

**ANO**  
**2023**



# **UNINTER**

## **ATIVIDADE PRÁTICA:** **LÓGICA DE PROGRAMAÇÃO** **E ALGORITMOS**

**Aluno:**

**EMERSON VAZ RU:4191452**



## QUESTÃO 1 de 4 (25 pontos)

### 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 vende em atacado. Uma das estratégias de vendas dessa empresa X é dar desconto maiores por unidade conforme a tabela abaixo:

Quantidades	Desconto
Até 9	0% na unidade
Entre 10 e 99	5% na unidade
Entre 100 e 999	10% na unidade
De 1000 para mais	15% na unidade

Elabore um programa em Python que:

1. Entre com o valor unitário do produto (Lembrar que número decimal é feito com ponto e não vírgula);
2. Entre com a quantidade desse produto;
3. O programa deve retornar o valor total sem desconto;
4. O programa deve retornar o valor total após o desconto;
5. Deve-se utilizar estruturas if, elif e else (EXIGÊNCIA 1 de 1);
6. Colocar um exemplo de SAIDA DE CONSOLE de compra de mais de 10 und. (para mostrar que o desconto foi aplicado)

Segue o exemplo de SAIDA DE CONSOLE:

```
Bem Vindo a Loja do Renan Portela Jorge | Identificador Pessoal
Entre com valor do produto: 10
Entre com valor do quantidade: 150
O valor sem desconto foi: R$ 1500.00
O valor com desconto foi: R$ 1350.00 (desconto 10%)
```

**OBS:** Para os números decimais fiquem com somente duas casas depois da vírgula utilize {:.2f}. Exemplo:

```
print('O valor sem desconto foi: R$ {:.2f}'.format(sub_total))
```

### Apresentação do Código

```
nome = 'Emerson'
ru = 4191452
print(f'Bem vindo a loja do {nome}\nRU:{ru}')
val = float(input('Valor do Produto: '))
Qtd = int(input('Quantidade de Produtos: '))
subtotal = val * Qtd
total = 0
if Qtd <= 9:
    total += subtotal
elif Qtd >= 10 and Qtd <= 99:
    print('Você ganhou 5% de desconto!\n')
    total += subtotal - subtotal * 0.05
```



```
elif Qtd >= 100 and Qtd <= 999:
    print('Você ganhou 10% de desconto!\n')
    total += subtotal - subtotal * 0.10
else:
    print('Você ganhou 15% de desconto!\n')
    total += subtotal - subtotal * 0.15
desconto = subtotal - total
print(f'Total sem Desconto:R$ {subtotal:.2f}')
print(f'Desconto Concedido:R$ {desconto:.2f}')
print(f'Total com Desconto:R$ {total:.2f}')
```

### Saída do Console

```
Bem vindo a loja do Emerson
RU:4191452
Valor do Produto: 10
Quantidade de Produtos: 150
Você ganhou 10% de desconto!

Total sem Desconto:R$ 1500.00
Desconto Concedido:R$ 150.00
Total com Desconto:R$ 1350.00

Process finished with exit code 0
```

## QUESTÃO 2 de 4 (25 pontos)

### CONTEÚDO ATÉ AULA 04

**Enunciado:** Você e sua equipe de programadores foram contratados para desenvolver um app de vendas para uma lanchonete. Você ficou com a parte de desenvolver a interface do cliente para retirada do produto.

A lanchonete possui seguinte tabela de produtos listados com sua descrição, códigos e valores:

Código	Descrição	Valor(R\$)
100	Cachorro-Quente	9,00
101	Cachorro-Quente Duplo	11,00
102	X-Egg	12,00
103	X-Salada	13,00
104	X-Bacon	14,00
105	X-Tudo	17,00
200	Refrigerante Lata	5,00
201	Chá Gelado	4,00

Elabore um programa em Python que:



1. Entre com o código do produto desejado;
2. Pergunte se o cliente quer pedir mais alguma coisa (se sim repetir o passo item 2. Caso contrário ir para próximo passo);
3. Encerre a conta do cliente com o valor total;
4. Deve-se utilizar estruturas if, elif e else (EXIGÊNCIA 1 de 3);
5. Se a pessoa digitar um NÚMERO diferente dos da tabela printar na tela: 'opção inválida' e voltar para o menu (EXIGÊNCIA 2 de 3);
6. Deve-se utilizar while, break, continue (EXIGÊNCIA 3 de 3);
  - (DICA: utilizar o continue dentro else que verifica a opção inválida)
  - (DICA: utilizar o break dentro elif que verifica a opção sair)
7. Colocar um exemplo de SAIDA DE CONSOLE com dois pedidos
8. Colocar um exemplo de SAIDA DE CONSOLE com erro ao digitar no pedido

