QUESTÃO 1 de 4 - Conteúdo até Aula 03

Enunciado: Imagina-se que você é um dos programadores responsáveis pela construção de app de vendas para uma determinada empresa X que aceita cartões de crédito. Uma das estratégias de vendas dessa empresa X é cobrar um Juros maior conforme a quantidade de parcelas que o cliente desejar, conforme a **listagem abaixo**:

* Se a quantidade de parcelas for **menor** que **4**, o Juros será de **0%** (0 / 100);
* Se a quantidade de parcelas for **igual ou maior** que **4** e **menor que** **6**, o Juros será de **4%** (4 / 100);
* Se a quantidade de parcelas for **igual ou maior** que **6** e **menor que** **9**, o Juros será de **8%** (8 / 100);
* Se a quantidade de parcelas for **igual ou maior** que **9** e **menor que** **13**, o Juros será de **16%** (16 / 100);
* Se a quantidade de parcelas for **igual ou maior** que **13**, o Juros será de **32%** (32 / 100);

O valor da parcela é calculado da seguinte maneira:

O valor total parcelado é calculado da seguinte maneira:

Elabore um programa em Python que:

1. Deve-se implementar o **print** com o seu **nome completo** (somente print, não usar input aqui).

Por exemplo: **print(“Bem-vindos a loja do Bruno Kostiuk”)** [EXIGÊNCIA DE CÓDIGO 1 de 6];

1. Deve-se implementar o input do **valorDoPedido** e da **quantidadeParcelas** [EXIGÊNCIA DE CÓDIGO 2 de 6];
2. Deve-se implementar o Juros **conforme a enunciado acima** (obs.: atente-se as condições de menor, igual e maior) [EXIGÊNCIA DE CÓDIGO 3 de 6];
3. Deve-se implementar o **valorDaParcela** e **valorTotalParcelado** [EXIGÊNCIA DE CÓDIGO 4 de 6];
4. Deve-se implementar as estruturas **if, elif e else (todas elas)** [EXIGÊNCIA DE CÓDIGO 5 de 6];
5. Deve-se inserir comentários **relevantes** no código [EXIGÊNCIA DE CÓDIGO 6 de 6];
6. Deve-se apresentar na saída de console uma mensagem com seu nome completo [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 2];
7. Deve-se apresentar na saída de console um parcelamento com Juros (**quantidadeParcelas** maior ou igual a 4) apresentando o valor da Parcela e o valor Total Parcelado [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 2];

EXEMPLO DE SAÍDA DE CONSOLE:

A imagem apresenta a seguinte saída de código: 
Bem-vindo a Loja do Bruno Kostiuk
Entre com o valor do pedido: 1000.00
Entre com a quantidade de parcelas: 14
O valor das parcelas é de:R$ 94.29
O valor Total Parcelado é de:R$ 1320.00

**Figura 1.1: Exemplo de saída de console que o aluno deve fazer. Em que se perguntar o valor do pedido (pode ser qualquer valor) a quantidade de parcelas (maior ou igual a 4** [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 2]**), e é apresentado o valorDaParcela e o valorTotalParcelado.**

Apresentação de **Código da Questão 1**:

# *Exigência de código (1 de 6) - Print com nome completo*

*print*('Bem-vindos a loja de Robson Aparecido Avelar dos Santos! \n')

# *Exigência de código (2 de 6) - Implementar input do valor do pedido e número de parcelas*

valorDoPedido = float(*input*('Digite o valor do pedido: '))

quantidadedeParcelas = int(*input*('Digite o número de parcelas: '))

