (cardápio)**

**Pedido com 2 itens de tamanho
e sabores diferentes**

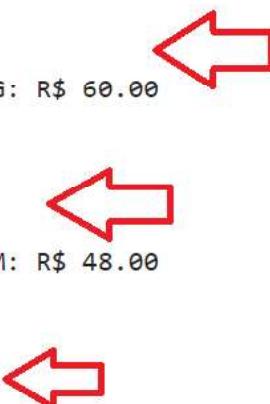


Figura 2: Exemplo de saída de console que o aluno deve fazer. Em que se perguntar o sabor ao tamanho. Há uma tentativa de pedido que se erro o sabor e outra que se errou o tamanho. Há também dois pedidos com sabores e tamanhos diferentes.

Apresentação de Código da Questão 2:

```
print()
#Menu Cardápio
print('-----Bem-vindo a Pizzaria do Simei Pereira Moura-----')
print('-----Cardápio-----')
print('-----')
print('--- | Tamanho | Pizza Salgada(PS) | Pizza Doce(PD) | ---')
print('--- | P | R$ 30.00 | R$ 34.00 | ---')
print('--- | M | R$ 45.00 | R$ 48.00 | ---')
print('--- | G | R$ 60.00 | R$ 66.00 | ---')
print('-----')

valor = 0 #variavel para acumular os valores

while True: #loop para validar o sabor desejado
    print()
    saborDesejado = input('Entre com o sabor desejado (PS/PD): ')
    if saborDesejado != 'PS' and saborDesejado != 'PD':
        print('Sabor invalido. Tente novamente')
        continue
    tamanho = input('Entre com o tamanho desejado (P/M/G): ')
    if tamanho != 'P' and tamanho != 'M' and tamanho != 'G':
        print('Tamanho inválido. Tente novamente')
        continue
    if saborDesejado == 'PS' and tamanho == 'P':
        print(f'Voce pediu uma Pizza Salgada no tamanho {tamanho}: R$ 30.00')
        valor += 30.00 #valores são acumulados na variavel acumuladora
    elif saborDesejado == 'PS' and tamanho == 'M':
        print(f'Voce pediu uma Pizza Salgada no tamanho {tamanho}: R$ 45.00 ')
        valor += 45.00
    elif saborDesejado == 'PS' and tamanho == 'G':
        print(f'Voce pediu uma Pizza Salgada no tamanho {tamanho}: R$ 60.00')
        valor += 60.00
    elif saborDesejado == 'PD' and tamanho == 'P':
        print(f'Voce pediu uma Pizza Doce no tamanho {tamanho}: R$ 34.00')
        valor += 34.00
    elif saborDesejado == 'PD' and tamanho == 'M':
        print(f'Voce pediu uma Pizza Doce no tamanho {tamanho}: R$ 48.00')
        valor += 48.00
    elif saborDesejado == 'PD' and tamanho == 'G':
        print(f'Voce pediu uma Pizza Doce no tamanho {tamanho}: R$ 66.00')
        valor += 66.00
    print()
    adicionar = input('Deseja mais alguma coisa? (S/N): ')

    if adicionar == 'S':
        continue #se digitar o S (sim) volta ao inicio

    else:
        print()
        print(f'O valor total a pagar: R$ {valor}') #valor total a pagar
        break
```

Apresentação de Saída do Console da Questão 2:

```
"C:\Users\PC\PycharmProjects\Trabalho - Log_Prog_Algoritmo\.venv\Scripts\python.exe" "C:\Users\PC\PycharmProjects\Trabalho - Log_Prog_Algoritmo\Questao2.py"

-----Bem-vindo a Pizzaria do Simei Pereira Moura-----
-----Cardápio-----

--- | Tamanho | Pizza Salgada(PS) | Pizza Doce(PD) | ---
--- | P       | R$ 30.00        | R$ 34.00      | ---
--- | M       | R$ 45.00        | R$ 48.00      | ---
--- | G       | R$ 60.00        | R$ 66.00      | ---

Entre com o sabor desejado (PS/PD): PG
Sabor invalido. Tente novamente

Entre com o sabor desejado (PS/PD): PS
Entre com o tamanho desejado (P/M/G): EXG6
Tamanho inválido. Tente novamente

Entre com o sabor desejado (PS/PD): PS
Entre com o tamanho desejado (P/M/G): G
Voce pediu uma Pizza Salgada no tamanho G: R$ 60.00

Deseja mais alguma coisa? (S/N): S

Entre com o sabor desejado (PS/PD): PD
Entre com o tamanho desejado (P/M/G): M
Voce pediu uma Pizza Doce no tamanho M: R$ 48.00

Deseja mais alguma coisa? (S/N): N

O valor total a pagar: R$ 108.0

Process finished with exit code 0
```