Segue o exemplo de SAIDA DE CONSOLE:

```
Bem Vindo a Lanchonete do Renan Portela Jorge |identificador
*****Cardápio*****
| Código | Descrição | Valor |
| 100 | Cachorro Quente | 9,00 |
| 101 | Cachorro Quente Duplo | 11,00 |
| 102 | X-Egg | 12,00 |
| 103 | X-Salada | 12,00 |
| 104 | X-Bacon | 14,00 |
| 105 | X-Tudo | 17,00 |
| 200 | Refrigerante Lata | 5,00 |
| 201 | Chá Gelado | 4,00 |

Entre com o código desejado: 100 Escolha de um produto
Você pediu um Cachorro-Quente no valor de 9,00
Deseja pedir mais alguma coisa?
1 - Sim
0 - Não
>>1 Pedir por mais coisa

Entre com o código desejado: 300 Errar o pedido
Opção Inválida
Entre com o código desejado: 201 Escolha de um produto
Você pediu um Chá Gelado no valor de 4,00
Deseja pedir mais alguma coisa?
1 - Sim
0 - Não
>>0 Não vai ser pedido mais nada
0 total a ser pago é: 13.00 Valor Final a ser pago
```

Figura 1: Exemplo de programa com 2 itens pedido (Sendo que um deles teve uma tentativa com erro) e no final o valor final é apresentado.

### Apresentação do Código

```
nome = 'Emerson Vaz'
ru = 4191452
```



```
print(f'Bem vindo a lanchonete do {nome}\nRU:{ru}\n')
print('*' * 17, 'Menu', '*' * 16)
print('|Código|          Descrição          |Valor|')
print('| 100 | Cachorro-Quente |9.00 |')
print('| 101 | Cachorro-Quente Duplo |11.00|')
print('| 102 | X-Egg |12.00|')
print('| 103 | X-Salada |13.00|')
print('| 104 | X-Bacon |14.00|')
print('| 105 | X-Tudo |17.00|')
print('| 200 | Refrigerante Lata |5.00 |')
print('| 201 | Chá Gelado |4.00 |')
print('-'*39)
total = 0

while True: #INÍCIO DO BLOCO DE CÓDIGO DE VERIFICAÇÃO DOS
CÓDIGOS DO MENU

    codigo = int(input('Digite o codigo do pedido!'))
    if codigo == 100:
        total += 9.00
        print('Você pediu um Cachorro-Quente no valor de:R$9.00')
        print(f'Total a ser pago é: R${total:.2f}')
        pedir_m1 = int(input('Deseja fazer mais um pedido?\n1 -
sim\n0 - Não\n'))

        if pedir_m1 == 1: #VERIFICA SE HAVERÁ UM NOVO PEDIDO
            continue
        elif pedir_m1 == 0:#ENCERRA A VERIFICAÇÃO DE UM NOVO
PEDIDO
            print(f'Obrigado e volte Sempre\nTotal a ser pago é:
R${total:.2f}')
            break

        while pedir_m1 != 1:#CONDEÇÃO DE PARADA OU REPETIÇÃO PARA
A PERGUNDA SOBRE UM NOVO PEDIDO
            print('Valor inválido, por favor digite um valor
válido!')
            pedir_m1 = int(input('Deseja fazer mais um pedido?\n1
- sim\n0 - Não\n'))

            continue

    elif codigo == 101:
        total += 11.00
        print('Você pediu um Cachorro-Quente Duplo no valor
de:R$11.00')
        print(f'Total a ser pago é: R${total:.2f}')
        pedir_m1 = int(input('Deseja pedir mais alguma coisa?\n1
- Sim\n0 - nao\n'))

        if pedir_m1 == 1:#VERIFICA SE HAVERÁ UM NOVO PEDIDO
```



```

        continue
    elif pedir_m1 == 0:#ENCERRA A VERIFICAÇÃO DE UM NOVO
PEDIDO
        print(f'Obrigado e volte Sempre\nTotal a ser pago é:
R${total:.2f}')
        break

    while pedir_m1 != 1:#CONDEÇÃO DE PARADA OU REPETIÇÃO PARA
A PERGUNDA SOBRE UM NOVO PEDIDO
        print('Opção inválida, por favor digite uma opção
válida')
        pedir_m1 = int(input('Deseja fazer mais um pedido?\n1
- sim\n0 - Não\n'))

        continue

    elif codigo == 102:
        total += 12.00
        print('Você pediu um X-Egg no valor de:R$12.00')
        print(f'Total a ser pago é: R${total:.2f}')
        pedir_m1 = int(input('Deseja pedir mais alguma coisa?\n1
- Sim\n0 - nao\n'))

        if pedir_m1 == 1:#VERIFICA SE HAVERÁ UM NOVO PEDIDO
            continue
        elif pedir_m1 == 0:#ENCERRA A VERIFICAÇÃO DE UM NOVO
PEDIDO
            print(f'Obrigado e volte Sempre\nTotal a ser pago é:
R${total:.2f}')
            break

        while pedir_m1 != 1:#CONDEÇÃO DE PARADA OU REPETIÇÃO PARA
A PERGUNDA SOBRE UM NOVO PEDIDO
            print('Opção inválida, por favor digite uma opção
válida')
            pedir_m1 = int(input('Deseja fazer mais um pedido?\n1
- sim\n0 - Não\n'))

            continue

    elif codigo == 103:
        total += 13.00
        print('Você pediu um X-Salada no valor de:R$13.00')
        print(f'Total a ser pago é: R${total:.2f}')
        pedir_m1 = int(input('Deseja pedir mais alguma coisa?\n1
- Sim\n0 - nao\n'))

        if pedir_m1 == 1:#VERIFICA SE HAVERÁ UM NOVO PEDIDO
            continue
        elif pedir_m1 == 0:#ENCERRA A VERIFICAÇÃO DE UM NOVO
PEDIDO