# *Exigência de código (3 de 6) e (5 de 6) - Implementar juros e estruturas condicionais*

*if* quantidadedeParcelas *<* 4:

    juros = 0.0

*elif* quantidadedeParcelas *>=*4 and quantidadedeParcelas *<*6:

    juros = 0.04

*elif* quantidadedeParcelas *>=*6 and quantidadedeParcelas *<*9:

    juros = 0.08

*elif* quantidadedeParcelas *>=*9 and quantidadedeParcelas *<*13:

    juros = 0.16

*else*:

    juros = 0.32

# *Exigência de código (4 de 6) - Implementar valor da parcela e valor total parcelado*

valordeParcela = (valorDoPedido \* (1 + juros)) / quantidadedeParcelas

valorTotalParcelado = valordeParcela \* quantidadedeParcelas

# *Exigência de código (6 de 6) - Fazer comentários relevantes no código*

# *Exigência de console (1 de 2) - Apresentar mensagem com o nome completo*

*print*('Atendimento realizado por Robson Aparecido Avelar dos Santos \n')

# *Exigência de console (2 de 2) - Apresentar parcelamento com juros se quantidade de parcelas for maior ou igual a 4.*

*if* quantidadedeParcelas *>=* 4:

*print*(f'Parcelamento em {quantidadedeParcelas} vezes com juros de {juros \* 100}%')

*print*(f'Valor da parcela: R$ {valordeParcela:.2f}')

*print*(f'Valor total parcelado: R$ {valorTotalParcelado:.2f}')

*else*:

*print*(f'Parcelamento em {quantidadedeParcelas} vezes sem juros')

*print*('Valor da parcela: R$ {valordeParcela:.2f}')

*print*(f'Valor total parcelado: R$ {valorTotalParcelado:.2f}')

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

Captura de pantalla de computadora

Descripción generada automáticamente

Captura de pantalla de computadora

Descripción generada automáticamente

Captura de pantalla de computadora

Descripción generada automáticamenteCaptura de pantalla de computadora

Descripción generada automáticamente

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 loja que vende Marmitas de Bife Acebolado ou Filé de Frango. Você ficou com a parte de desenvolver a interface do cliente para retirada do produto.

A Loja possui seguinte relação:

* Tamanho **P** de Bife Acebolado (**BA**) custa 16 reais e o Filé de Frango (**FF**) custa 15 reais;
* Tamanho **M** de Bife Acebolado (**BA**) custa 18 reais e o Filé de Frango (**FF**) custa 17 reais;
* Tamanho **G** de Bife Acebolado (**BA**) custa 22 reais e o Filé de Frango (**FF**) custa 21 reais;

Elabore um programa em Python que:

1. Deve-se implementar o **print** com o seu **nome completo** (somente print, não usar input aqui).

Por exemplo: **print(“Bem vindos a loja de Marmitas 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];

1. Deve-se implementar o input do **sabor** (BA/FF) e o print “Sabor inválido. Tente novamente" se o usuário entra com valor diferente de BA e FF [EXIGÊNCIA DE CÓDIGO 2 de 8];
2. 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];
3. 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];
4. Deve-se implementar um **acumulador** para somar os valores dos pedidos [EXIGÊNCIA DE CÓDIGO 5 de 8];
5. 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];
6. Deve-se implementar as estruturas de **while, break, continue (todas elas)** [EXIGÊNCIA DE CÓDIGO 7 de 8];
7. Deve-se inserir comentários **relevantes** no código [EXIGÊNCIA DE CÓDIGO 8 de 8];
8. 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];
9. Deve-se apresentar na saída de console um pedido em que o usuário errou o **sabor** [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 4];
10. Deve-se apresentar na saída de console um pedido em que o usuário errou o **tamanho** [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 4];
11. 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:

A imagem apresenta a seguinte saída de código:

------ Bem-vindo a Loja de Marmitas do Bruno Kostiuk -----------
------------------------------Cardápio--------------------------
----------------------------------------------------------------
---| Tamanho  |  Bife Acebolado(BA)  |  Filé de Frango(FF)  |---
---|    P     |       R$ 16.00       |       R$ 15.00       |---
---|    M     |       R$ 18.00       |       R$ 17.00       |---
---|    G     |       R$ 22.00       |       R$ 21.00       |---
----------------------------------------------------------------
Entre com o sabor desejado (BA/FF): BF
Sabor inválido. Tente novamente

Entre com o sabor desejado (BA/FF): BA
Entre com o tamanho desejado (P/M/G): EXGG
Tamanho inválido. Tente novamente

