

# QUESTÃO 1 de 4 - Conteúdo até Aula 03

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:

$$valorMensal=valorBase*porcentagem$$

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

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

Elabore um programa em Python que:

- A. Deve-se implementar o **print** com o seu **nome completo** (somente print, não usar input aqui).  
Por exemplo: **print("Sistema desenvolvido por Bruno KostiuK")** [EXIGÊNCIA DE CÓDIGO 1 de 6];
- B. Deve-se implementar o input do **valorBase** do plano e da **idade** do cliente [EXIGÊNCIA DE CÓDIGO 2 de 6];
- C. 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];
- D. Deve-se implementar o **valorMensal** [EXIGÊNCIA DE CÓDIGO 4 de 6];
- E. Deve-se implementar as estruturas **if, elif e else (todas elas)** [EXIGÊNCIA DE CÓDIGO 5 de 6];
- F. Deve-se inserir comentários **relevantes** no código [EXIGÊNCIA DE CÓDIGO 6 de 6];
- G. Deve-se apresentar na saída de console uma mensagem com seu nome completo [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 2];
- H. 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 1.1:** 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('Sistema desenvolvido por Bruno Machion Lobo')

plb = int(input('Informe o valor do Plano Base do cliente: '))
idade = int(input('Informe a idade do cliente: '))
#Estrutura If de acordo com a tabela de idade fornecida
if(idade>=0 and idade<19):
    perc = 1 #100%
elif(idade>=19 and idade<29):
    perc = 1.5 #150%
elif(idade>=29 and idade<39):
    perc = 2.25 #225%
elif(idade>=39 and idade<49):
    perc = 2.4 #240%
elif(idade>=49 and idade<59):
    perc = 3.5 #350%
elif(idade>=59):
    perc = 6 #600%
total = plb*perc
#formatação de número para utilizar 2 casas após a vírgula
print(f'O valor mensal do Plano é R${total:.2f}')
```

Apresentação de **Saída do Console da Questão 1:**

```
⇒ Sistema desenvolvido por Bruno Machion Lobo
Informe o valor do Plano Base do cliente: 500
Informe a idade do cliente: 35
O valor mensal do Plano é R$1125.00
```

## 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:

- A. Deve-se implementar o **print** com o seu **nome completo** (somente print, não usar input aqui).  
Por exemplo: **print("Bem-vindos a Pizzaria do Bruno KostiuK")**  
Além do seu nome completo, deve-se implementar um **print com um Menu** para o cliente. [EXIGÊNCIA DE CÓDIGO 1 de 8];
- B. 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];
- C. 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];
- D. 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];
- E. Deve-se implementar um **acumulador** para somar os valores dos pedidos (valor total do pedido) [EXIGÊNCIA DE CÓDIGO 5 de 8];
- F. 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];
- G. Deve-se implementar as estruturas de **while, break, continue (todas elas)** [EXIGÊNCIA DE CÓDIGO 7 de 8];
- H. Deve-se inserir comentários **relevantes** no código [EXIGÊNCIA DE CÓDIGO 8 de 8];
- I. Deve-se apresentar na saída de console uma mensagem com o seu nome completo e o menu para o cliente conhecer as opções [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 4];
- J. Deve-se apresentar na saída de console um pedido em que o usuário errou o **sabor** [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 4];
- K. Deve-se apresentar na saída de console um pedido em que o usuário errou o **tamanho** [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 4];
- L. 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
Sabor inválido. Tente novamente

Entre com o sabor desejado (PS/PD): PS
Entre com o tamanho desejado (P/M/G): EXGG
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 uma Pizza Doce 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)

Usuário errou o sabor

Usuário errou o tamanho

Pedido com 2 itens de tamanho e sabores diferentes

Figura 2.1: Exemplo de saída de console que o aluno deve fazer. Em que se perguntar o sabor e o tamanho. Há uma tentativa de pedido que se errou o sabor e outra que se errou o tamanho. Há também um pedido com dois itens com sabores e tamanhos diferentes.



# Apresentação de *Código da Questão 2:*