```



```

        print(f'Obrigado e volte Sempre\nTotal a ser pago é:
R${total:.2f}')
        break

    while pedir_m1 != 1:#CONDEÇÃO DE PARADA OU REPETIÇÃO PARA
A PERGUNDA SOBRE UM NOVO PEDIDO
        print('Opção inválida, por favor digite uma opção
válida')
        pedir_m1 = int(input('Deseja fazer mais um pedido?\n1
- sim\n0 - Não\n'))

        continue

    elif codigo == 104:
        total += 14.00
        print('Você pediu um X-Bacon no valor de:R$14.00')
        print(f'Total a ser pago é: R${total:.2f}')
        pedir_m1 = int(input('Deseja pedir mais alguma coisa?\n1
- Sim\n0 - nao\n'))

        if pedir_m1 == 1:#VERIFICA SE HAVERÁ UM NOVO PEDIDO
            continue
        elif pedir_m1 == 0:#ENCERRA A VERIFICAÇÃO DE UM NOVO
PEDIDO
            print(f'Obrigado e volte Sempre\nTotal a ser pago é:
R${total:.2f}')
            break

    while pedir_m1 != 1:#CONDEÇÃO DE PARADA OU REPETIÇÃO PARA
A PERGUNDA SOBRE UM NOVO PEDIDO
        print('Opção inválida, por favor digite uma opção
válida')
        pedir_m1 = int(input('Deseja fazer mais um pedido?\n1
- sim\n0 - Não\n'))

        continue

    elif codigo == 105:
        total += 17.00
        print('Você pediu um X-Tudo no valor de:R$17.00')
        print(f'Total a ser pago é: R${total:.2f}')
        pedir_m1 = int(input('Deseja pedir mais alguma coisa?\n1
- Sim\n0 - nao\n'))

        if pedir_m1 == 1:#VERIFICA SE HAVERÁ UM NOVO PEDIDO
            continue
        elif pedir_m1 == 0:#ENCERRA A VERIFICAÇÃO DE UM NOVO
PEDIDO
            print(f'Obrigado e volte Sempre\nTotal a ser pago é:
R${total:.2f}')
            break

```



```

        while pedir_m1 != 1: #CONDEÇÃO DE PARADA OU REPETIÇÃO PARA
A PERGUNDA SOBRE UM NOVO PEDIDO
            print('Opção inválida, por favor digite uma opção
válida')
            pedir_m1 = int(input('Deseja fazer mais um pedido?\n1
- sim\n0 - Não\n'))

            continue

        elif codigo == 200:
            total += 5.00
            print('Você pediu um Refrigerante Lata no valor
de:R$5.00')
            print(f'Total a ser pago é: R${total:.2f}')
            pedir_m1 = int(input('Deseja pedir mais alguma coisa?\n1
- Sim\n0 - nao\n'))

            if pedir_m1 == 1: #VERIFICA SE HAVERÁ UM NOVO PEDIDO
                continue
            elif pedir_m1 == 0: #ENCERRA A VERIFICAÇÃO DE UM NOVO
PEDIDO
                print(f'Obrigado e volte Sempre\nTotal a ser pago é:
R${total:.2f}')
                break

            while pedir_m1 != 1: #CONDEÇÃO DE PARADA OU REPETIÇÃO PARA
A PERGUNDA SOBRE UM NOVO PEDIDO
                print('Opção inválida, por favor digite uma opção
válida')
                pedir_m1 = int(input('Deseja fazer mais um pedido?\n1
- sim\n0 - Não\n'))

                continue

            elif codigo == 201:
                total += 4.00
                print('Você pediu um Chá Gelado no valor de:R$4.00')
                print(f'Total a ser pago é: R${total:.2f}')
                pedir_m1 = int(input('Deseja pedir mais alguma coisa?\n1
- Sim\n0 - nao\n'))

                if pedir_m1 == 1: #VERIFICA SE HAVERÁ UM NOVO PEDIDO
                    continue
                elif pedir_m1 == 0: #ENCERRA A VERIFICAÇÃO DE UM NOVO
PEDIDO
                    print(f'Obrigado e volte Sempre\nTotal a ser pago é:
R${total:.2f}')
                    break

            while pedir_m1 != 1: #CONDEÇÃO DE PARADA OU REPETIÇÃO

```





```

PARA A PERGUNDA SOBRE UM NOVO PEDIDO
    print('Opção inválida, por favor digite uma opção
válida')
    pedir_m1 = int(input('Deseja fazer mais um pedido?\n1
- sim\n0 - Não\n'))

    continue

else:
    print('Opção inválida, por favor digite uma opção
válida')
    continue
#FIM DO BLOCO DE CÓDIGO DE VERIFICAÇÃO DOS CÓDIGOS DO MENU
    
```