Entre com o sabor desejado (BA/FF): BA
Entre com o tamanho desejado (P/M/G): M
Você pediu um Bife Acebolado no tamanho M: R$ 18.00

Deseja mais alguma coisa? (S/N): S
Entre com o sabor desejado (BA/FF): FF
Entre com o tamanho desejado (P/M/G): G
Você pediu um Filé de Frango no tamanho G: R$ 21.00

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

O valor total a ser pago: R$ 39.00

**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 ao restaurante de Robson Aparecido Avelar dos Santos')

*print*('--'\*13, 'Cardápio', '--'\*13)

*print*('---| Tamanho | Bife Acebolado (BA) | Filé de Frango (FF) | ---')

*print*('---|    P    |     R$ 16,00        |     R$ 15,00        | ---')

*print*('---|    M    |     R$ 18,00        |     R$ 17,00        | ---')

*print*('---|    G    |     R$ 22,00        |     R$ 21,00        | ---')

*print*('--'\*31)

total = 0

continuar = True

*while* continuar:

    # *Escolhendo o sabor do prato*

    sabor = *input*('Digite o sabor do prato (BA ou FF): ')

*while* sabor not in ['BA', 'FF']:

*print*('Sabor inválido. Tente novamente.')

        # *Caso de sabor inválido, refazendo a pergunta*

        sabor = *input*('Digite o sabor do prato (BA ou FF): ')

    # *Escolhendo o tamanho do prato*

    tamanho = *input*('Escolha o tamanho do prato desejado (P, M ou G): ')

*while* tamanho not in ['P', 'M', 'G']:

*print*('Tamanho inválido. Tente novamente.')

        # *Caso de tamanho inválido, refazendo a pergunta*

        tamanho = *input*('Escolha o tamanho do prato desejado (P, M ou G): ')

    # *Calculando o preço do prato dependendo do tamanho e sabor*

*if* sabor *==* 'BA':

*if* tamanho *==* 'P':

            preco = 16.00

*elif* tamanho *==* 'M':

            preco = 18.00

*elif* tamanho *==* 'G':

            preco = 22.00

*elif* sabor *==* 'FF':

*if* tamanho *==* 'P':

            preco = 15.00

*elif* tamanho *==* 'M':

            preco = 17.00

*elif* tamanho *==* 'G':

            preco = 21.00

    total += preco

*print*(f'Você pediu um {sabor} no tamanho {tamanho}. O preço é R$ {preco:.2f}.')

    # *Perguntando ao cliente se deseja mais alguma coisa*

    mais\_alguma\_coisa = *input*('Deseja pedir mais alguma coisa? (S/N): ')

*while* mais\_alguma\_coisa not in ['S', 'N']:

*print*('Opção inválida. Deseja pedir mais alguma coisa? (S/N): ')

*if* mais\_alguma\_coisa *==* 'N':

        continuar = False

# *Exibindo o valor total da conta ao cliente*

*print*(f'O total da sua conta é R$ {total:.2f}. Obrigado pela preferência!')

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

Texto

Descripción generada automáticamente

QUESTÃO 3 de 4 - Conteúdo até aula 05

Enunciado: Você foi contratado para desenvolver um sistema de cobrança de serviços de uma fábrica que vende Camisetas em atacado. Você ficou com a parte de desenvolver a interface com o funcionário.

A Fábrica opera as vendas da seguinte maneira:

* Camiseta Manga Curta Simples (MCS), o valor unitário é de um real e oitenta centavos;
* Camiseta Manga Longa Simples (MLS), o valor unitário é de dois reais e dez centavos;
* Camiseta Manga Curta Com Estampa (MCE), o valor unitário é de dois reais e noventa centavos;
* Camiseta Manga Longa Com Estampa (MLE), o valor unitário é de três reais e vinte centavos;
* Se número de camisetas for **menor** que 20 não há desconto na venda;
* Se número de camisetas for **igual ou maior** que 20 e **menor** que 200, o desconto será de 5%;
* Se número de camisetas for **igual ou maior** que 200 e **menor** que 2000, o desconto será de 7%;
* Se número de camisetas for **igual ou maior** que 2000 e **menor ou igual** que 20000, o desconto será de 12%;
* Se número de camisetas for **maior** que 20000, não é aceito pedidos nessa quantidade de camisetas;