QUESTÃO 3 de 4 - Conteúdo até aula 05

Enunciado: Você foi contratado para desenvolver um sistema de Venda de uma Empresa Y que vende toras de arvore para outras empresas que vendem madeira. Você ficou com a parte de desenvolver a interface com o cliente.

A Empresa Y opera as vendas da seguinte maneira:

- Tora de Pinho (PIN), o valor do metro cúbico (m^3) é de cento e cinquenta reais e quarenta centavos;
 - Tora de Peroba (PER), o valor do metro cúbico (m^3) é de cento e setenta reais e vinte centavos;
 - Tora de Mogno (MOG), o valor do metro cúbico (m^3) é de cento e noventa reais e noventa centavos;
 - Tora de Ipê (IPE), o valor do metro cúbico (m^3) é de duzentos e dez reais e dez centavos;
 - Tora de Imbuia (IMB), o valor do metro cúbico (m^3) é de duzentos e vinte reais e setenta centavos;
-
- Se a quantidade (em m^3) de toras for **menor** que 100 não há desconto na venda (0/100);
 - Se a quantidade (em m^3) de toras for **igual ou maior** que 100 e **menor** que 500, o desconto será de 4% (4/100);
 - Se a quantidade (em m^3) de toras for **igual ou maior** que 500 e **menor** que 1000, o desconto será de 9% (9/100);
 - Se a quantidade (em m^3) de toras for **igual ou maior** que 1000 e **menor ou igual** que 2000, o desconto será de 16% (16/100);
 - Se a quantidade (em m^3) de toras for **maior** que 2000, não é aceito pedidos com essa quantidade de toras;
-
- ◆ Para o **adicional** de transporte rodoviário (1) é cobrado um valor **extra** de 1000 reais;
 - ◆ Para o **adicional** de transporte ferroviário (2) é cobrado um valor **extra** de 2000 reais;
 - ◆ Para o **adicional** de transporte hidroviário (3) é cobrado um valor **extra** de 2500 reais;

O valor final da conta é calculado da seguinte maneira:

$$\text{total} = ((\text{tipoMadeira} * \text{qtdToras}) * (1-\text{desconto})) + \text{transporte}$$

Elabore um programa em Python que:

- A. Deve-se implementar o print com uma mensagem de boas-vindas que apareça o seu **nome e sobrenome** (somente print, não usar input aqui). [EXIGÊNCIA DE CÓDIGO 1 de 7];
- B. Deve-se implementar a função **escolha_tipo()** que **não** recebe parâmetros e que: [EXIGÊNCIA DE CÓDIGO 2 de 7].
 - a. Pergunta o **tipo de madeira** desejado;
 - b. **Retorna o VALOR do tipo de madeira** com base na escolha do usuário (use **return**);
 - c. Repete a pergunta do item **B.a** se digitar uma opção diferente de: PIN/PER/MOG/IPE/IMB;
- C. Deve-se implementar a função **qtd_toras()** que **não** recebe parâmetros e que: [EXIGÊNCIA DE CÓDIGO 3 de 7].
 - a. Pergunta a **quantidade de toras**;
 - b. **Retorna (use return)** a **quantidade de toras E o valor do desconto** (os dois valores) seguindo a regra do enunciado;
 - c. Repete a pergunta do item **C.a** se digitar um valor acima de 2000 ou valor não numérico (use try/except para não numérico)
- D. Deve-se implementar a função **transporte()** que **não** recebe parâmetros e que: [EXIGÊNCIA DE CÓDIGO 4 de 7].
 - a. Pergunta pelo serviço **adicional de transporte**;

- b. Retorna (use `return`) o valor de apenas uma das opções de transporte;
 - c. Repetir a pergunta item D.a se digitar uma opção diferente de: 1/2/3;
- E. Deve-se implementar o total a pagar no código principal (`main`), ou seja, não pode estar dentro de função, conforme o enunciado [EXIGÊNCIA DE CÓDIGO 5 de 7];
- F. Deve-se implementar `try/except` [EXIGÊNCIA DE CÓDIGO 6 de 7];
- G. Deve-se inserir comentários relevantes no código [EXIGÊNCIA DE CÓDIGO 7 de 7];

Teste seu código atendendo as seguintes exigências:

- H. Deve-se apresentar na saída de console uma mensagem com o seu nome e sobrenome [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 4];
- I. Deve-se apresentar na saída de console um pedido no qual o usuário errou a opção de tipo de madeira [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 4];
- J. Deve-se apresentar na saída de console um pedido no qual o usuário digitou um valor que ultrapasse a quantidade máxima de toras aceitas (2000) [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 4];
- K. Deve-se apresentar na saída de console um pedido com opção de tipo de madeira, quantidade de toras e transporte válidos [EXIGÊNCIA DE SAÍDA DE CONSOLE 4 de 4];

EXEMPLO DE SAÍDA DE CONSOLE:

```
→ Bem vindo a Madeireira do Lenhador Bruno Kostiuk Nome completo  
Entre com o Tipo de Madeira desejado  
PIN - Tora de Pinho  
PER - Tora de Peroba  
MOG - Tora de Mogno  
IPE - Tora de Ipê  
IMB - Tora de Imbuia           Errou o tipo de Madeira  
>>TÁBUA  
Escolha inválida, entre com o modelo novamente  
  
Entre com o Tipo de Madeira desejado  
PIN - Tora de Pinho  
PER - Tora de Peroba  
MOG - Tora de Mogno  
IPE - Tora de Ipê  
IMB - Tora de Imbuia           Errou a quantidade de toras  
>>IPE  
Entre com a quantidade de toras (m³): 500000  
Não aceitamos pedidos com essa quantidade de toras.  
Por favor, entre com a quantidade novamente.  
  
Entre com a quantidade de toras (m³): 500  
  
Escolha o tipo de Transporte:  
1 - Transporte Rodoviário - R$ 1000.00  
2 - Transporte Ferroviário - R$ 2000.00  
3 - Transporte Hidroviário - R$ 2500.00  
>>3           Pedido com tipo de tora, quantidade de tora e  
Total: R$ 98095.50           transporte válidos
```

Figura 3: Exemplo de saída de console que o aluno deve fazer. Em que se pergunta pelo tipo de tora e se erra opção inicialmente, e que se passa a quantidade de toras acima do aceito. Na sequência, o usuário digitou um tipo de tora, quantidade de toras e transporte válidos.

Apresentação de Código da Questão 3:

```
def escolha_tipo():
    #Descrição de tipo de madeira
    print('Entre com o tipo de Madeira desejada:')
    print('PIN - Tora de pinho')
    print('PER - Tora de Perola')
    print('MOG - Tora de Mogno')
    print('IPE - Tora de Ipe')
    print('IMB - Tora de Imbuia')

    tipoMadeira = 0
    while True:
        #Verificando se o tipo de madeira é valida
        tipo_Madeira = input('>> ').upper()
        if tipo_Madeira not in ['PIN', 'PER', 'MOG', 'IPE', 'IMB']:
            print('Escolha inválida, entre com o modelo novamente.')
            return escolha_tipo()
        #Valores de cada tipo de madeira
        if tipo_Madeira == 'PIN':
            tipoMadeira = 150.40
        elif tipo_Madeira == 'PER':
            tipoMadeira = 170.20
        elif tipo_Madeira == 'MOG':
            tipoMadeira = 190.90
        elif tipo_Madeira == 'IPE':
            tipoMadeira = 210.10
        elif tipo_Madeira == 'IMB':
            tipoMadeira = 220.70
        return tipoMadeira

def qtd_toras():
    while True: #implementando um loop para que a entrada da quantidade seja validada
        try:
            quantidadeToras = float(input('Entre com a quantidade de toras (m³): '))
            desconto = 0 #Variável para armazenar o desconto
            #Decisão para verificação de quantidade de toras e aplicando o desconto
            if 1 <= quantidadeToras < 100:
                desconto = 0 #não tem desconto
            elif 100 <= quantidadeToras < 500:
                desconto = 4 / 100 #desconto de 4%
            elif 500 <= quantidadeToras < 1000:
                desconto = 9 / 100 #desconto de 9%
            elif 1000 <= quantidadeToras <= 2000:
                desconto = 16 / 100 # desconto de 16%
            else:
                print('Não aceitamos pedido com essa quantidade de toras.')
                print('Por favor, entre com a quantidade novamente.')
                return qtd_toras() #retorna para o inicio se a quantidade não for validade
            return quantidadeToras, desconto
        except ValueError:
            print('Erro no sistema')

def transporte():
    #Menu para escolha do transporte
    print('Escolha o tipo de Transporte:')
    print('1 - Transporte Rodoviário - R$ 1000.00')
    print('2 - Transporte Ferroviário - R$ 2000.00')
    print('3 - Transporte Hidroviário - R$ 2500.00')

    while True:
        tipoTransporte = int(input('>> '))
        if tipoTransporte == 1:
            return 1000 #custo do transporte
```

```
if tipoTransporte == 2:
    return 2000
if tipoTransporte == 3:
    return 2500
else:
    print('opção inválida, Tente novamente.')
    return transporte() #retorna para o inicio se a opção for inválida

print('Bem vindo a Madeireira do Lenhador Simei Pereira Moura')#descrição de boas vindas
print()
#chamando as funções
tipoMadeira = escolha_tipo()
quantidadeToras, desconto = qtd_toras()
transporte = transporte()

#calculando o total
total = (tipoMadeira * quantidadeToras) * (1 - desconto) + transporte
print()
print(f'Total: R$ {total:.2f}')
```

Apresentação de Saída do Console da Questão 3:

```
"C:\Users\PC\PycharmProjects\Trabalho - Log_Prog_Algoritmo\.venv\Scripts\python.exe" "C:\Users\PC\PycharmProjects\Trabalho - Log_Prog_Algoritmo\Questao3.py"
Bem vindo a Madeireira do Lenhador Simei Pereira Moura

Entre com o tipo de Madeira desejada:
PIN - Tora de pinho
PER - Tora de Perola
MOG - Tora de Mogno
IPE - Tora de Ipe
IMB - Tora de Imbuia
>> TABUA
Escolha inválida, entre com o modelo novamente.
Entre com o tipo de Madeira desejada:
PIN - Tora de pinho
PER - Tora de Perola
MOG - Tora de Mogno
IPE - Tora de Ipe
IMB - Tora de Imbuia
>> IPE
Entre com a quantidade de toras (m³): 5000
Não aceitamos pedido com essa quantidade de toras.
Por favor, entre com a quantidade novamente.
Entre com a quantidade de toras (m³): 500
Escolha o tipo de Transporte:
1 - Transporte Rodoviário - R$ 1000.00
2 - Transporte Ferroviário - R$ 2000.00
3 - Transporte Hidroviário - R$ 2500.00
>> 3
Total: R$ 98095.50

Process finished with exit code 0
```

QUESTÃO 4 de 4 - Conteúdo até aula 06

Enunciado: Você e sua equipe de programadores foram contratados por uma pequena empresa para desenvolver um software de gerenciamento de Contatos Comerciais. Este software deve ter o seguinte menu e opções:

- 1) Cadastrar Contato
- 2) Consultar Contato
 1. Consultar Todos
 2. Consultar por Id
 3. Consultar por Atividade
 4. Retornar ao menu
- 3) Remover Contato
- 4) Encerrar Programa

Elabore um programa em Python que:

- A. Deve-se implementar o print com uma mensagem de boas-vindas que apareça o seu **nome e sobrenome** (somente print, não usar input aqui) [EXIGÊNCIA DE CÓDIGO 1 de 8];
- B. Deve-se implementar uma lista com o nome de **lista_contatos** e a variável **id_global** com valor igual ao número de seu RU [EXIGÊNCIA DE CÓDIGO 2 de 8];
- C. Deve-se implementar uma função chamada **cadastrar_contato(id)** que recebe **apenas id** como parâmetro e que: [EXIGÊNCIA DE CÓDIGO 3 de 8];
 - a. Pergunta **nome, atividade, telefone** do contato;
 - b. Armazena o **id** (este é fornecido via parâmetro da função), **nome, atividade, telefone** dentro de um dicionário;
 - c. **Copiar** o dicionário para dentro da **lista_contatos** (utilizar o **copy**);
- D. Deve-se implementar uma função chamada **consultar_contatos()** que **não** recebe parâmetros e que: [EXIGÊNCIA DE CÓDIGO 4 de 8];
 - a. Deve-se perguntar qual opção deseja (1. Consultar Todos / 2. Consultar por Id / 3. Consultar por Setor / 4. Retornar ao menu):
 - i. Se Consultar Todos, apresentar todos os contatos com todos os seus dados cadastrados;
 - ii. Se Consultar por Id, solicitar ao usuário que informe um id, e apresentar o contato **específico** (apenas 1) com todos os seus dados cadastrados;
 - iii. Se Consultar por Atividade, solicitar ao usuário que informe a atividade, e apresentar o(s) contato(s) que exercem aquela atividade com todos os seus dados cadastrados;
 - iv. Se Retornar ao menu, deve-se **retornar** ao menu principal (return);
 - v. Se Entrar com um valor diferente de 1, 2, 3 ou 4, printar "Opção inválida" e repetir a pergunta D.a.
 - vi. Enquanto o usuário não escolher a opção 4, o menu consultar contatos deve se repetir.
- E. Deve-se implementar uma função chamada **remover_contato()** em que: [EXIGÊNCIA DE CÓDIGO 5 de 8];
 - a. Deve-se perguntar pelo **id** do contato a ser removido;
 - b. Remover o contato da **lista_contatos**;
 - c. Se o id fornecido não for de um contato da lista, printar "**Id inválido**" e repetir a pergunta E.a.
- F. Deve-se implementar uma estrutura de menu no código principal (**main**), ou seja, **não pode estar dentro de função**, em que: [EXIGÊNCIA DE CÓDIGO 6 de 8];
 - a. Deve-se perguntar qual opção deseja (1. Cadastrar Contato / 2. Consultar Contato / 3. Remover Contato / 4. Encerrar Programa):

- i. Se Cadastrar Contato, chamar a função `cadastrar_contato (id_global)` e em seguida, **incrementar** em um `id_global`;
 - ii. Se Consultar Contato, chamar função `consultar_contato ()`;
 - iii. Se Remover Contato, chamar função `remover_contato ()`;
 - iv. Se Encerrar Programa, sair do menu (e com isso acabar a execução do código);
 - v. Se Entrar com um valor diferente de 1, 2, 3 ou 4, printar "Opção inválida" e repetir a pergunta **F.a.**
 - vi. Enquanto o usuário não escolher a opção 4, o menu deve se repetir.
- G. Deve-se implementar uma **lista de dicionários** (uma lista contendo dicionários dentro) [EXIGÊNCIA DE CÓDIGO 7 de 8];
- H. Deve-se inserir comentários relevantes no código [EXIGÊNCIA DE CÓDIGO 8 de 8];

Teste seu código atendendo as seguintes exigências:

- I. Deve-se apresentar na saída de console um cadastro do **seu contato** da seguinte forma: para **nome** informe seu **nome completo** (não usar apelidos ou abreviações), para **atividade** informar como **estudante**, e para **telefone** informe sua **RU**. [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 6];
- J. Deve-se apresentar na saída de console um cadastro de **mais 2** contatos com mesmo tipo de atividade (por exemplo: marceneiro, padeiro, pintor, pedreiro) [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 6];
- K. Deve-se apresentar na saída de console uma consulta de todos os contatos [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 6];
- L. Deve-se apresentar na saída de console uma consulta por código (id) de um dos contados [EXIGÊNCIA DE SAÍDA DE CONSOLE 4 de 6];
- M. Deve-se apresentar na saída de console uma consulta por atividade em que **2** contatos exerçam a mesma atividade [EXIGÊNCIA DE SAÍDA DE CONSOLE 5 de 6];
- N. Deve-se apresentar na saída de console uma remoção de um dos contatos e em seguida de uma consulta de todos os contatos, provando que o contato foi removido [EXIGÊNCIA DE SAÍDA DE CONSOLE 6 de 6];

EXEMPLO DE SAÍDA DE CONSOLE:

Bem vindo a Lista de Contatos do Bruno Kostiuk Nome completo

----- MENU PRINCIPAL -----

Escolha a opção desejada:

1 - Cadastrar Contato
2 - Consultar Contato(s)
3 - Remover Contato
4 - Sair

>>1

----- MENU CADASTRAR CONTATO -----

Id do Contato: 4297914

Por favor entre com o nome do Contato: Bruno Kostiuk

Por favor entre com a Atividade do contato: Estudante

Por favor entre com o telefone do contato: 4297913

Cadastro do primeiro contato, com seu nome completo, atividade estudante e telefone igual ao seu RU

Figura 4.1: Exemplo de saída de console que o aluno deve fazer. Apresenta o print com seu nome completo e é realizado o cadastro do primeiro contato, note que o ID do contato não inicia em 1, pois ele deve iniciar com o seu RU (caso o RU informado não seja o seu, irá receber zero em toda questão). O primeiro contato deve ser cadastrado com SEU NOME COMPLETO, em Atividade informe Estudante e em Contato informe o SEU RU.

----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Contato
2 - Consultar Contato(s)
3 - Remover Contato
4 - Sair
>>1

----- MENU CADASTRAR CONTATO -----
Id do Contato: 4297915
Por favor entre com o nome do Contato: Tamy
Por favor entre com a Atividade do contato: Professor
Por favor entre com o telefone do contato: 9999988888

----- MENU PRINCIPAL -----
Escolha a opção desejada:
1 - Cadastrar Contato
2 - Consultar Contato(s)
3 - Remover Contato
4 - Sair
>>1

----- MENU CADASTRAR CONTATO -----
Id do Contato: 4297916
Por favor entre com o nome do Contato: Osmar
Por favor entre com a Atividade do contato: Professor
Por favor entre com o telefone do contato: 888899999

Figura 4.2: Exemplo de saída de console que o aluno deve fazer. São cadastrados mais dois contatos com mesmo tipo de Atividade.

```

----- MENU CONSULTAR CONTATOS -----
Escolha a opção desejada:
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
>>1
-----
id: 4297914
nome: Bruno Kostiuk
atividade: Estudante
telefone: 4297913

id: 4297915                               Consulta todos os
nome: Tamy                                contatos cadastrados
atividade: Professor
telefone: 99998888

id: 4297916
nome: Osmar
atividade: Professor
telefone: 88889999
-----
```

Figura 4.3: Exemplo de saída de console que o aluno deve fazer. Em que se consulta Todos os contatos cadastrados.

```

----- Escolha a opção desejada: -----
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
>>2
Digite o id do contato: 4297914
-----
id: 4297914                               Consulta por id
nome: Bruno Kostiuk
atividade: Estudante
telefone: 4297913
-----
----- MENU CONSULTAR CONTATOS -----
Escolha a opção desejada:
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
>>3
Digite a Atividade do(s) Contato(s): Professor
-----
id: 4297915
nome: Tamy
atividade: Professor
telefone: 99998888

id: 4297916                               Consulta por Atividade
nome: Osmar
atividade: Professor
telefone: 88889999
-----
```

Figura 4.4: Exemplo de saída de console que o aluno deve fazer. Em que se consulta o contato com id número 4297914 e consulta pelo nome da Atividade (Professor).

```
-----  
----- MENU PRINCIPAL -----  
Escolha a opção desejada:  
1 - Cadastrar Contato  
2 - Consultar Contato(s)  
3 - Remover Contato  
4 - Sair  
>>3  
-----  
----- MENU REMOVER CONTATO -----  
Digite o id do contato a ser removido: 4297914  
Contato removido com sucesso!  
-----  
----- MENU PRINCIPAL -----  
Escolha a opção desejada:  
1 - Cadastrar Contato  
2 - Consultar Contato(s)  
3 - Remover Contato  
4 - Sair  
>>2  
-----  
----- MENU CONSULTAR CONTATOS -----  
Escolha a opção desejada:  
1 - Consultar Todos os Contatos  
2 - Consultar Contato por id  
3 - Consultar Contato(s) por Atividade  
4 - Retornar  
>>1  
-----  
id: 4297915  
nome: Tamy  
atividade: Professor  
telefone: 99998888  
-----  
id: 4297916  
nome: Osmar  
atividade: Professor  
telefone: 88889999  
-----
```

Remove um contato

Realiza o consultar Todos mostrando
que o contato foi removido

Figura 4.5: Exemplo de saída de console que o aluno deve fazer. Em que se remove o contato de Id número 4297914 e depois se faz uma consulta de todos os contatos.

Apresentação de Código da Questão 4:

```
#Lista para armazenar os dados e variável global
lista_contatos = []
id_global = 4297913

#função de cadastro
def cadastrar_contato(id):
    print('-----')
    print('-----MENU CADASTRAR CONTATO-----')
    print(f'Id do Contato: {id}')
    nome = input('Por favor entre com o nome do Contato: ')
    atividade = input('Por favor entre com a Atividade do contato: ')
    telefone = input('Por favor entre com o telefone do contato: ')
    #em seguida os dados e armazenado em um dicionário.
    dicionario_contatos = {'id': id, 'nome': nome, 'Atividade': atividade, 'telefone': telefone}
    #adicionamos uma copia da lista de contatos
    lista_contatos.append(dicionario_contatos.copy())

#função de consulta contatos
def consultar_contatos():
    print('-----')
    print('-----MENU CONSULTAR CONTATOS-----')
    while True:
        opcaoConsulta = int(input('Escolha a opção desejada: \n' +
                                  '1 - Consultar Todos os Contatos \n' +
                                  '2 - Consultar Contato por id \n' +
                                  '3 - Consultar Contato(s) por Atividade \n' +
                                  '4 - Retornar \n' +
                                  '>> '))
        if opcaoConsulta == 1:
            for contato in lista_contatos: #aqui adicionamos cada item da lista
                for id, dados in contato.items():
                    print(f'{id}: {dados}')
            print()
        elif opcaoConsulta == 2:
            id_desejado = int(input('Digite o id do contato: '))
            for contato in lista_contatos:
                if contato['id'] == id_desejado:
                    print('-----')
                    for id, dados in contato.items():
                        print(f'{id}: {dados}')
                    print('-----')
        elif opcaoConsulta == 3:
            id_desejado = input('Digite a Atividade do(s) Contato(s): ')
            for contato in lista_contatos:
                if contato['Atividade'] == id_desejado:
                    print('-----')
                    for id, dados in contato.items():
                        print(f'{id}: {dados}')
                    print('-----')
        elif opcaoConsulta == 4:
            return #sai da função e retorna para o código principal
        else: #se digitar opção diferente
            print('Opção inválida.')
            continue #enquanto não digitar a opção correta, vai ficar retornando

#função remover contato
def remover_contato():
    print('-----')
    print('-----MENU REMOVER CONTATO-----')
    id_desejado = int(input('Digite o id do contato a ser removido: '))
    for contato in lista_contatos:
```

```
if contato ['id'] == id_desejado:
    lista_contatos.remove(contato)
    print('Contato removido com sucesso!')
print('-' *80)
print('Bem vindo a Lista de Contatos do Simei Pereira Moura')
while True:
    print('-----')
    print('-----MENU PRINCIPAL-----')
    opcao_menu = int(input('Escolha a opção desejada: \n' +
                           '1 - Cadastrar Contato \n' +
                           '2 - Consultar Contato(s) \n' +
                           '3 - Remover Contato \n' +
                           '4 - Sair \n' +
                           '>> ')
    )
    if opcao_menu == 1:
        id_global += 1 #adiciona mais um contato
        cadastrar_contato(id_global)
    elif opcao_menu == 2:
        consultar_contatos()
    elif opcao_menu == 3:
        remover_contato()
    elif opcao_menu == 4:
        print('Encerrando Sistema.....')
        break # encerra aqui o programa
    else:
        print('Opção inválida.')
        continue
```

Apresentação de Saída do Console da Questão 4:

```
"C:\Users\PC\PycharmProjects\Trabalho - Log_Prog_Algoritmo\.venv\Scripts\python.exe" "C:\Users\PC\PycharmProjects\Trabalho - Log_Prog_Algoritmo\Questao4.py"
-----
Bem vindo a Lista de Contatos do Simei Pereira Moura
-----
-----MENU PRINCIPAL-----
Escolha a opção desejada:
1 - Cadastrar Contato
2 - Consultar Contato(s)
3 - Remover Contato
4 - Sair
>> 1
-----
-----MENU CADASTRAR CONTATO-----
Id do Contato: 4297914
Por favor entre com o nome do Contato: Simei Pereira Moura
Por favor entre com a Atividade do contato: Estudante
Por favor entre com o telefone do contato: 5489190
-----
-----MENU PRINCIPAL-----
Escolha a opção desejada:
1 - Cadastrar Contato
2 - Consultar Contato(s)
3 - Remover Contato
4 - Sair
>> 1
-----
-----MENU CADASTRAR CONTATO-----
Id do Contato: 4297915
Por favor entre com o nome do Contato: Tamy
Por favor entre com a Atividade do contato: Professor
Por favor entre com o telefone do contato: 99998888
-----
-----MENU PRINCIPAL-----
Escolha a opção desejada:
1 - Cadastrar Contato
2 - Consultar Contato(s)
3 - Remover Contato
4 - Sair
>> 1
-----
-----MENU CADASTRAR CONTATO-----
Id do Contato: 4297916
Por favor entre com o nome do Contato: Osmar
Por favor entre com a Atividade do contato: Professor
Por favor entre com o telefone do contato: 88889999
-----
-----MENU PRINCIPAL-----
Escolha a opção desejada:
1 - Cadastrar Contato
2 - Consultar Contato(s)
3 - Remover Contato
4 - Sair
>> 2
-----
-----MENU CONSULTAR CONTATOS-----
Escolha a opção desejada:
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
>> 1
id: 4297914
nome: Simei Pereira Moura
Atividade: Estudante
telefone: 5489190

id: 4297915
nome: Tamy
Atividade: Professor
telefone: 99998888
```

```
id: 4297916
nome: Osmar
Atividade: Professor
telefone: 88889999
```

```
Escolha a opção desejada:
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
>> |
```

```
Escolha a opção desejada:
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
```

```
>> 2
Digite o id do contato: 4297914
```

```
-----  
id: 4297914
nome: Simei Pereira Moura
Atividade: Estudante
telefone: 5489190
```

```
-----  
Escolha a opção desejada:
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
```

```
>> |
```

```
Escolha a opção desejada:
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
```

```
>> 3
Digite a Atividade do(s) Contato(s): Professor
```

```
-----  
id: 4297915
nome: Tamy
Atividade: Professor
telefone: 99998888
```

```
-----  
id: 4297916
nome: Osmar
Atividade: Professor
telefone: 88889999
```

```
-----  
Escolha a opção desejada:
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
```

```
>> |
```

```
Escolha a opção desejada:
1 - Consultar Todos os Contatos
2 - Consultar Contato por id
3 - Consultar Contato(s) por Atividade
4 - Retornar
```

```
>> 4
```

```
-----MENU PRINCIPAL-----
```

```
Escolha a opção desejada:
1 - Cadastrar Contato
2 - Consultar Contato(s)
3 - Remover Contato
4 - Sair
```

```
>> |
```

-----MENU PRINCIPAL-----

Escolha a opção desejada:

1 - Cadastrar Contato

2 - Consultar Contato(s)

3 - Remover Contato

4 - Sair

>> 3

-----MENU REMOVER CONTATO-----

Digite o id do contato a ser removido: 4297914

Contato removido com sucesso!

-----MENU PRINCIPAL-----

Escolha a opção desejada:

1 - Cadastrar Contato

2 - Consultar Contato(s)

3 - Remover Contato

4 - Sair

>> |

-----MENU PRINCIPAL-----

Escolha a opção desejada:

1 - Cadastrar Contato

2 - Consultar Contato(s)

3 - Remover Contato

4 - Sair

>> 2

-----MENU CONSULTAR CONTATOS-----

Escolha a opção desejada:

1 - Consultar Todos os Contatos

2 - Consultar Contato por id

3 - Consultar Contato(s) por Atividade

4 - Retornar

>> 1

id: 4297915

nome: Tamy

Atividade: Professor

telefone: 99998888

id: 4297916

nome: Osmar

Atividade: Professor

telefone: 88889999

Escolha a opção desejada:

1 - Consultar Todos os Contatos

2 - Consultar Contato por id

3 - Consultar Contato(s) por Atividade

4 - Retornar

>> 4

-----MENU PRINCIPAL-----

Escolha a opção desejada:

1 - Cadastrar Contato

2 - Consultar Contato(s)

3 - Remover Contato

4 - Sair

>> 4

Encerrando Sistema.....

Process finished with exit code 0

|