## Saída do Console

```

Bem vindo a lanchonete do Emerson Vaz
RU:4191452

***** Menu *****
|Código|      Descrição      |Valor|
| 100 | Cachorro-Quente  |9.00 |
| 101 | Cachorro-Quente Duplo |11.00|
| 102 | X-Egg           |12.00|
| 103 | X-Salada        |13.00|
| 104 | X-Bacon         |14.00|
| 105 | X-Tudo          |17.00|
| 200 | Refrigerante Lata |5.00 |
| 201 | Chá Gelado      |4.00 |
-----
Digite o codigo do pedido!100
Você pediu um Cachorro-Quente no valor de:R$9.00
Total a ser pago é: R$9.00
Deseja fazer mais um pedido?
1 - sim
0 - Não
1
Digite o codigo do pedido!300
Opção inválida, por favor digite uma opção válida
Digite o codigo do pedido!201
Você pediu um Chá Gelado no valor de:R$4.00
Total a ser pago é: R$13.00
Deseja pedir mais alguma coisa?
1 - Sim
0 - nao
0
Obrigado e volte Sempre
Total a ser pago é: R$13.00
    
```



## QUESTÃO 3 de 4 (25 pontos)

### CONTEÚDO ATÉ AULA 05

**Enunciado:** Imagina-se que você e sua equipe foram contratados por uma empresa de logística que acabou de entrar no ramo. Essa empresa trabalha com encomendas de pequeno e médio porte e opera somente entre 3 cidades.

O valor que a empresa cobra por objeto é dado pela seguinte equação:

$$total = dimensões * peso * rota$$

Em que cada uma das variáveis que compõe o preço total é quantizada da seguinte maneira:

**Quadro 1: Dimensões versus Valor**

dimensões (cm³)	valor (R\$)
Até 1000	10
Entre 1001 e 10000	20
Entre 10001 e 30000	30
Entre 30001 e 100000	50
Acima 100000	Não é aceito

**Quadro 2: Peso versus multiplicador**

peso(kg)	multiplicador
Até 0.1	1
Entre 0.11 e 1	1.5
Entre 1.10 e 10	2
Entre 10.1 e 30	3
Acima de 30	Não é aceito

**Quadro 3: Rota versus multiplicador**

rota	multiplicador
RS - De Rio de Janeiro até São Paulo	1
SR - De São Paulo até Rio de Janeiro	1
BS - De Brasília até São Paulo	1.2
SB - De São Paulo até Brasília	1.2
BR - De Brasília até Rio de Janeiro	1.5
RB - Rio de Janeiro até Brasília	1.5

**Obs.:** Pode-se mudar o nome das cidades e siglas. Utilizando 3 cidades está ótimo

Elabore um programa em Python que:

1. Pergunte a altura (em cm), comprimento (em cm) e largura (em cm) do objeto. Se digitar um valor não numérico e/ou as dimensões passarem do limite aceito repetir a pergunta;
2. Pergunte o peso do objeto (em kg). Se digitar um valor não numérico e/ou o peso passar do limite aceito repetir a pergunta;
3. Pergunte a rota do objeto. Se digitar uma opção que não esteja na tabela repetir a pergunta;
4. Encerre o total a ser pago com base na equação desse enunciado;
5. Deve-se codificar uma função `dimensoesObjeto` (EXIGÊNCIA 1 de 3);
  - o Dentro da função perguntar altura do objeto (em cm);
  - o Dentro da função perguntar o comprimento do objeto (em cm);
  - o Dentro da função perguntar a largura do objeto (em cm)



- Calcular o volume (em cm) da caixa para objeto (altura \* largura \* comprimento);
  - Deve-se ter um try/except para o caso de o usuário digitar um valor não numérico;
  - Deve-se retornar o valor em (RS) conforme a Quadro 1
6. Deve-se codificar uma função pesoObjeto (EXIGÊNCIA 2 de 3);
- Dentro da função perguntar peso do objeto (em kg);
  - Deve-se ter um try/except para o caso de o usuário digitar um valor não numérico;
  - Deve-se retornar o multiplicador conforme o Quadro 2
7. Deve-se codificar uma função rotaObjeto (EXIGÊNCIA 3 de 3);
- Dentro da função perguntar a rota do objeto desejada (Sugestão: utilize as siglas para facilitar os testes);
  - OBS: PODE MUDAR O NOME DAS CIDADES E SUAS SIGLAS
  - Deve-se retornar o multiplicador conforme o Quadro 3
8. Colocar um exemplo de SAIDA DE CONSOLE uma encomenda com peso, dimensões e rota válidos
9. Colocar um exemplo de SAIDA DE CONSOLE com o tratamento de erro quando digitado um valor não numérico é digitado no campo peso ou dimensões)

Segue o exemplo de SAIDA DE CONSOLE:



Bem Vindo a Companhia de Logística Renan Portela Jorge S.A.