* Para o **adicional** de frete por transportadora (1) é cobrado um valor **extra** de 100 reais;
* Para o **adicional** de frete por Sedex (2) é cobrado um valor **extra** de 200 reais;
* Para o **adicional** de retirar o pedido na fábrica (0) é cobrado um valor **extra** de 0 reais;

O valor final da conta é calculado da seguinte maneira:

total = (**modelo \* num\_camisetas) + frete**

Elabore um programa em Python que:

1. Deve-se implementar o **print** com o seu **nome completo** (somente print, não usar input aqui).

Por exemplo: **print(“Bem vindos a Fábrica de Camisetas do Bruno Kostiuk”)** [EXIGÊNCIA DE CÓDIGO 1 de 7];

1. Deve-se implementar a função **escolha\_modelo()** em que**:** [EXIGÊNCIA DE CÓDIGO 2 de 7];
   1. Pergunta o **modelo** desejado;
   2. **Retorna** **o valor** **do modelo** com base na escolha do usuário (use **return**);
   3. Repete a pergunta do item **B.a** se digitar uma opção diferente de: MCS/MLS/MCE/MLE;
2. Deve-se implementar a função **num\_camisetas()** em que**:** [EXIGÊNCIA DE CÓDIGO 3 de 7];
   1. Pergunta o **número de camisetas**;
   2. **Retorna** (use **return**) o **número de camisetas** com desconto seguindo a regra do enunciado (desconto calculado em cima do número de camisetas);
   3. Repete a pergunta do item **C.a** se digitar um valor acima de 20000 ou valor não numérico (use try/except para não numérico)
3. Deve-se implementar a função **frete()** em que**:** [EXIGÊNCIA DE CÓDIGO 4 de 7];
   1. Pergunta pelo serviço **adicional de frete**;
   2. **Retorna** (use **return**) o **valor** de apenas uma das **opções** de **frete**
   3. Repetir a pergunta item **D.a** se digitar uma opção diferente de: 1/2/0;
4. 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];
5. Deve-se implementar **try/except** [EXIGÊNCIA DE CÓDIGO 6 de 7];
6. Deve-se inserir comentários relevantes no código [EXIGÊNCIA DE CÓDIGO 7 de 7];
7. Deve-se apresentar na saída de console uma mensagem com o seu nome completo [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 4];
8. Deve-se apresentar na saída de console um pedido no qual o usuário errou a opção de modelo [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 4];
9. Deve-se apresentar na saída de console um pedido no qual o usuário digitou ultrapassou no número de camisetas [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 4];
10. Deve-se apresentar na saída de console um pedido com opção de modelo, número de camisetas e frete válidos [EXIGÊNCIA DE SAÍDA DE CONSOLE 4 de 4];

EXEMPLO DE SAÍDA DE CONSOLE:

A imagem apresenta uma saída de console com o seguinte texto:

Bem vindo a Fábrica de Camisetas do do Bruno Kostiuk

Entre com o modelo desejado
MCS - Manga Curta Simples
MLS - Manga Longa Simples
MCE - Manga Curta Com Estampa
MLE - Manga Longa Com Estampa
>>MLL
Escolha inválida, entre com o modelo novamente


Entre com o modelo desejado
MCS - Manga Curta Simples
MLS - Manga Longa Simples
MCE - Manga Curta Com Estampa
MLE - Manga Longa Com Estampa
>>MCS
Entre com o número de camisetas: 300000
Não aceitamos tantas camisetas de uma vez.
Por favor, entre com o número de camisetas novamente.

