

ATIVIDADE PRÁTICA:

Lógica de Programação e Algoritmos

**ANO**

**2023**

|  |
| --- |
| **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:**    Colocar o seu nome  **OBS: Para os números decimais ficarem com somente duas casas depois da vírgula utilize {:.2f). Exemplo:**  print('O valor sem desconto foi: R$ {:.2f}' .format(sub\_total)) |

|  |
| --- |
| **Apresentação do Código (FORMATO TEXTO)** |
| print("Bem Vindo a Loja de Alisson de Souza Rodrigues")    valor\_produto = float(input('Insira o valor do produto: '))  qtd\_produto = int(input('Insira a quantidade do produto: '))    total = 0 # valor total sem desconto  total\_desconto = 0 # valor total com desconto    if qtd\_produto <= 9:  # desconto de 0% para até 9 itens  total = valor\_produto \* qtd\_produto  total\_desconto = valor\_produto \* qtd\_produto    print('o valor total da venda sem desconto é: R${:.2f}'.format(total))  print('o valor total da venda com desconto é: R${:.2f}'.format(total\_desconto))    elif qtd\_produto <= 99:  # desconto de 5% para 10 até 99 itens  desconto = 0.95  total = valor\_produto \* qtd\_produto  total\_desconto = valor\_produto \* desconto \* qtd\_produto    print('o valor total da venda sem desconto é: R${:.2f}'.format(total))  print('o valor total da venda com desconto é: R${:.2f} (desconto 5%)'.format(total\_desconto))    elif qtd\_produto <= 999:  # desconto de 10% para 100 até 999 itens  desconto = 0.9  total = valor\_produto \* qtd\_produto  total\_desconto = valor\_produto \* desconto \* qtd\_produto    print('o valor total da venda sem desconto é: R${:.2f}'.format(total))  print('o valor total da venda com desconto é: R${:.2f} (desconto 10%)'.format(total\_desconto))    else:  # desconto de 15% para acima de 1000 itens  desconto = 0.85  total = valor\_produto \* qtd\_produto  total\_desconto = valor\_produto \* desconto \* qtd\_produto    print('o valor total da venda sem desconto é: R${:.2f}'.format(total))  print('o valor total da venda com desconto é: R${:.2f} (desconto 15%)'.format(total\_desconto)) |

|  |
| --- |
| **Saída do Console (FORMATO IMAGEM – PRINT DA TELA)** |
|  |

|  |
| --- |
| **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:**    **Figura: 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 (FORMATO TEXTO)** |
| print('Bem Vindo a Lanchonete de Alisson de Souza Rodrigues')    print('=' \* 20 + ' CARDÁPIO ' + '=' \* 20)  print('| Código |' + ' Descrição ' + '| Valor (R$) |')  print('| 100 |' + ' Cachorro-Quente ' + '| 09,00 |')  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 |' + ' Refrigerente Lata ' + '| 05,00 |')  print('| 201 |' + ' Chá Gelado ' + '| 04,00 |')    pedido = 0 # código do produto solicitado  conta = 0 # valor total da conta    while True:  pedido = int(input('Entre com o código do produto desejado: '))    if pedido == 100:  print('Você pediu um Cachorro-Quente no valor de R$09,00')  conta += 9 # adiciona o valor do produto escolhido ao valor total da conta    resposta = int(input('Deseja pedir mais alguma coisa? \n 1 - Sim \n 0 - Não \n'))  if resposta == 1:  # volta para o início do laço  continue  elif resposta == 0:  # termina o laço  break  elif pedido == 101:  print('Você pediu um Cachorro-Quente Duplo no valor de R$11,00')  conta += 11    resposta = int(input('Deseja pedir mais alguma coisa? \n 1 - Sim \n 0 - Não \n'))  if resposta == 1:  continue  elif resposta == 0:  break  elif pedido == 102:  print('Você pediu um X-Egg no valor de R$12,00')  conta += 12    resposta = int(input('Deseja pedir mais alguma coisa? \n 1 - Sim \n 0 - Não \n'))  if resposta == 1:  continue  elif resposta == 0:  break  elif pedido == 103:  print('Você pediu um X-Salada no valor de R$13,00')  conta += 13    resposta = int(input('Deseja pedir mais alguma coisa? \n 1 - Sim \n 0 - Não \n'))  if resposta == 1:  continue  elif resposta == 0:  break  elif pedido == 104:  print('Você pediu um X-Bacon no valor de R$14,00')  conta += 14    resposta = int(input('Deseja pedir mais alguma coisa? \n 1 - Sim \n 0 - Não \n'))  if resposta == 1:  continue  elif resposta == 0:  break  elif pedido == 105:  print('Você pediu um X-Tudo no valor de R$17,00')  conta += 17    resposta = int(input('Deseja pedir mais alguma coisa? \n 1 - Sim \n 0 - Não \n'))  if resposta == 1:  continue  elif resposta == 0:  break  elif pedido == 200:  print('Você pediu um Refrigerante Lata no valor de R$5,00')  conta += 5    resposta = int(input('Deseja pedir mais alguma coisa? \n 1 - Sim \n 0 - Não \n'))  if resposta == 1:  continue  elif resposta == 0:  break  elif pedido == 201:  print('Você pediu um Chá Gelado no valor de R$04,00')  conta += 4    resposta = int(input('Deseja pedir mais alguma coisa? \n 1 - Sim \n 0 - Não \n'))  if resposta == 0:  break  elif resposta == 1:  continue  else:  # volta para o início do laço caso um código de produto inexistente for digitado  print('Opção Inválida')  continue    print('\nSua conta deu R${:.2f}'.format(conta)) |

|  |
| --- |
| **Saída do Console (FORMATO IMAGEM – PRINT DA TELA)** |
|  |
| **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:**  **que você e sua equipe**  **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$) | | volume < 1000 | **10** | | 1000 <= volume < 10000 | **20** | | 10000 <= volume < 30000 | **30** | | 30000 <= volume < 100000 | **50** | | volume >= 100000 | **Não é aceito** | | **Quadro 2: Peso versus multiplicador**   |  |  | | --- | --- | | peso(kg) | multiplicador | | peso <= 0.1 | **1** | | 0.1 <= peso < 1 | **1.5** | | 1 <= peso < 10 | **2** | | 10 <= peso < 30 | **3** | | peso => 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);**    * **Dentro da função perguntar altura do objeto (em cm);**    * **Dentro da função perguntar o comprimento do objeto (em cm);**    * **Dentro da função perguntar a largura do objeto (em cm)**    * **Calcular o volume (em cm) da caixa p/a objeto (altura\*largura\*comprimento);**    * **Deve-se ter try/except para o caso do 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:**  Colocar o seu nome    **Figura: Exemplo de programa com tratamento de valor e erro de digitação** |
| **Apresentação do Código (FORMATO TEXTO)** |
| print('Bem Vindo ao Centro Logístico Alisson de Souza Rodrigues Ltda')  def dimensoesObjeto():  while True:  try:  # variávies para o cálculo do volume do objeto  altura = float(input('Insira a altura do objeto (em cm): '))  comprimento = float(input('Insira o comprimento do objeto (em cm): '))  largura = float(input('Insira a largura do objeto (em cm): '))    volume = altura \* comprimento \* largura # volume do objeto  if volume >= 100000:  # volta para o início do laço caso o volume do objeto seja maior que 100000 cm³  print('O volume do objeto é {} cm³, não aceitamos objetos com volume tão grande.'.format(volume))  continue  print('O volume do objeto é {} cm³'.format(volume))    # retorna o preço conforme o volume  if volume < 1000:  return 10  elif volume < 10000:  # volume de 1000 a 9999 cm³  return 20  elif volume < 30000:  # volume de 10000 a 29999 cm³  return 30  elif volume < 100000:  # volume de 30000 a 99999 cm³  return 50    break # encerra o looping infinito  except:  print('Insira um valor válido')  continue # retorna para o início do laço em caso de valor inválido    def pesoObjeto():  while True:  try:  peso = float(input('Insira o peso do objeto (em kg): '))    print('O objeto tem {} kg '.format(peso))    if peso >= 30:  # volta para o início do laço caso o objeto tenha 30 kg ou mais  print('Objeto muito pesado, tente novamente')  continue    # retorna o multiplicador conforme o peso  if peso < 0.1:  return 1  elif peso < 1:  # peso de 0.1 a 0.999 kg  return 1.5  elif peso < 10:  # peso de 1 a 9.999 kg  return 2  elif peso < 30:  # peso de 10 a 29.999 kg  return 3  break  except:  print('Insira um valor válido')  continue # volta para o início do laço em caso de valor inválido    def rotaObjeto():  while True:  print('Escolha a rota do objeto:')  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')    rota = input()    # retorna o multiplicador de cada rota  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('Insira uma rota válida')  continue    break    total = dimensoesObjeto() \* pesoObjeto() \*rotaObjeto()  print('O total a ser pago é R${}'.format(total)) |

|  |
| --- |
| **Saída do Console (FORMATO IMAGEM – PRINT DA TELA)** |
|  |

|  |
| --- |
| **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 peça 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:**    **Figura 1: Exemplo de Cadastrar de três peças. Perceba que 2 delas tem o mesmo fabricante.**    **Figura 2: Exemplo de Consulta Todas as Peças.**    **Figura 3: Exemplo de Consultar por Código.**    **Figura 4: Exemplo de Consultar Peças por Fabricante**    **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 (FORMATO TEXTO)** |
| print('Bem Vindo ao Controle de Estoque da Bicicletaria do Alisson de Souza Rodrigues')  estoque = [] # lista que armazena cada peça  def cadastrarPeca(codigo):  peca = {'codigo': '000', 'nome':' ', 'fabricante': ' ', 'valor': 0} # formato padrão de cadastro de peça    peca['codigo'] = '%03d' % codigo # preenche com zeros a esquerda do número para manter o padrão de três dígitos  peca['nome'] = input('Digite o nome da peça: ')  peca['fabricante'] = input('Digite o fabricante da peça: ')  peca['valor'] = input('Digite o valor (R$) da peça: ')  estoque.append(peca) # adiciona o dicionário 'peca' sempre ao final da lista 'estoque''  print('Peça cadastrada')  print('O código desta peça é: ' + peca['codigo'])    def consultarPeca():  while True:  try:  print('Escolha o tipo de consulta: ')  print('1 - Todas as peças')  print('2 - Código da peça')  print('3 - Peça por frabricante')  print('4 - Retornar')    opcaoEscolhida = int(input())    if opcaoEscolhida == 1:  for i in estoque:  # 'i' recebe cada dicionário presente em 'estoque'  print('-' \* 15)  for j, k in i.items():  # j e k recebe respectivamente chave e valor presente em cada dicionário  print(j + ' : ' + k)  elif opcaoEscolhida == 2:  codigo = int(input('Insira o código da peça para consulta: '))  pecaEncontrada = 0 # recebe o valor 1 se a peça for encontrada  for i in estoque:  if int(i['codigo']) == codigo:  # consulta todos os dicionários presentes no estoque e faz o print da peça com o código indicado  pecaEncontrada = 1  for j, k in i.items():  print(j + ' : ' + k)  if pecaEncontrada == 0:  print('Peça não encontrada')    elif opcaoEscolhida == 3:  fabricante = input('Insira o fabricante da peça para consulta: ')  for i in estoque:  # converte a string em letras minúsculas para facilitar a busca pelo fabricante  if (i['fabricante']).lower() == (fabricante).lower():  print('-' \* 15)  for j, k in i.items():  print(j + ' : ' + k)  else:  print('Peça não encontrada')  elif opcaoEscolhida == 4:  break  except:  continue    def removerPeca():  pecaRemover = int(input('Digite o código da peça que deseja remover: '))  for i in estoque:  if int(i['codigo']) == pecaRemover:  estoque.remove(i) # deleta o dicionário da lista    while True:  try:  print('Escolha a opção desejada: ')  print('1 - Cadastrar peças')  print('2 - Consultar peças')  print('3 - Remover peças')  print('4 - Sair')    opcaoEscolhida = int(input())      if opcaoEscolhida == 1:  codigo = len(estoque)  # determina o código da peça com base na quantidade itens no estoque  cadastrarPeca(codigo + 1) # '+ 1' impede o cadastro com o código 000  continue  elif opcaoEscolhida == 2:  consultarPeca()  continue  elif opcaoEscolhida == 3:  removerPeca()  continue  elif opcaoEscolhida == 4:  break  else:  # caso um valor numérico não válido for digitado  print('Escolha uma opção válida')  continue  except:  # caso um valor não numérico for digitado  print('Escolha um valor numérico válido')  continue    break |

|  |
| --- |
| **Saída do Console (FORMATO IMAGEM – PRINT DA TELA)** |
| **Figura 1: Cadastro de três peças com duas do mesmo fabricante.**    **Figura 2: Consulta de todas as peças.**    **Figura 3: Consulta por código.**    **Figura 4: Consulta por Fabricante**    **Figura 5: Remoção de peça do cadastro e consulta de todas as peças.** |