Digite o comprimento do objeto (em cm): 0

Você digitou alguma dimensão do objeto com valor não numérico

Por favor entre com as dimensões desejados novamente

Digite o comprimento do objeto (em cm): 100

Digite a largura do objeto (em cm): 100

Digite a altura do objeto (em cm): 100

O volume do objeto é (em cm³): 1000000.0

Não aceitamos objetos com dimensões tão grandes

Entre com as dimensões desejados novamente

Digite o comprimento do objeto (em cm): 10

Digite a largura do objeto (em cm): 10

Digite a altura do objeto (em cm): 10

O volume do objeto é (em cm³): 1000.0

Digite o peso do objeto (em kg): 0

Você digitou peso do objeto com valor não numérico

Por favor entre com o peso desejado novamente

Digite o peso do objeto (em kg): 10000

Não aceitamos objetos tão pesados

Entre com o peso desejado novamente

Digite o peso do objeto (em kg): 10

Selecione a rota:

BR - De Brasília para Rio de Janeiro

BS - De Brasília para São Paulo

RB - De Rio de Janeiro para Brasília

RS - De Rio de Janeiro para São Paulo

SR - De São Paulo para Rio de Janeiro

SB - De São Paulo para Brasília

>> 00

Você digitou uma rota que não existe

Por favor entre com a rota desejada novamente

Selecione a rota:

BR - De Brasília para Rio de Janeiro

BS - De Brasília para São Paulo

RB - De Rio de Janeiro para Brasília

RS - De Rio de Janeiro para São Paulo

SR - De São Paulo para Rio de Janeiro

SB - De São Paulo para Brasília

>> BR

Total a pagar(R\$): 90.00 (dimensões: 20 \* peso: 3 \* rota: 1.5)

**Identificador**

Tratamento de erro quando digitado um valor não numérico

Tratamento de valor inválido

Tratamento de erro quando digitado um valor não numérico

Tratamento de valor inválido

Tratamento de valor inválido

Valor total a pagar

## Apresentação do Código

```
nome = 'Emerson Vaz'
ru = 4181452
print(f'\nBem vindo a FastLog empresa de logística fundada por
{nome}\ne que já realizou mais de {ru} entregas em todo
sudeste!\n')

def dimensoesObjetos():

    while True: #BLOCO DE CÓDIGO PARA VERIFICAR A DIMENSÃO DO
OBJETO
        try:
            altura = float(input('Digite a altura do objeto (em cm):
'))
            comprimento = float(input('Digite o comprimento do objeto
(em cm): '))
```



```
    largura = float(input('Digite a largura do objeto (em
cm): '))

    volume = comprimento * largura * altura
    print(f'O volume do objeto é: {volume}')

    if volume <= 1000:
        return 10
    elif 1001 <= volume <=10000:
        return 20
    elif 10001 <= volume <=30000:
        return 30
    elif 30001 <= volume < 100000:
        return 50
    else:
        print('Não aceitamos objetos com dimensões tão
grandes')
        print('Por favor tente novamente')
        continue
except ValueError:
    print('Valor não numérico')
    print('Por favor tente novamente')
    continue

def pesoObjeto():

    while True: #BLOCO DE CÓDIGO PARA VERIFICAR A PESO DO OBJETO

        try:
            peso = float(input('Digite o peso do objeto em (Kg)\n'))

            if peso <= 0.1:
                return 1
            elif 0.11 < peso <= 1:
                return 1.5
            elif 1.10 <= peso <= 10:
                return 2
            elif 10.1 <= peso <= 30:
                return 3
            else:
                print('nao aceito')
                continue
        except ValueError:
            print('Valor não numérico')
            print('Por favor tente novamente')
            continue

def rotaObjeto():

    print('RS - De Rio de Janeiro até São Paulo')
```



```
print('SR - De São Paulo até Rio de Janeiro')
print('BS - De Brasília até São Paulo')
print('SB - De São Paulo até Brasília')
print('BR - De Brasília até Rio de Janeiro')
print('RB - De Rio de Janeiro até Brasília\n')

while True: #BLOCO DE CÓDIGO PARA VERIFICAR A ROTA QUE O OBJETO
FARÁ
    rota = str(input('Selecione a rota :')).upper()

    if rota == 'RS':
        return 1
    elif rota == 'SR':
        return 1
    elif rota == 'BS':
        return 1.2
    elif rota == 'SB':
        return 1.2
    elif rota == 'BR':
        return 1.5
    elif rota == 'RB':
        return 1.5
    else:
        print('Rota inválida')
        print('Por favor tente novamente')
        continue

Dimensoes = dimensoesObjetos()
Peso = pesoObjeto()
Rota = rotaObjeto()

total = Dimensoes * Peso * Rota
print(f'O valor total é : R${total:.2f}\nDimensoes:
{Dimensoes:.2f} * Peso {Peso:.2f} * Rota: {Rota:.2f}')
```

## Saída do Console



```
Bem vindo a FastLog empresa de logística fundada por Emerson Vaz  
e que já realizou mais de 4181452 entregas em todo sudeste!
```

```
Digite a altura do objeto (em cm): 100  
Digite o comprimento do objeto (em cm): 100  
Digite a largura do objeto (em cm): 100  
O volume do objeto é: 1000000.0  
Não aceitamos objetos com dimensões tão grandes  
Por favor tente novamente  
Digite a altura do objeto (em cm): 10  
Digite o comprimento do objeto (em cm): 10  
Digite a largura do objeto (em cm): 10  
O volume do objeto é: 1000.0  
Digite o peso do objeto em (Kg)
```

```
a
```

```
Valor não numérico  
Por favor tente novamente  
Digite o peso do objeto em (Kg)
```

```
10  
RS - De Rio de Janeiro até São Paulo  
SR - De São Paulo até Rio de Janeiro  
BS - De Brasília até São Paulo  
SB - De São Paulo até Brasília  
BR - De Brasília até Rio de Janeiro  
RB - De Rio de Janeiro até Brasília
```

```
Selecione a rota :sd  
Rota inválida  
Por favor tente novamente  
Selecione a rota :br
```



## QUESTÃO 4 de 4 (25 pontos)

### CONTEÚDO ATÉ AULA 06

Enunciado: Imagina-se que você está desenvolvendo um software de controle de estoque para uma bicicletaria. Este software deve ter o seguinte menu e opções:

1. Cadastrar Peça
2. Consultar Peça
  - 1) Consultar Todas as Peças
  - 2) Consulta Peças por Código
  - 3) Consulta Peças por Fabricante
  - 4) Retornar
3. Remover Peça
4. Sair

Elabore um programa em Python que:

1. Deve-se codificar uma função `cadastrarPeca` (código) (EXIGÊNCIA 1);
  - Essa função recebe como parâmetro um código exclusivo para cada cliente cadastrado (DICA: utilize um contador como parâmetro)
  - Dentro da função perguntar o nome da peça;
  - Dentro da função perguntar o fabricante da peça;
  - Dentro da função perguntar o valor da peça
  - Cada peça cadastrada deve ter os seus dados armazenados num DICIONÁRIO (DICA: Conferir material escrito da p. 22 até p24 da AULA 06)
2. Deve-se codificar uma função `consultarPeca` (EXIGÊNCIA 2);
  - Dentro da função ter um menu com as seguintes opções:
    - Consultar Todas as Peças
    - Consultar Peças por Código
    - Consultar Peças por Fabricante
    - Retornar
3. Deve-se codificar uma função chamada `removerPeca` (EXIGÊNCIA 3);
  - Dentro da função perguntar qual o código do produto que se deseja remover do cadastro (da lista de dicionário)
4. Colocar um exemplo de SAIDA DO CONSOLE com o cadastro de 3 (ou mais) peças. Sendo que 2 delas do mesmo fabricante – ver figura 1
5. Colocar um exemplo de SAIDA DO CONSOLE com a consulta a todas as peças cadastradas – ver figura 2
6. Colocar um exemplo de SAIDA DO CONSOLE com uma consulta por código – ver figura 3
7. Colocar um exemplo de SAIDA DO CONSOLE com uma consulta por fabricante – ver figura 4
8. Colocar um exemplo de SAIDA DO CONSOLE ao remover um cadastro e mostrando depois todos os cadastros– ver figura 5

Segue os exemplos de SAIDA DE CONSOLE:



```
Bem Vindo ao Controle de Estoque da Bicicletaria do Renan Portela Jorge
Escolha a opção desejada:
1-Cadastrar Peças
2-Consultar Peças
3-Remover Peças
4-Sair
>>1
Você Selecionou a Opção de Cadastrar Peça
Código da Peça 001 código exclusivo da peça
Por favor entre com o NOME da peça:kit cambio 21v
Por favor entre com o FABRICANTE da peça:shimano
Por favor entre com o VALOR(R$) da peça:300
Escolha a opção desejada:
1-Cadastrar Peças
2-Consultar Peças
3-Remover Peças
4-Sair
>>1
Você Selecionou a Opção de Cadastrar Peça
Código da Peça 002 código exclusivo da peça
Por favor entre com o NOME da peça:pedivela
Por favor entre com o FABRICANTE da peça:shimano
Por favor entre com o VALOR(R$) da peça:130
Escolha a opção desejada:
1-Cadastrar Peças
2-Consultar Peças
3-Remover Peças
4-Sair
>>1
Você Selecionou a Opção de Cadastrar Peça
Código da Peça 003 código exclusivo da peça
Por favor entre com o NOME da peça:pneu 27 polegadas
Por favor entre com o FABRICANTE da peça:pirelli
Por favor entre com o VALOR(R$) da peça:60
Escolha a opção desejada:
1-Cadastrar Peças
2-Consultar Peças
3-Remover Peças
4-Sair
```




Figura 1: Exemplo de Cadastrar de três peças. Perceba que 2 delas tem o mesmo fabricante.

```
Escolha a opção desejada:
1-Cadastrar Peças
2-Consultar Peças
3-Remover Peças
4-Sair
>>2
Você Selecionou a Opção de Consultar Pecas
Escolha a opção desejada:
1-Consultar Todas as Peças
2-Consultar Peças por Código
3-Consultar Peças por Fabricante
4-Retornar
>>1
-----
codigo : 1
nome : kit cambio 21v
fabricante : shimano
valor : 300.0
codigo : 2
nome : pedivela
fabricante : shimano
valor : 130.0
codigo : 3
nome : pneu 27 polegadas
fabricante : pirelli
valor : 60.0
-----
Escolha a opção desejada:
1-Consultar Todas as Peças
2-Consultar Peças por Código
3-Consultar Peças por Fabricante
4-Retornar
```

Figura 2: Exemplo de Consulta Todas as Peças.

```
Escolha a opção desejada:
```

```
1-Cadastrar Peças
```

```
2-Consultar Peças
```

```
3-Remover Peças
```

```
4-Sair
```

```
>>2
```

```
Você Selecionou a Opção de Consultar Peças
```

```
Escolha a opção desejada:
```

```
1-Consultar Todas as Peças
```

```
2-Consultar Peças por Código
```

```
3-Consultar Peças por Fabricante
```

```
4-Retornar
```

```
>>2
```

```
Digite o CÓDIGO da Peça: 2
```

```
-----  
codigo : 2  
nome : pedivela  
fabricante : shimano  
valor : 130.0  
-----
```

```
Escolha a opção desejada:
```

```
1-Consultar Todas as Peças
```

```
2-Consultar Peças por Código
```

```
3-Consultar Peças por Fabricante
```

```
4-Retornar
```

```
>>|
```

Figura 3: Exemplo de Consultar por Código.

Escolha a opção desejada:

1-Cadastrar Peças

2-Consultar Peças

3-Remover Peças

4-Sair

>>2

Você Selecionou a Opção de Consultar Peças

Escolha a opção desejada:

1-Consultar Todas as Peças

2-Consultar Peças por Código

3-Consultar Peças por Fabricante

4-Retornar

>>3

Digite o FABRICANTE da Peça: *shimano*

codigo : 1

nome : kit cambio 21v

fabricante : shimano

valor : 300.0

-----

codigo : 2

nome : pedivela

fabricante : shimano

valor : 130.0

-----

Escolha a opção desejada:

1-Consultar Todas as Peças

2-Consultar Peças por Código

3-Consultar Peças por Fabricante

4-Retornar

Figura 4: Exemplo de Consultar Peças por Fabricante

```
Escolha a opção desejada:
```

```
1-Cadastrar Peças
```

```
2-Consultar Peças
```

```
3-Remover Peças
```

```
4-Sair
```

```
>>3
```

```
Você Selecionou a Opção de Remover Peça
```

```
Digite o código da peça a ser removida: 1
```

```
Escolha a opção desejada:
```

```
1-Cadastrar Peças
```

```
2-Consultar Peças
```

```
3-Remover Peças
```

```
4-Sair
```

```
>>2
```

```
Você Selecionou a Opção de Consultar Peças
```

```
Escolha a opção desejada:
```

```
1-Consultar Todas as Peças
```

```
2-Consultar Peças por Código
```

```
3-Consultar Peças por Fabricante
```

```
4-Retornar
```

```
>>1
```

```
-----  
codigo : 2
```

```
nome : pedivela
```

```
fabricante : shimano
```

```
valor : 130.0
```

```
codigo : 3
```

```
nome : pneu 27 polegadas
```

```
fabricante : pirelli
```

```
valor : 60.0  
-----
```

```
Escolha a opção desejada:
```

```
1-Consultar Todas as Peças
```

```
2-Consultar Peças por Código
```

```
3-Consultar Peças por Fabricante
```

```
4-Retornar
```

```
>>
```

Perceba que a peça de  
código 1 foi removida

Figura 5: Exemplo de Remover Peça do cadastro e depois Consultar Todas as Peças. Veja que a peça de código 1 foi removida e não aparece mais no sistema.



### Apresentação do Código:

```
listaPecas = []

# ----- COMEÇO cadastrarPeça -----
def cadastrarPeca(codigo):
    print('Bem Vindo(a) ao cadastro de Peças')
    print(f'Codigo da Peça {codigo}')
    nome = input("Digite o nome da peça: ")
    fabricante = input("Digite o nome do fabricante da peça: ")
    valor = float(input("Digite o valor da peça R$: "))
    dicionarioPecas = {'codigo': codigo,
                       'nome': nome,
                       'fabricante': fabricante,
                       'valor': valor}
    listaPecas.append(dicionarioPecas.copy())

# ----- FIM cadastrarPeça -----

# ----- COMEÇO consultarPeça -----

def consultarPeca():
    while True:
        try:
            print("Bem Vindo(a) a consulta de Peças")
            opcaoConsultar = int(input("Entre com a opção desejada:\n"
                                     "1-Consultar todas as peças\n"
                                     "2-Consultar peças por código\n"
                                     "3-Consultar peças por fabricante\n"
                                     "4-Retornar\n"
                                     "\n>>>"))

            if opcaoConsultar == 1:
                print("Você selecionou a opção consultar todas as peças")
                for pecas in listaPecas:
                    for key, value in pecas.items():
                        print(f'{key} : {value}')

            elif opcaoConsultar == 2:
                print("Você selecionou a opção consultar peças por código")
                entrada = int(input("Digite o código desejado:"))
```



```
)
    for pecas in listaPecas:
        if (pecas['codigo'] == entrada):
            for key, value in pecas.items():
                print(f'{key} : {value}')

    elif opcaoConsultar == 3:
        print("Você selecionou a opção consultar peças
pelo fabricante")
        entrada = input("Digite o fabricante desejado: ")
        for pecas in listaPecas:
            if (pecas['fabricante'] == entrada):
                for key, value in pecas.items():
                    print(f'{key} : {value}')

    elif opcaoConsultar == 4:
        return

    else:
        print('Você digitou uma opção inválida, favor
entrar com uma das opções acima')
        continue

except ValueError:
    print('Pare de digitar valores não inteiros')

# ----- FIM consultarPeça -----

# ----- COMEÇO removerPeça -----

def removerPeca():
    print('Gostaria de remover de Peças?')
    entrada = int(input('Digite o código da peça que deseja
remover: '))
    for pecas in listaPecas:
        if (pecas['codigo'] == entrada):
            listaPecas.remove(pecas)

# ----- FIM removerPeça -----

# ----- COMEÇO Main -----
nome = "Emerson Vaz"
ru = 4191452
print(f'Bem Vindo(a) ao controle de estoque da Bicicletaria do
{nome} RU:{ru}\n')
registroPecas = 0
while True:
    try:
        opcao = int(input('Digite a opção desejada:\n'))
```



```
        '1-Cadastrar peça\n'  
        '2-Consultar Peça\n'  
        '3-Remover Peça\n'  
        '4-Sair'\n        '\n>>>'))\n\nif opcao == 1:\n    registroPecas = registroPecas + 1\n    cadastrarPeca(registroPecas)\nelif opcao == 2:\n    consultarPeca()\nelif opcao == 3:\n    removerPeca()\nelif opcao == 4:\n    print('Programa finalizado')\n    break\nelse:\n    print('Pare de digitar números que não existe no\nmenu')\n    continue\nexcept ValueError:\n    print('Pare de digitar valores não inteiros')\n# ----- FIM Main -----
```

## Saída do Console





```
D:\pythonProject_UNINTER\venv\Scripts\python.exe D:\pythonProject_UNINTER\EX005.py
Bem Vindo(a) ao controle de estoque da Bicicletaria do Emerson Vaz RU:4191452

Digite a opção desejada:
1-Cadastrar peça
2-Consultar Peça
3-Remover Peça
4-Sair
>>>1
Bem Vindo(a) ao cadastro de Peças
Codigo da Peça 1
Digite o nome da peça: kit cambio 21v
Digite o nome do fabricante da peça: shimano
Digite o valor da peça R$: 300
Digite a opção desejada:
1-Cadastrar peça
2-Consultar Peça
3-Remover Peça
4-Sair
>>>1
Bem Vindo(a) ao cadastro de Peças
Codigo da Peça 2
Digite o nome da peça: pedivela
Digite o nome do fabricante da peça: shimano
Digite o valor da peça R$: 130
Digite a opção desejada:
1-Cadastrar peça
2-Consultar Peça
3-Remover Peça
4-Sair
>>>1
Bem Vindo(a) ao cadastro de Peças
Codigo da Peça 3
Digite o nome da peça: pneu 27 polegadas
Digite o nome do fabricante da peça: pirelli
Digite o valor da peça R$: 60
Digite a opção desejada:
1-Cadastrar peça
2-Consultar Peça
```



```
3-Remover Peça
4-Sair
>>>2
Bem Vindo(a) a consulta de Peças
Entre com a opção desejada:
1-Consultar todas as peças
2-Consultar peças por código
3-Consultar peças por fabricante
4-Retornar

>>>1
Você selecionou a opção consultar todas as peças
codigo : 1
nome : kit cambio 21v
fabricante : shimano
valor : 300.0
codigo : 2
nome : pedivela
fabricante : shimano
valor : 130.0
codigo : 3
nome : pneu 27 polegadas
fabricante : pirelli
valor : 60.0
Bem Vindo(a) a consulta de Peças
Entre com a opção desejada:
1-Consultar todas as peças
2-Consultar peças por código
3-Consultar peças por fabricante
4-Retornar

>>>2
Você selecionou a opção consultar peças por código
Digite o código desejado: 2
codigo : 2
nome : pedivela
```



```
fabricante : shimano
valor : 130.0
Bem Vindo(a) a consulta de Peças
Entre com a opção desejada:
1-Consultar todas as peças
2-Consultar peças por código
3-Consultar peças por fabricante
4-Retornar

>>>4
Digite a opção desejada:
1-Cadastrar peça
2-Consultar Peça
3-Remover Peça
4-Sair
>>>2
Bem Vindo(a) a consulta de Peças
Entre com a opção desejada:
1-Consultar todas as peças
2-Consultar peças por código
3-Consultar peças por fabricante
4-Retornar

>>>3
Você selecionou a opção consultar peças pelo fabricante
Digite o fabricante desejado: shimano
codigo : 1
nome : kit cambio 21v
fabricante : shimano
valor : 300.0
codigo : 2
nome : pedivela
fabricante : shimano
valor : 130.0
Bem Vindo(a) a consulta de Peças
Entre com a opção desejada:
```



```
1-Consultar todas as peças
2-Consultar peças por código
3-Consultar peças por fabricante
4-Retornar

>>>4
Digite a opção desejada:
1-Cadastrar peça
2-Consultar Peça
3-Remover Peça
4-Sair
>>>3
Gostaria de remover de Peças?
Digite o código da peça que deseja remover: 1
Digite a opção desejada:
1-Cadastrar peça
2-Consultar Peça
3-Remover Peça
4-Sair
>>>2
Bem Vindo(a) a consulta de Peças
Entre com a opção desejada:
1-Consultar todas as peças
2-Consultar peças por código
3-Consultar peças por fabricante
4-Retornar

>>>1
Você selecionou a opção consultar todas as peças
codigo : 2
nome : pedivela
fabricante : shimano
valor : 130.0
codigo : 3
nome : pneu 27 polegadas
fabricante : pirelli
valor : 60.0
Bem Vindo(a) a consulta de Peças
Entre com a opção desejada:
1-Consultar todas as peças
2-Consultar peças por código
3-Consultar peças por fabricante
4-Retornar

>>>
```