Entre com o número de camisetas: 10000

Escolha o tipo de frete:
1 - Frete por transportadora - R$ 100.00
2 - Frete por Sedex - R$ 200.00
0 - Retirar pedido na fábrica - R$ 0.00
>>2
Total: R$ 16040.00 (Modelo: 1.80 * Quantidade(com desconto): 8800 + frete: 200.00)

**Figura 3.1: Exemplo de saída de console que o aluno deve fazer. Em que se pergunta pelo modelo e se erra opção inicialmente, que se passa do número de camisetas acima do aceito. Na sequência, o usuário digitou um modelo, número de camisetas e frete válidos.**

Apresentação de Código da Questão 3:

# *Mensagem de boa-vinda*

*print*("Bem vindos a Fábrica de Camisetas do Robson Aparecido Avelar dos Santos")

# *Função para escolher o modelo de camiseta*

def *escolha\_modelo*():

*while* True:

*print*("Escolha o modelo desejado")

*print*("MCS - Manga Curta Simples")

*print*("MLS - Manga Longa Simples")

*print*("MCE - Manga Curta Com Estampa")

*print*("MLE - Manga Longa Com Estampa")

        modelo = *input*(">>").*upper*()

*if* modelo *==* 'MCS':

*return* 1.80

*elif* modelo *==* 'MLS':

*return* 2.10

*elif* modelo *==* 'MCE':

*return* 2.90

*elif* modelo *==* 'MLE':

*return* 3.20

*else*:

*print*('Escolha inválida, entre com o modelo novamente') # *Mensagem informando que a escolha é inválida*

# *Função para obter o número de camisetas e calcular o desconto*

def *num\_camisetas*():

*while* True:

*try*:

            numero = int(*input*('Escolha o número de camisetas: '))

*if* numero *>* 20000:

*print*('Não aceitamos tantas camisetas de uma vez. Por favor, entre com o número de camisetas novamente.') # *Mensagem informando que a quantidade de camisetas é inválida*

*else*:

*if* numero *<* 20:

                    desconto = 0

*elif* 20 *<=* numero *<* 200:

                    desconto = 0.05

*elif* 200 *<=* numero *<* 2000:

                    desconto = 0.07

*elif* 2000 *<=* numero *<=* 20000:

                    desconto = 0.12

*return* numero \* (1 - desconto)

*except* ValueError:

*print*('Valor não numérico. Tente novamente.')

# *Função para escolher o tipo de frete*

def *escolha\_frete*():

*while* True:

*print*("Escolha o tipo de frete:")

*print*("1 - Frete por transportadora - R$ 100.00")

*print*("2 - Frete por Sedex - R$ 200.00")

*print*("0 - Retirar pedido na fábrica - R$ 0.00")

        frete = *input*(">>")

*if* frete *==* '1':

*return* 100

*elif* frete *==* '2':

*return* 200

*elif* frete *==* '0':

*return* 0

*else*:

*print*('Opção inválida, tente novamente')

# *Cálculo do total a pagar*

modelo = *escolha\_modelo*()

quantidade = *num\_camisetas*()

frete = *escolha\_frete*()

total = (modelo \* quantidade) + frete

# *Resultado do total da compra*

*print*(f"Total: R$ {total:.2f} (Modelo: {modelo:.2f} \* Quantidade(com desconto): {quantidade} + frete: {frete:.2f})")

# *Comentários relevantes*

# *O código acima implementa um sistema de cobrança para uma fábrica de camisetas*

# *Os preços das camisetas variam conforme o modelo escolhido*

# *A quantidade de camisetas compradas pode proporcionar descontos*

# *O tipo de frete pode adicionar um custo extra ao total*

# *O programa valida as entradas do usuário e calcula o total a pagar com base nas escolhas feitas*

Apresentação de Saída do Console da Questão 3:

Texto

Descripción generada automáticamente

QUESTÃO 4 de 4 - Conteúdo até aula 06

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

1. Cadastrar Funcionário
2. Consultar Funcionário
   1. Consultar Todos
   2. Consultar por Id
   3. Consultar por setor
   4. Retornar ao menu
3. Remover Funcionário
4. Encerrar Programa

Elabore um programa em Python que:

1. Deve-se implementar o **print** com o seu **nome completo** (somente print, não usar input aqui).

Por exemplo: **print(“Bem vindos a empresa do Bruno Kostiuk”)** [EXIGÊNCIA DE CÓDIGO 1 de 8];