```
print('----- Bem vindo(a) à Pizzaria de Bruno Machion Lobo -----')

print('----- Cardápio -----')
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('-----')

#estrutura while para escolha do sabor
def escolha_sabor():
    while True:
        sabor = input('Escolha o sabor de sua pizza (PS ou PD): ').upper()
        if (sabor in ['PS','PD']):
            break
        else:
            print('Tamanho inválido. Tente novamente.')
    return sabor

#estrutura while para escolha do tamanho
def escolha_tamanho():
    while True:
        tamanho = input('Escolha o tamanho de sua pizza (P,M,G): ').upper()
        if (tamanho in ['P','M','G']):
            break
        else:
            print('Tamanho inválido. Tente novamente.')
    return tamanho

def calculo_preco(tamanho,sabor):
#Condicionais tamanho P
    if(tamanho == 'P' and sabor == 'PS'):
        print('Você pediu uma Pizza Salgada Tamanho P: R$30,00')
        return 30
    elif(tamanho == 'P' and sabor == 'PD'):
        print('Você pediu uma Pizza Doce Tamanho P: R$34,00')
        return 34
#Condicionais tamanho M
    elif(tamanho == 'M' and sabor == 'PS'):
        print('Você pediu uma Pizza Salgada Tamanho M: R$45,00')
        return 45
    elif(tamanho == 'M' and sabor == 'PD'):
        print('Você pediu uma Pizza Doce Tamanho M: R$48,00')
        return 48
#Condicionais tamanho G
    elif(tamanho == 'G' and sabor == 'PS'):
        print('Você pediu uma Pizza Salgada Tamanho G: R$60,00')
        return 60
    elif(tamanho == 'G' and sabor == 'PD'):
        print('Você pediu uma Pizza Doce Tamanho G: R$66,00')
        return 66

#Valor primeiro pedido
sabor = escolha_sabor()
tamanho = escolha_tamanho()
preco = calculo_preco(tamanho, sabor)

#Segundo pedido
while True:
    pedido2 = input('Deseja pedir mais alguma coisa (S/N): ').upper()
    if(pedido2=='S'):
        sabor= escolha_sabor()
        tamanho= escolha_tamanho()
        preco += calculo_preco(tamanho, sabor)
    elif(pedido2=='N'):
        break
    else:
        print('Opção inválida. Tente novamente.')
        continue

print(f'O valor total do seu pedido é R${preco:.2f}')
```

Apresentação de **Saída do Console da Questão 2:**

```
-----Bem vindo(a) à Pizzaria de Bruno Machion Lobo-----
-----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    |---
-----
Escolha o sabor de sua pizza (PS ou PD): ps
Escolha o tamanho de sua pizza (P,M,G): p
Você pediu uma Pizza Salgada Tamanho P: R$30,00
Deseja pedir mais alguma coisa (S/N): s
Escolha o sabor de sua pizza (PS ou PD): pd
Escolha o tamanho de sua pizza (P,M,G): m
Você pediu uma Pizza Doce Tamanho M: R$48,00
Deseja pedir mais alguma coisa (S/N): n
O valor total do seu pedido é R$78.00
```

# 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 (m³) é de cento e cinquenta reais e quarenta centavos;
  - Tora de Peroba (PER), o valor do metro cúbico (m³) é de cento e setenta reais e vinte centavos;
  - Tora de Mogno (MOG), o valor do metro cúbico (m³) é de cento e noventa reais e noventa centavos;
  - Tora de Ipê (IPE), o valor do metro cúbico (m³) é de duzentos e dez reais e dez centavos;
  - Tora de Imbuia (IMB), o valor do metro cúbico (m³) é de duzentos e vinte reais e setenta centavos;
- 
- Se a quantidade (em m³) de toras for **menor** que 100 não há desconto na venda (0/100);
  - Se a quantidade (em m³) de toras for **igual ou maior** que 100 e **menor** que 500, o desconto será de 4% (4/100);
  - Se a quantidade (em m³) de toras for **igual ou maior** que 500 e **menor** que 1000, o desconto será de 9% (9/100);
  - Se a quantidade (em m³) 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 m³) 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 o seu **nome completo** (somente print, não usar input aqui).  
Por exemplo: **print("Bem-vindos a Madeireira do Lenhador Bruno KostiuK")** [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];
- H. Deve-se apresentar na saída de console uma mensagem com o seu nome completo [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 3.1: 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:

```
print('Bem vindos a madeireira do lenhador Bruno Machion Lobo')

#Função para a escolha da madeira
def escolha_tipo():
    while True:
        print('Selecione o tipo de madeira desejado: ')
        print('PIN - Tora de Pinho')
        print('PER - Tora de Peroba')
        print('MOG - Tora de Mogno')
        print('IPE - Tora de Ipê')
        print('IMB - Tora de Imbuia')
        tipo=input('>>').upper()
        #Condicionais para o tipo de madeira
        if(tipo=='PIN'):
            return 150.40

        elif(tipo=='PER'):
            return 170.20

        elif(tipo=='MOG'):
            return 190.90

        elif(tipo=='IPE'):
            return 210.10

        elif(tipo=='IMB'):
            return 220.70

        else:
            print('Escolha inválida. Tente novamente.')

#Função para definição da quantidade a ser comprada
def qtd_toras():
    while True:
        try:
            qtd=int(input(('Selecione a quantidade de toras (m³): ')))
            if(qtd<100):
                return qtd, 0

            elif(100 <= qtd < 500):
                return qtd, 0.04

            elif(500 <= qtd <1000):
                return qtd, 0.09

            elif(1000 <= qtd <=2000):
                return qtd, 0.16

            else:
                print('Não aceitamos pedidos com essa quantidade de toras.')
        except ValueError:
            print('Quantidade inválida. Por favor, tente novamente.')

#Função para definição do transporte
def transporte():
    while True:
        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')
        transp=int(input('>>'))
        if(transp==1):
            return 1000

        elif(transp==2):
            return 2000

        elif(transp==3):
            return 2500

        else:
            print('Opção inválida. Tente novamente.')

tmad= escolha_tipo()

quant, desconto = qtd_toras()

trmad= transporte()

total = ((tmad*quant)*(1-desconto))+trmad

print(f'O total da sua compra é de R${total:.2f}')
```

*Apresentação de Saída do Console da Questão 3:*

```
Bem vindos a madeireira do lenhador Bruno Machion Lobo
Selecione 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
>>das
Escolha inválida. Tente novamente.
Selecione 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
>>pin
Selecione a quantidade de toras (m³): 2.65
Quantidade inválida. Por favor, tente novamente.
Selecione a quantidade de toras (m³): 5000
Não aceitamos pedidos com essa quantidade de toras.
Selecione a quantidade de toras (m³): 1500
Escolha o tipo de transporte:
1 - Transporte Rodoviário - R$1000,00
2 - Transporte Ferroviário - R$2000,00
3 - Transporte Hidroviário - R$2500,00
>>3
0 total da sua compra é de R$192004.00
```

# 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 o seu **nome completo** (somente print, não usar input aqui).  
Por exemplo: **print("Bem vindos a lista de contatos do Bruno Kostiuk")** [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 inicial 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 pergunta 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 pergunta qual opção deseja (1. Cadastrar Contato / 2. Consultar Contato / 3. Remover Contato / 4. Encerrar Programa):
    - i. Se Cadastrar Contato, **incrementar** em um **id\_global** e **em seguida**, chamar a função **cadastrar\_contato (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 contento dicionários dentro) [EXIGÊNCIA DE CÓDIGO 7 de 8];
- H. Deve-se inserir comentários **relevantes** no código [EXIGÊNCIA DE CÓDIGO 8 de 8];
- 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
-----
```

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: 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
-----
```

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  
nome: Tamy  
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  
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  
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  
                                     Remove um contato  
>>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  
                                     Realiza o consultar Todos mostrando  
                                     que o contato foi removido  
  
id: 4297916  
nome: Osmar  
atividade: Professor  
telefone: 88889999  
-----
```

**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:*

*id\_global* = 5047764

lista\_contatos = []

```
def cadastrar_contato(id):
    print('-'*15,'Menu Cadastrar Contato','-'*14)
    nome=input('Por favor informe o nome do contato: ')
    atividade=input('Por favor informe a atividade do contato: ')
    telefone=int(input('Por favor informe o telefone do contato: '))
    contato={"id": id, "nome": nome, "atividade":atividade, "telefone":telefone}
    lista_contatos.append(contato.copy())
```

```
def consultar_contatos():
    while True:
        print('-'*14,'Menu Consultar Contato','-'*14)
        print('Escolha a opção desejada:')
        print('1 - Consultar Todos')
        print('2 - Consultar por Id')
        print('3 - Consultar por Atividade')
        print('4 - Retornar ao Menu Principal')
```

```
    try:
        consulta=int(input('>> '))
    except ValueError:
        print('Por favor, insira um número válido.')
```

```
    if(consulta==1):
        if not lista_contatos:
            print('Nenhum contato cadastrado')
        for contato in lista_contatos:
            print('Id:', contato ["id"])
            print('Nome:', contato ["nome"])
            print('Atividade:', contato ["atividade"])
            print('Telefone', contato ['telefone'])
            print()
```

```
    elif(consulta==2):
        try:
            id_contato=int(input('Informe o Id: '))
            encontrado = False
            for contato in lista_contatos:
                if contato ["id"] == id_contato:
                    print('Id:', contato ["id"])
                    print('Nome:', contato ["nome"])
                    print('Atividade:', contato ["atividade"])
                    print('Telefone', contato ['telefone'])
                    encontrado = True
                    break
            if not encontrado:
                print('Id não encontrado. Tente novamente')
        except ValueError:
            print('Por favor, digite um número válido.')
```

```
    elif(consulta==3):
        atividade_consulta=input('Informe a Atividade do Contato: ')
        encontrados= [contato for contato in lista_contatos
                        if contato ["atividade"] == atividade_consulta]
        if(encontrados):
            for contato in encontrados:
                print('Id:', contato["id"])
                print('Nome:', contato["nome"])
                print('Atividade:', contato["atividade"])
                print('Telefone:', contato['telefone'])
                print()
        else:
            print('Nenhum contato com essa atividade foi encontrado.')
```

```
    elif(consulta==4):
        break
    else:
        print('Opção Inválida. Tente novamente.')
```

```
def remover_contato():
    while True:
        print('-'*15,'Menu Remover Contato','-'*15)
        remove=int(input('Informe o número de Id a ser removido: '))
        for contato in lista_contatos:
            if(contato ["id"] == remove):
                lista_contatos.remove(contato)
                print('O contato foi removido com sucesso.')
                return
        print('Id inválido. Tente novamente')
```

```
while True:
    print('Bem vindos a lista de contatos de Bruno Machion Lobo')
```

```
print('-'*52)
print('-'*18,'Menu Principal', '-'*18)
print('1 - Cadastrar Contato')
print('2 - Consultar Contato')
print('3 - Remover Contato')
print('4 - Sair')
print('Escolha a opção desejada:')
menu=int(input('>> '))
if(menu==1):
    cadastrar_contato(id_global)
    id_global +=1
elif(menu==2):
    consultar_contatos()
elif(menu==3):
    remover_contato()
elif(menu==4):
    print('Programa Encerrado')
    break
else:
    print('Opção inválida. Tente novamente.')
```

Apresentação de **Saída do Console da Questão 4:**

```
Bem vindos a lista de contatos de Bruno Machion Lobo
-----
----- Menu Principal -----
1 - Cadastrar Contato
2 - Consultar Contato
3 - Remover Contato
4 - Sair
Escolha a opção desejada:
>> 1
----- Menu Cadastrar Contato -----
Por favor informe o nome do contato: Bruno Machion Lobo
Por favor informe a atividade do contato: Estudante
Por favor informe o telefone do contato: 5047764
Bem vindos a lista de contatos de Bruno Machion Lobo
-----
----- Menu Principal -----
1 - Cadastrar Contato
2 - Consultar Contato
3 - Remover Contato
4 - Sair
Escolha a opção desejada:
>> 1
----- Menu Cadastrar Contato -----
Por favor informe o nome do contato: Thiago
Por favor informe a atividade do contato: Professor
Por favor informe o telefone do contato: 1154545454
Bem vindos a lista de contatos de Bruno Machion Lobo
-----
```

```
----- Menu Principal -----
1 - Cadastrar Contato
2 - Consultar Contato
3 - Remover Contato
4 - Sair
Escolha a opção desejada:
>> 1
----- Menu Cadastrar Contato -----
Por favor informe o nome do contato: Thomas
Por favor informe a atividade do contato: Professor
Por favor informe o telefone do contato: 1164646464
Bem vindos a lista de contatos de Bruno Machion Lobo
-----
----- Menu Principal -----
1 - Cadastrar Contato
2 - Consultar Contato
3 - Remover Contato
4 - Sair
Escolha a opção desejada:
>> 2
----- Menu Consultar Contato -----
Escolha a opção desejada:
1 - Consultar Todos
2 - Consultar por Id
3 - Consultar por Atividade
4 - Retornar ao Menu Principal
>> 1
Id: 5047764
Nome: Bruno Machion Lobo
Atividade: Estudante
Telefone 5047764

Id: 5047765
Nome: Thiago
Atividade: Professor
Telefone 1154545454

Id: 5047766
Nome: Thomas
Atividade: Professor
Telefone 1164646464
```

```
----- Menu Consultar Contato -----
Escolha a opção desejada:
1 - Consultar Todos
2 - Consultar por Id
3 - Consultar por Atividade
4 - Retornar ao Menu Principal
>> 2
Informe o Id: 5047765
Id: 5047765
Nome: Thiago
Atividade: Professor
Telefone 1154545454
----- Menu Consultar Contato -----
Escolha a opção desejada:
1 - Consultar Todos
2 - Consultar por Id
3 - Consultar por Atividade
4 - Retornar ao Menu Principal
>> 3
Informe a Atividade do Contato: Professor
Id: 5047765
Nome: Thiago
Atividade: Professor
Telefone: 1154545454

Id: 5047766
Nome: Thomas
Atividade: Professor
Telefone: 1164646464

----- Menu Consultar Contato -----
Escolha a opção desejada:
1 - Consultar Todos
2 - Consultar por Id
3 - Consultar por Atividade
4 - Retornar ao Menu Principal
>> 4
```

```
----- Menu Principal -----
1 - Cadastrar Contato
2 - Consultar Contato
3 - Remover Contato
4 - Sair
Escolha a opção desejada:
>> 3
----- Menu Remover Contato -----
Informe o número de Id a ser removido: 5047766
O contato foi removido com sucesso.
Bem vindos a lista de contatos de Bruno Machion Lobo
-----
----- Menu Principal -----
1 - Cadastrar Contato
2 - Consultar Contato
3 - Remover Contato
4 - Sair
Escolha a opção desejada:
>> 2
----- Menu Consultar Contato -----
Escolha a opção desejada:
1 - Consultar Todos
2 - Consultar por Id
3 - Consultar por Atividade
4 - Retornar ao Menu Principal
>> 1
Id: 5047764
Nome: Bruno Machion Lobo
Atividade: Estudante
Telefone 5047764

Id: 5047765
Nome: Thiago
Atividade: Professor
Telefone 1154545454

----- Menu Consultar Contato -----
Escolha a opção desejada:
1 - Consultar Todos
2 - Consultar por Id
3 - Consultar por Atividade
4 - Retornar ao Menu Principal
>> 4
```