1. Deve-se implementar uma lista com o nome de **lista\_funcionarios** e a variável **id\_global** com valor inicial igual ao número de seu RU [EXIGÊNCIA DE CÓDIGO 2 de 8];
2. Deve-se implementar uma função chamada **cadastrar\_funcionario(id)** em que**:** [EXIGÊNCIA DE CÓDIGO 3 de 8];
   1. Pergunta **nome**, **setor**, **salario** do funcionário;
   2. Armazena o **id** (este é fornecido via parâmetro da função), **nome**, **setor**, **salario** dentro de um dicionário;
   3. **Copiar** o dicionário para dentro da **lista\_funcionarios** (utilizar o **copy**);
3. Deve-se implementar uma função chamada **consultar\_funcionarios()** em que**:** [EXIGÊNCIA DE CÓDIGO 4 de 8];
   1. Deve-se perguntar qual opção deseja (1. Consultar Todos / 2. Consultar por Id / 3. Consultar por Setor / 4. Retornar ao menu):
      1. Se Consultar Todos, apresentar todos os funcionários com todos os seus dados cadastrados;
      2. Se Consultar por Id, solicitar ao usuário que informe um id, e apresentar o funcionário específico com todos os seus dados cadastrados;
      3. Se Consultar por Setor, solicitar ao usuário que informe o setor, e apresentar o(s) funcionário(s) do setor com todos os seus dados cadastrados;
      4. Se Retornar ao menu, deve-se **retornar** ao menu principal (return);
      5. Se Entrar com um valor diferente de 1, 2, 3 ou 4, printar “Opção inválida" e repetir a pergunta **D.a**.
      6. Enquanto o usuário não escolher a opção 4, o menu consultar funcionários deve se repetir.
4. Deve-se implementar uma função chamada **remover\_funcionario()** em que**:** [EXIGÊNCIA DE CÓDIGO 5 de 8];
   1. Deve-se pergunta pelo **id** do funcionário a ser removido;
   2. Remover o funcionário da **lista\_funcionarios**;
   3. Se o id fornecido não for de um funcionário da lista, printar “Id inválido” e repetir a pergunta **E.a**.
5. 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];
   1. Deve-se pergunta qual opção deseja (1. Cadastrar Funcionário / 2. Consultar Funcionário / 3. Remover Funcionário / 4. Encerrar Programa):
      1. Se Cadastrar Funcionário, **incrementar** em um **id\_ global** e chamar a função **cadastrar\_funcionario(id\_ global)**;
      2. Se Consultar Funcionário, chamar função **consultar\_funcionario ()**;
      3. Se Remover Funcionário, chamar função **remover\_funcionario()**;
      4. Se Encerrar Programa, sair do menu (e com isso acabar a execução do código);
      5. Se Entrar com um valor diferente de 1, 2, 3 ou 4, printar “Opção inválida" e repetir a pergunta **F.a**.
      6. Enquanto o usuário não escolher a opção 4, o menu deve se repetir.
6. Deve-se implementar uma **lista de dicionários** (uma lista contento dicionários dentro)[EXIGÊNCIA DE CÓDIGO 7 de 8];
7. Deve-se inserir comentários **relevantes** no código [EXIGÊNCIA DE CÓDIGO 8 de 8];
8. Deve-se apresentar na saída de console uma mensagem com o seu nome completo [EXIGÊNCIA DE SAÍDA DE CONSOLE 1 de 6];
9. Deve-se apresentar na saída de console um cadastro de 3 funcionários (sendo **2** deles no mesmo setor) [EXIGÊNCIA DE SAÍDA DE CONSOLE 2 de 6];
10. Deve-se apresentar na saída de console uma consulta de todos os funcionários [EXIGÊNCIA DE SAÍDA DE CONSOLE 3 de 6];
11. Deve-se apresentar na saída de console uma consulta por código (id) de um dos funcionários [EXIGÊNCIA DE SAÍDA DE CONSOLE 4 de 6];
12. Deve-se apresentar na saída de console uma consulta por setor em que **2** funcionários sejam do mesmo setor [EXIGÊNCIA DE SAÍDA DE CONSOLE 5 de 6];
13. Deve-se apresentar na saída de console uma remoção de um dos funcionários seguida de uma consulta de todos os funcionários [EXIGÊNCIA DE SAÍDA DE CONSOLE 6 de 6];

EXEMPLO DE SAÍDA DE CONSOLE:

A imagem apresenta o seguinte texto de saída de console: 

Bem vindo a Empresa do Bruno Kostiuk
--------------------------------------------------
--------------- MENU PRINCIPAL -------------------
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>>1
--------------------------------------------------
---------- MENU CADASTRAR FUNCIONÁRIO ------------------
Id do Funcionário: 4297914
Por favor entre com o nome do Funcionário: Bruno
Por favor entre com o setor do Funcionário: Configurações
Por favor entre com o salário do Funcionário: 1000.00
--------------------------------------------------

**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 funcionário, note que o ID do funcionário inicia em 4297914 pois o id\_global é inicializado em 4297913.**

A imagem apresenta o seguinte texto de saída de console:

--------------- MENU PRINCIPAL -------------------
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>>1
--------------------------------------------------
---------- MENU CADASTRAR FUNCIONÁRIO ------------------
Id do Funcionário: 4297915
Por favor entre com o nome do Funcionário: Tamy
Por favor entre com o setor do Funcionário: Tutorias
Por favor entre com o salário do Funcionário: 1000
--------------------------------------------------

--------------------------------------------------
--------------- MENU PRINCIPAL -------------------
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>>1
--------------------------------------------------
---------- MENU CADASTRAR FUNCIONÁRIO ------------------
Id do Funcionário: 4297916
Por favor entre com o nome do Funcionário: Osmar
Por favor entre com o setor do Funcionário: Tutorias
Por favor entre com o salário do Funcionário: 1000.00
--------------------------------------------------

**Figura 4.2: Exemplo de saída de console que o aluno deve fazer. São cadastrados mais dois funcionários com mesmo setor.**

**A imagem apresenta o seguinte texto de saída de console:

--------------------------------------------------
--------------- MENU PRINCIPAL -------------------
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>>2
--------------------------------------------------
---------- MENU CONSULTAR FUNCIONÁRIO ------------------
Escolha a opção desejada:
1 - Consultar Todos os Funcionários
2 - Consultar Funcionário por id
3 - Consultar Funcionário(s) por setor
4 - Retornar
>>1
----------------
id: 4297914
nome: Bruno
setor: Configurações
salário: 1000.0

id: 4297915
nome: Tamy
setor: Tutorias
salário: 1000.0

id: 4297916
nome: Osmar
setor: Tutorias
salário: 1000.0

----------------**

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

A imagem apresenta o seguinte texto de saída de console: 

--------------------------------------------------
---------- MENU CONSULTAR FUNCIONÁRIO ------------------
Escolha a opção desejada:
1 - Consultar Todos os Funcionários
2 - Consultar Funcionário por id
3 - Consultar Funcionário(s) por setor
4 - Retornar
>>2
Digite o id do funcionário: 4297914
----------------
id: 4297914
nome: Bruno
setor: Configurações
salario: 1000.0

----------------
--------------------------------------------------

--------------------------------------------------
---------- MENU CONSULTAR FUNCIONÁRIO ------------------
Escolha a opção desejada:
1 - Consultar Todos os Funcionários
2 - Consultar Funcionário por id
3 - Consultar Funcionário(s) por setor
4 - Retornar
>>3
Digite o setor do(s) funcionário(s): Tutorias
----------------
id: 4297915
nome: Tamy
setor: Tutorias
salario: 1000.0

id: 4297916
nome: Osmar
setor: Tutorias
salario: 1000.0

----------------

Figura 4.4: Exemplo de saída de console que o aluno deve fazer. Em que se consulta o funcionário com id número 4297914 e consulta pelo nome do setor (Tutorias).

A imagem apresenta o seguinte texto de saída do console:

--------------- MENU PRINCIPAL -------------------
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>>3
--------------------------------------------------
------------ MENU REMOVER FUNCIONÁRIO ------------------
Digite o id do funcionario a ser removido: 4297914
Funcionário removido com sucesso!
--------------------------------------------------
--------------- MENU PRINCIPAL -------------------
Escolha a opção desejada:
1 - Cadastrar Funcionários
2 - Consultar Funcionário(s)
3 - Remover Funcionário
4 - Sair
>>2
--------------------------------------------------
---------- MENU CONSULTAR FUNCIONÁRIO ------------------
Escolha a opção desejada:
1 - Consultar Todos os Funcionários
2 - Consultar Funcionário por id
3 - Consultar Funcionário(s) por setor
4 - Retornar
>>1
----------------
id: 4297915
nome: Tamy
setor: Tutorias
salário: 1000.0

id: 4297916
nome: Osmar
setor: Tutorias
salário: 1000.0

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

Apresentação de **Código da Questão 4**:

*print*("Bem vindos a empresa do Robson Aparecido Avelar dos Santos")  # *Empresa com o meu nome*

# *Lista para armazenar os funcionários e o ID global inicial*

lista\_funcionarios = []

id\_global = 1298790

def *cadastrar\_funcionario*(id):  # *Função de cadastro de funcionário*

*print*(f'ID do novo funcionário: {id}')

    nome = *input*("Digite o nome do funcionário: ")

    setor = *input*("Digite o setor do funcionário: ")

    salario = float(*input*("Digite o salário do funcionário: "))

    funcionario = {

        'id': id,

        'nome': nome,

        'setor': setor,

        'salario': salario

    }

    lista\_funcionarios.*append*(funcionario.*copy*())

def *consultar\_funcionarios*():  # *Função de consulta de funcionários*

*while* True:

*print*("Consultar Funcionário")

*print*("1. Consultar Todos")

*print*("2. Consultar por Id")

*print*("3. Consultar por Setor")

*print*("4. Retornar ao menu")

        opcao = int(*input*("Escolha uma opção: "))

*if* opcao *==* 1:

*for* func *in* lista\_funcionarios:

*print*(func)

*elif* opcao *==* 2:

            id\_consulta = int(*input*("Digite o ID do funcionário: "))

            encontrado = False

*for* func *in* lista\_funcionarios:

*if* func['id'] *==* id\_consulta:

*print*(func)

                    encontrado = True

*break*

*if* not encontrado:

*print*("Id inválido")

*elif* opcao *==* 3:

            setor\_consulta = *input*("Digite o setor dos funcionários: ")

            encontrados = [func *for* func *in* lista\_funcionarios *if* func['setor'] *==* setor\_consulta]

*if* encontrados:

*for* func *in* encontrados:

*print*(func)

*else*:

*print*("Nenhum funcionário encontrado nesse setor")

*elif* opcao *==* 4:

*return*

*else*:

*print*("Opção inválida")

def *remover\_funcionario*():  # *Função de remoção de funcionário da lista*

*while* True:

        id\_remover = int(*input*("Digite o ID do funcionário a ser removido: "))

        encontrado = False

*for* func *in* lista\_funcionarios:

*if* func['id'] *==* id\_remover:

                lista\_funcionarios.*remove*(func)

*print*("Funcionário removido com sucesso.")

                encontrado = True

*break*

*if* not encontrado:

*print*("Id inválido")

*else*:

*break*

*while* True:  # *Estrutura de menu no código principal*

*print*("Menu:")

*print*("1. Cadastrar Funcionário")

*print*("2. Consultar Funcionário")

*print*("3. Remover Funcionário")

*print*("4. Encerrar Programa")

    opcao = int(*input*("Escolha uma opção: "))

*if* opcao *==* 1:

        id\_global += 1

*cadastrar\_funcionario*(id\_global)

*elif* opcao *==* 2:

*consultar\_funcionarios*()

*elif* opcao *==* 3:

*remover\_funcionario*()

*elif* opcao *==* 4:

*print*("Encerrando o programa...")

*break*

*else*:

*print*("Opção inválida")

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

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente