**UNIVERSIDADE UNIGRANRIO | AFYA**

CURSO DE TECNÓLOGO EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

JEFFERSON DOS SANTOS ALVES LIMA 1009415

[JEFFERSON.LIMA@UNIGRANRIO.BR](mailto:jefferson.lima@unigranrio.br)

**PRÁTICA DE PROGRAMAÇÃO EM SISTEMAS DE INFORMAÇÃO**

Programas em Python

DUQUE DE CAXIAS

2023

JEFFERSON DOS SANTOS ALVES LIMA 1009415

[JEFFERSON.LIMA@UNIGRANRIO.BR](mailto:jefferson.lima@unigranrio.br)

**PRÁTICA DE PROGRAMAÇÃO EM SISTEMAS DE INFORMAÇÃO**

Mercado de trabalho para desenvolvedores

Trabalho sobre a elaboração de programas em Python da disciplina prática de programação em sistemas de informação.

Orientador: Prof.: Oswaldo Borges Peres.

DUQUE DE CAXIAS

2023

SUMÁRIO

[**1.** **PROGRAMA 1** 3](#_Toc130123469)

[**2.** **PROGRAMA 2** 4](#_Toc130123470)

[**3.** **PROGRAMA 3** 5](#_Toc130123471)

[**4.** **PROGRAMA 4** 6](#_Toc130123472)

[**5.** **PROGRAMA 5** 8](#_Toc130123473)

[**6.** **PROGRAMA 6** 10](#_Toc130123474)

[**7.** **PROGRAMA 7** 12](#_Toc130123475)

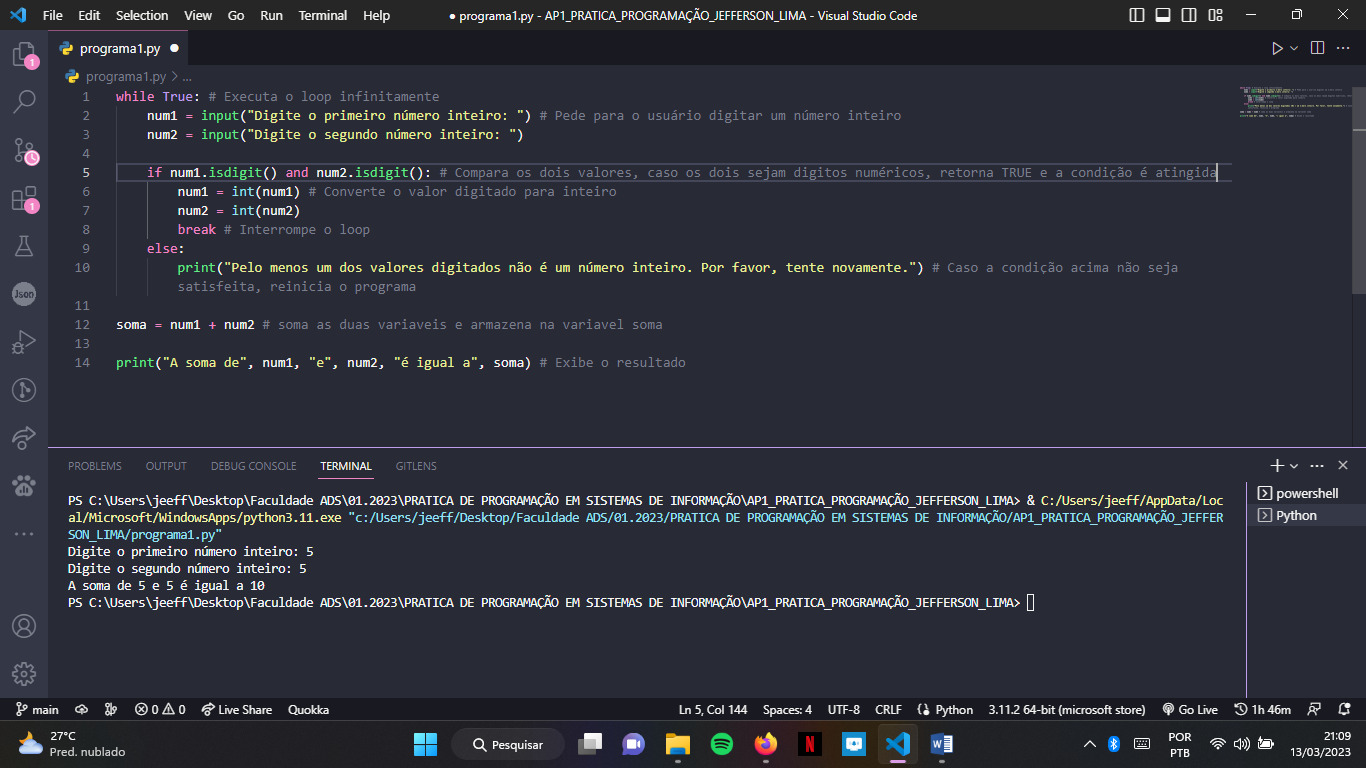
[**8.** **PROGRAMA 8** 13](#_Toc130123476)

[**9.** **PROGRAMA 9** 15](#_Toc130123477)

[**10.** **PROGRAMA 10** 17](#_Toc130123478)

[**11.** **CONCLUSÃO** 19](#_Toc130123479)

1. **PROGRAMA 1**



Código do primeiro programa:

while True: # Executa o loop infinitamente

    num1 = input("Digite o primeiro número inteiro: ") # Pede para o usuário digitar um número inteiro

    num2 = input("Digite o segundo número inteiro: ")

    if num1.isdigit() and num2.isdigit(): # Compara os dois valores, caso os dois sejam digitos numéricos, retorna TRUE e a condição é atingida

        num1 = int(num1) # Converte o valor digitado para inteiro

        num2 = int(num2)

        break # Interrompe o loop

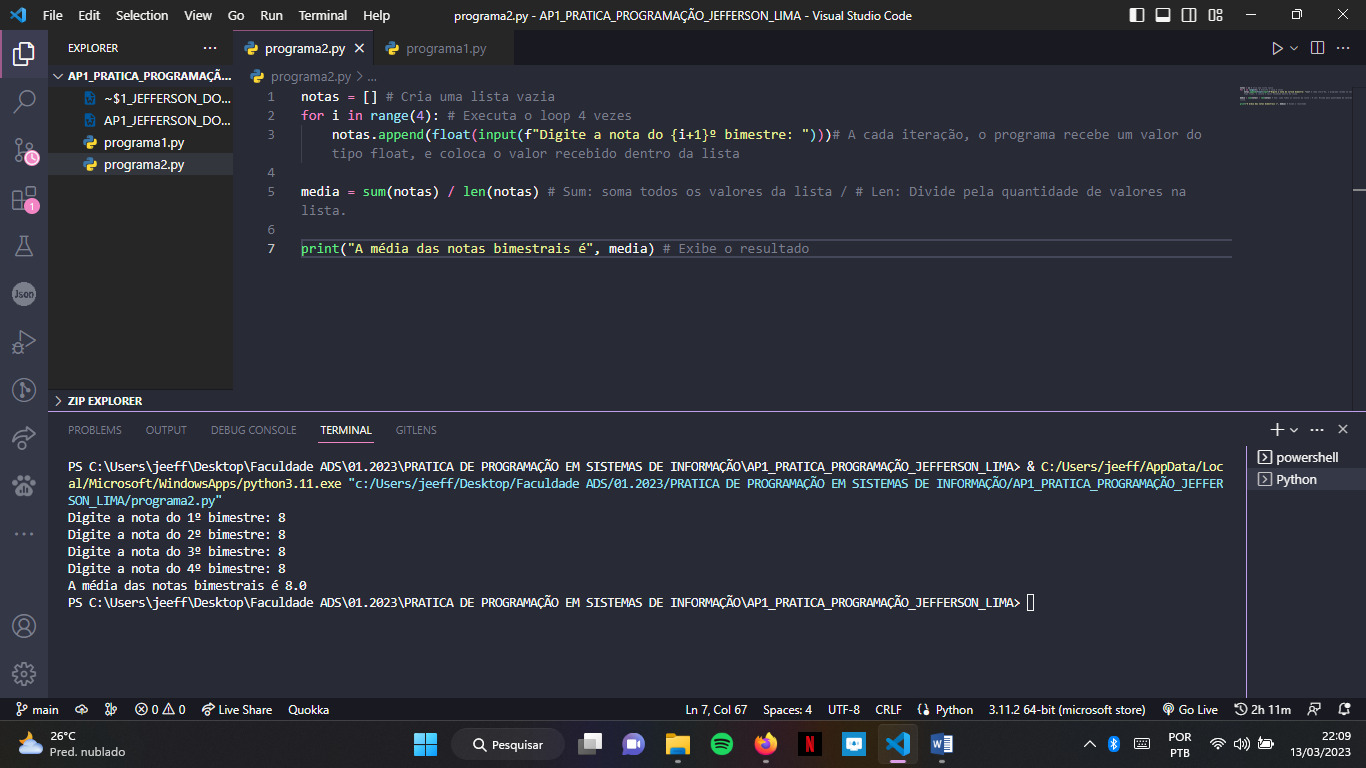
    else:

        print("Pelo menos um dos valores digitados não é um número inteiro. Por favor, tente novamente.") # Caso a condição acima não seja satisfeita, reinicia o programa

soma = num1 + num2 # soma as duas variaveis e armazena na variavel soma

print("A soma de", num1, "e", num2, "é igual a", soma) # Exibe o resultado

1. **PROGRAMA 2**

****

Código do segundo programa:

notas = [] # Cria uma lista vazia

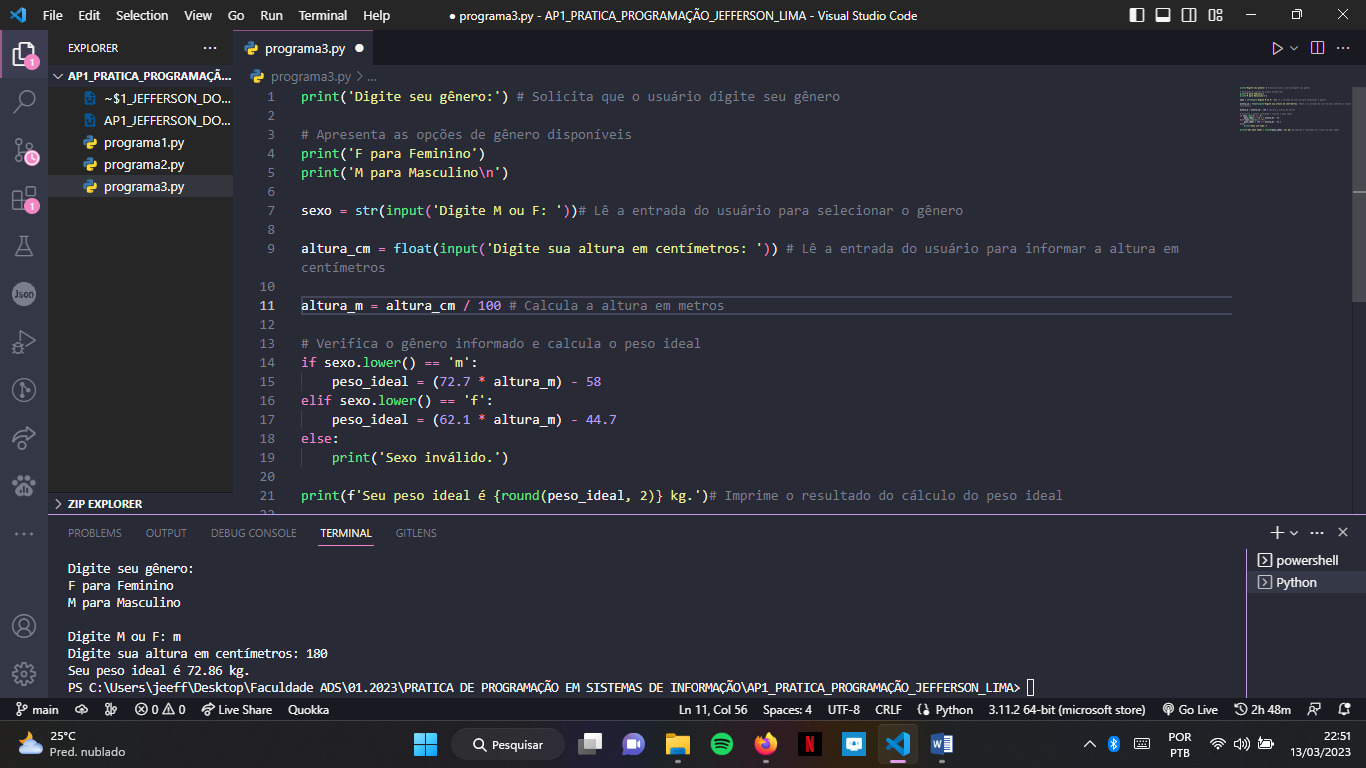
for i in range(4): # Executa o loop 4 vezes

    notas.append(float(input(f"Digite a nota do {i+1}º bimestre: ")))# A cada iteração, o programa recebe um valor do tipo float, e coloca o valor recebido dentro da lista

media = sum(notas) / len(notas) # Sum: soma todos os valores da lista / # Len: Divide pela quantidade de valores na lista.

print("A média das notas bimestrais é", media) # Exibe o resultado

1. **PROGRAMA 3**



Código do terceiro programa:

print('Digite seu gênero:') # Solicita que o usuário digite seu gênero

# Apresenta as opções de gênero disponíveis

print('F para Feminino')

print('M para Masculino\n')

sexo = str(input('Digite M ou F: '))# Lê a entrada do usuário para selecionar o gênero

altura\_cm = float(input('Digite sua altura em centímetros: ')) # Lê a entrada do usuário para informar a altura em centímetros

altura\_m = altura\_cm / 100 # Calcula a altura em metros

# Verifica o gênero informado e calcula o peso ideal

if sexo.lower() == 'm':

    peso\_ideal = (72.7 \* altura\_m) - 58

elif sexo.lower() == 'f':

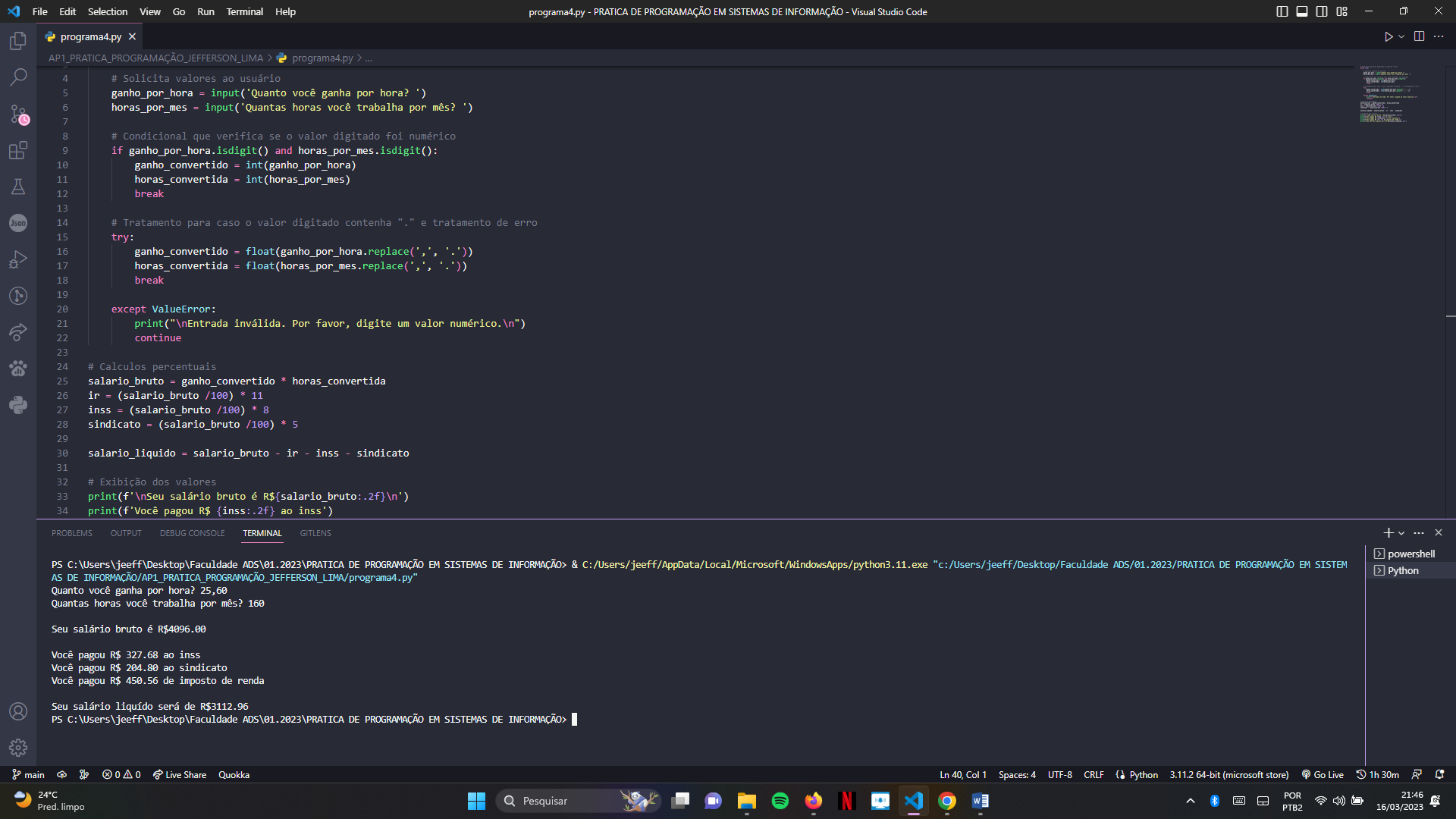
    peso\_ideal = (62.1 \* altura\_m) - 44.7

else:

    print('Sexo inválido.')

print(f'Seu peso ideal é {round(peso\_ideal, 2)} kg.')# Imprime o resultado do cálculo do peso ideal

1. **PROGRAMA 4**



Código do quarto programa:

# Loop para não parar aplicação em caso de "erro"

while True:

    # Solicita valores ao usuário

    ganho\_por\_hora = input('Quanto você ganha por hora? ')

    horas\_por\_mes = input('Quantas horas você trabalha por mês? ')

    # Condicional que verifica se o valor digitado foi numérico

    if ganho\_por\_hora.isdigit() and horas\_por\_mes.isdigit():

        ganho\_convertido = int(ganho\_por\_hora)

        horas\_convertida = int(horas\_por\_mes)

        break

    # Tratamento para caso o valor digitado contenha "." e tratamento de erro

    try:

        ganho\_convertido = float(ganho\_por\_hora.replace(',', '.'))

        horas\_convertida = float(horas\_por\_mes.replace(',', '.'))

        break

    except ValueError:

        print("\nEntrada inválida. Por favor, digite um valor numérico.\n")

        continue

# Calculos percentuais

salario\_bruto = ganho\_convertido \* horas\_convertida

ir = (salario\_bruto /100) \* 11

inss = (salario\_bruto /100) \* 8

sindicato = (salario\_bruto /100) \* 5

salario\_liquido = salario\_bruto - ir - inss - sindicato

# Exibição dos valores

print(f'\nSeu salário bruto é R${salario\_bruto:.2f}\n')

print(f'Você pagou R$ {inss:.2f} ao inss')

print(f'Você pagou R$ {sindicato:.2f} ao sindicato')

print(f'Você pagou R$ {ir:.2f} de imposto de renda\n')

print(f'Seu salário liquído será de R${salario\_liquido:.2f}')

print(f'\nSeu salário bruto é R${salario\_bruto:.2f}\n')

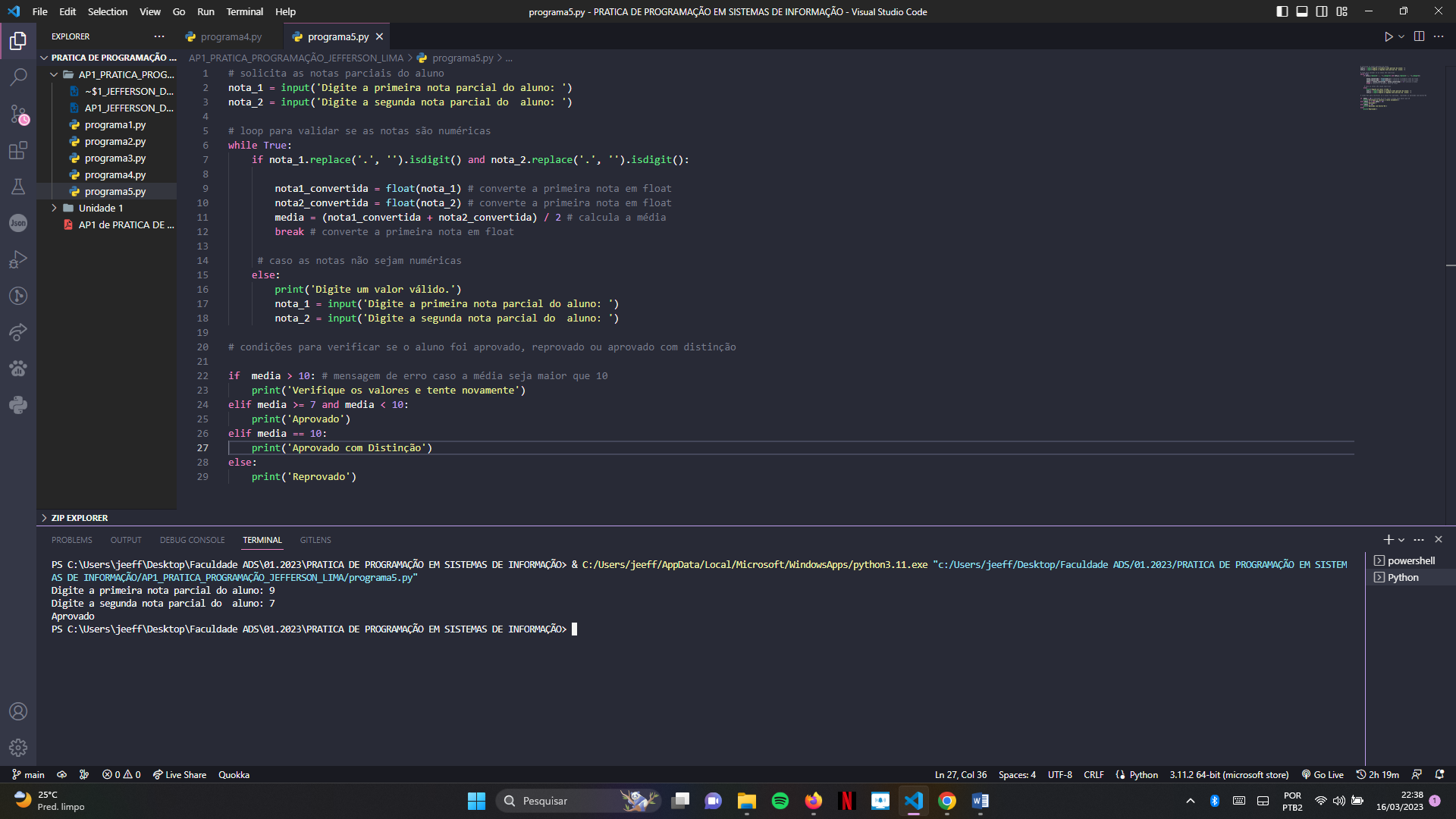
print(f'Você pagou R$ {inss:.2f} ao inss')

print(f'Você pagou R$ {sindicato:.2f} ao sindicato')

print(f'Você pagou R$ {ir:.2f} de imposto de renda\n')

print(f'Seu salário liquído será de R${salario\_liquido:.2f}')

1. **PROGRAMA 5**



Código do quinto programa:

# solicita as notas parciais do aluno

nota\_1 = input('Digite a primeira nota parcial do aluno: ')

nota\_2 = input('Digite a segunda nota parcial do  aluno: ')

# loop para validar se as notas são numéricas

while True:

    if nota\_1.replace('.', '').isdigit() and nota\_2.replace('.', '').isdigit():

        nota1\_convertida = float(nota\_1) # converte a primeira nota em float

        nota2\_convertida = float(nota\_2) # converte a primeira nota em float

        media = (nota1\_convertida + nota2\_convertida) / 2 # calcula a média

        break # converte a primeira nota em float

     # caso as notas não sejam numéricas

    else:

        print('Digite um valor válido.')

        nota\_1 = input('Digite a primeira nota parcial do aluno: ')

        nota\_2 = input('Digite a segunda nota parcial do  aluno: ')

# condições para verificar se o aluno foi aprovado, reprovado ou aprovado com distinção

if  media > 10: # mensagem de erro caso a média seja maior que 10

    print('Verifique os valores e tente novamente')

elif media >= 7 and media < 10:

    print('Aprovado')

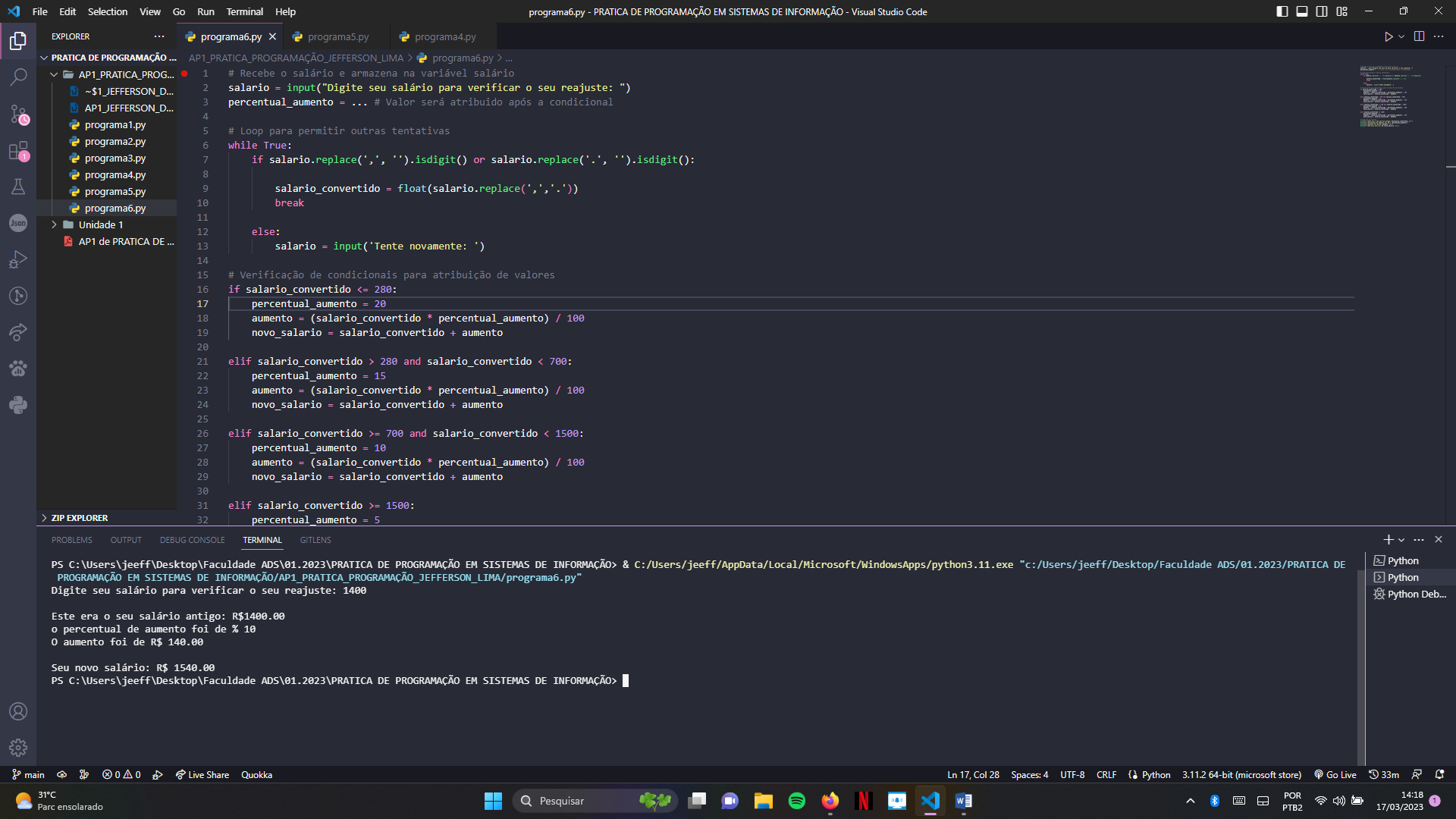
elif media == 10:

    print('Aprovado com Distinção')

else:

    print('Reprovado')

1. **PROGRAMA 6**



Código do sexto programa:

# Recebe o salário e armazena na variável salário

salario = input("Digite seu salário para verificar o seu reajuste: ")

percentual\_aumento = ... # Valor será atribuido após a condicional

# Loop para permitir outras tentativas

while True:

    if salario.replace(',', '').isdigit() or salario.replace('.', '').isdigit():

        salario\_convertido = float(salario.replace(',','.'))

        break

    else:

        salario = input('Tente novamente: ')

# Verificação de condicionais para atribuição de valores

if salario\_convertido <= 280:

    percentual\_aumento = 20

    aumento = (salario\_convertido \* percentual\_aumento) / 100

    novo\_salario = salario\_convertido + aumento

elif salario\_convertido > 280 and salario\_convertido < 700:

    percentual\_aumento = 15

    aumento = (salario\_convertido \* percentual\_aumento) / 100

    novo\_salario = salario\_convertido + aumento

elif salario\_convertido >= 700 and salario\_convertido < 1500:

    percentual\_aumento = 10

    aumento = (salario\_convertido \* percentual\_aumento) / 100

    novo\_salario = salario\_convertido + aumento

elif salario\_convertido >= 1500:

    percentual\_aumento = 5

    aumento = (salario\_convertido \* percentual\_aumento) / 100

    novo\_salario = salario\_convertido + aumento

# Exibição de Valores

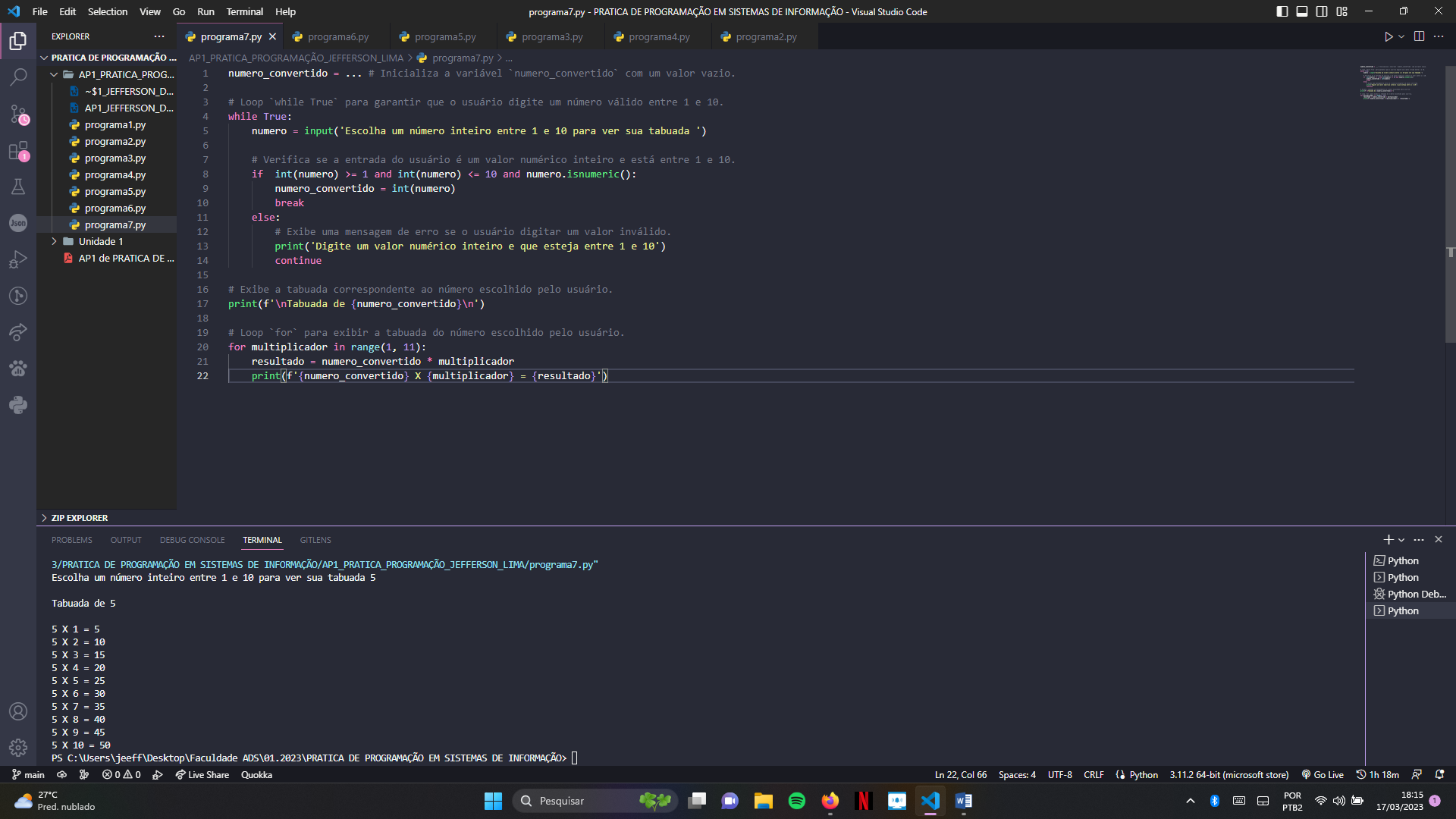
print(f'\nEste era o seu salário antigo: R${salario\_convertido:.2f}')

print(f'o percentual de aumento foi de % {percentual\_aumento}')

print(f'O aumento foi de R$ {aumento:.2f}')

print(f'\nSeu novo salário: R$ {novo\_salario:.2f}')

1. **PROGRAMA 7**



Código do sétimo programa:

numero\_convertido = ... # Inicializa a variável `numero\_convertido` com um valor vazio.

# Loop `while True` para garantir que o usuário digite um número válido entre 1 e 10.

while True:

    numero = input('Escolha um número inteiro entre 1 e 10 para ver sua tabuada ')

    # Verifica se a entrada do usuário é um valor numérico inteiro e está entre 1 e 10.

    if  int(numero) >= 1 and int(numero) <= 10 and numero.isnumeric():

        numero\_convertido = int(numero)

        break

    else:

        # Exibe uma mensagem de erro se o usuário digitar um valor inválido.

        print('Digite um valor numérico inteiro e que esteja entre 1 e 10')

        continue

# Exibe a tabuada correspondente ao número escolhido pelo usuário.

print(f'\nTabuada de {numero\_convertido}\n')

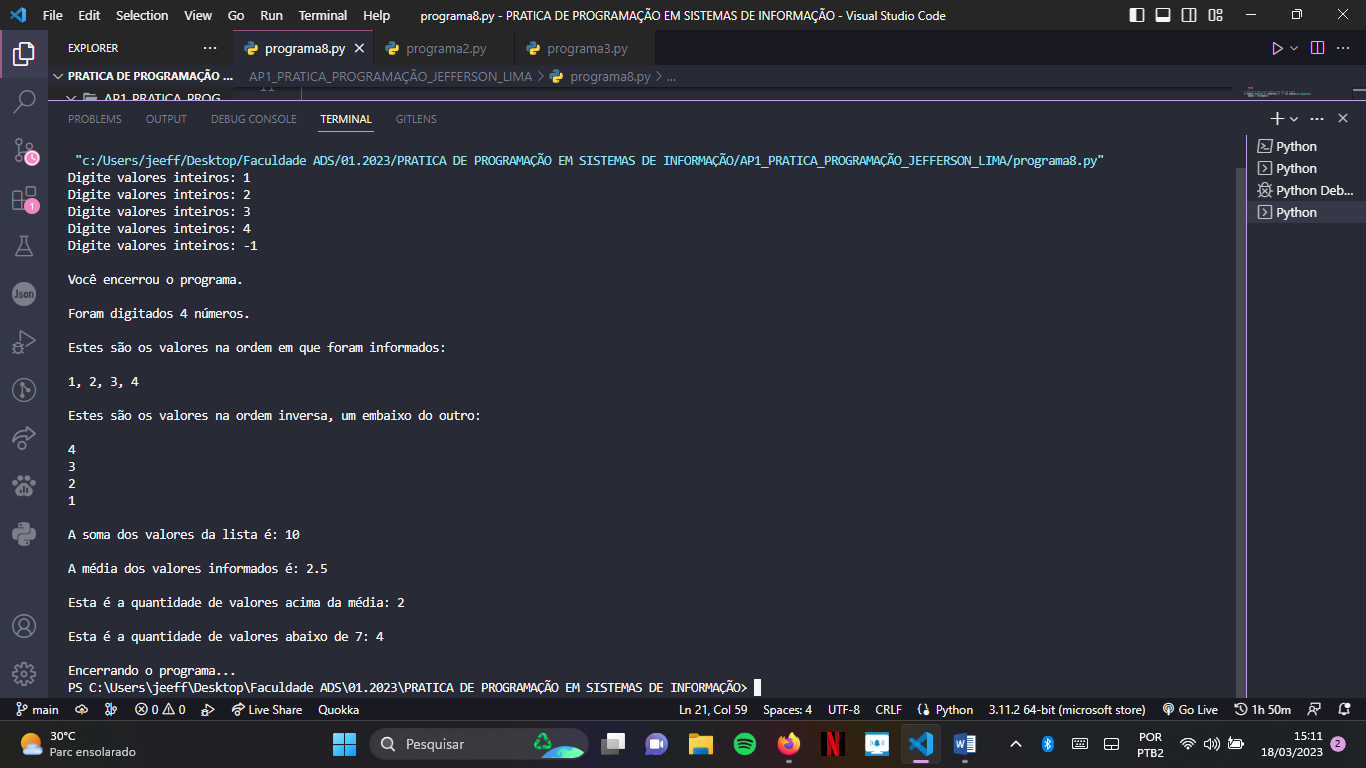
# Loop `for` para exibir a tabuada do número escolhido pelo usuário.

for multiplicador in range(1, 11):

    resultado = numero\_convertido \* multiplicador

    print(f'{numero\_convertido} X {multiplicador} = {resultado}')

1. **PROGRAMA 8**



Código do oitavo programa:

lista = []

numero = 0

while numero != -1:

    numero = input('Digite valores inteiros: ')

    # Verifica se o usuário digitou um valor vazio e encerra o programa

    if numero == '':

        print('Você digitou um valor vazio e encerrou o programa!')

        break

    # Converte o valor para inteiro caso seja um número

    if numero.isdigit() or (numero[0] == '-' and numero[1:].isdigit()):

        numero = int(numero)

    # Verifica se o usuário encerrou o programa

    if numero == -1:

        print('\nVocê encerrou o programa.\n')

        break

    # Adiciona o valor à lista caso seja um número inteiro

    elif type(numero) == int:

        lista.append(numero)

    # Caso o usuário tenha digitado um valor textual, encerra o programa

    else:

        print('\nVocê digitou um valor textual e encerrou o programa.\n')

        break

# Mostra a quantidade de valores lidos

print(f'Foram digitados {len(lista)} números.')

# Mostra os valores na ordem em que foram informados

print(f'\nEstes são os valores na ordem em que foram informados:\n')

print(', '.join(str(n) for n in lista))

# Mostra os valores na ordem inversa

lista.reverse()

print(f'\nEstes são os valores na ordem inversa, um embaixo do outro:\n')

for n in lista:

    print (n)

# Calcula e mostra a soma dos valores

lista\_somada = sum(lista)

print(f'\nA soma dos valores da lista é: {lista\_somada}')

# Calcula e mostra a média dos valores

media = sum(lista) / len(lista)

print(f'\nA média dos valores informados é: {media:.1f}')

# Calcula e mostra a quantidade de valores acima da média

valores\_acima\_media = []

for n in lista:

    if n > media:

        valores\_acima\_media.append(n)

print(f'\nEsta é a quantidade de valores acima da média: {len(valores\_acima\_media)}')

# Calcula e mostra a quantidade de valores abaixo de sete

abaixo\_sete = []

for n in lista:

    if n < 7:

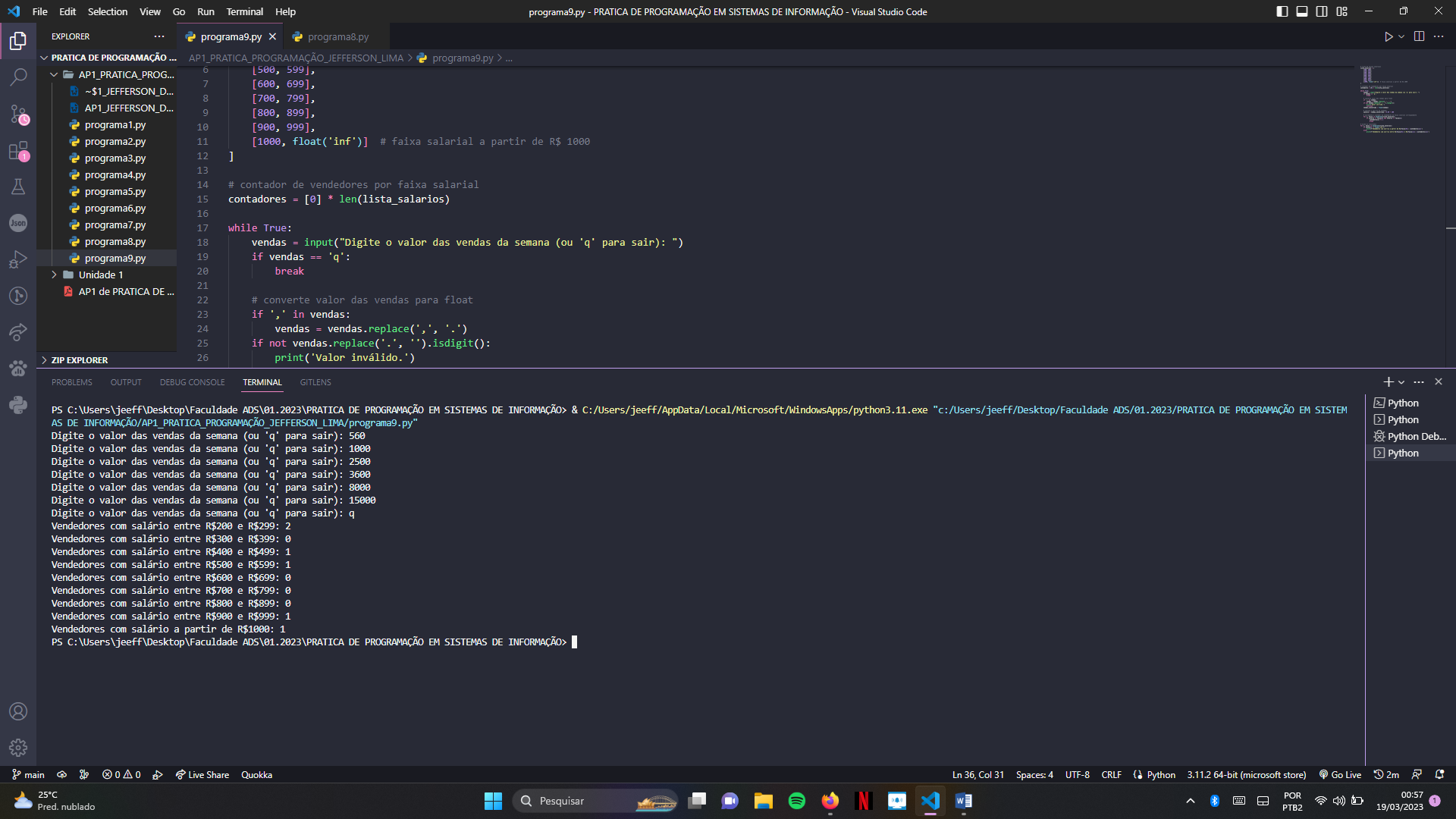
        abaixo\_sete.append(n)

print(f'\nEsta é a quantidade de valores abaixo de 7: {len(abaixo\_sete)}')

# Encerra o programa

print('\nEncerrando o programa...')

1. **PROGRAMA 9**



Código do nono programa:

# lista de faixas salariais

lista\_salarios = [

    [200, 299],

    [300, 399],

    [400, 499],

    [500, 599],

    [600, 699],

    [700, 799],

    [800, 899],

    [900, 999],

    [1000, float('inf')]  # faixa salarial a partir de R$ 1000

]

# contador de vendedores por faixa salarial

contadores = [0] \* len(lista\_salarios)

while True:

    vendas = input("Digite o valor das vendas da semana (ou 'q' para sair): ")

    if vendas == 'q':

        break

    # converte valor das vendas para float

    if ',' in vendas:

        vendas = vendas.replace(',', '.')

    if not vendas.replace('.', '').isdigit():

        print('Valor inválido.')

        continue

    vendas\_convertido = float(vendas)

    # calcula o salário do vendedor

    salario = vendas\_convertido \* 0.09 + 200

    # incrementa o contador de vendedores na faixa salarial correspondente

    for i, faixa in enumerate(lista\_salarios):

        if salario >= faixa[0] and salario <= faixa[1]:

            contadores[i] += 1

            break

# exibe os resultados

for i, faixa in enumerate(lista\_salarios):

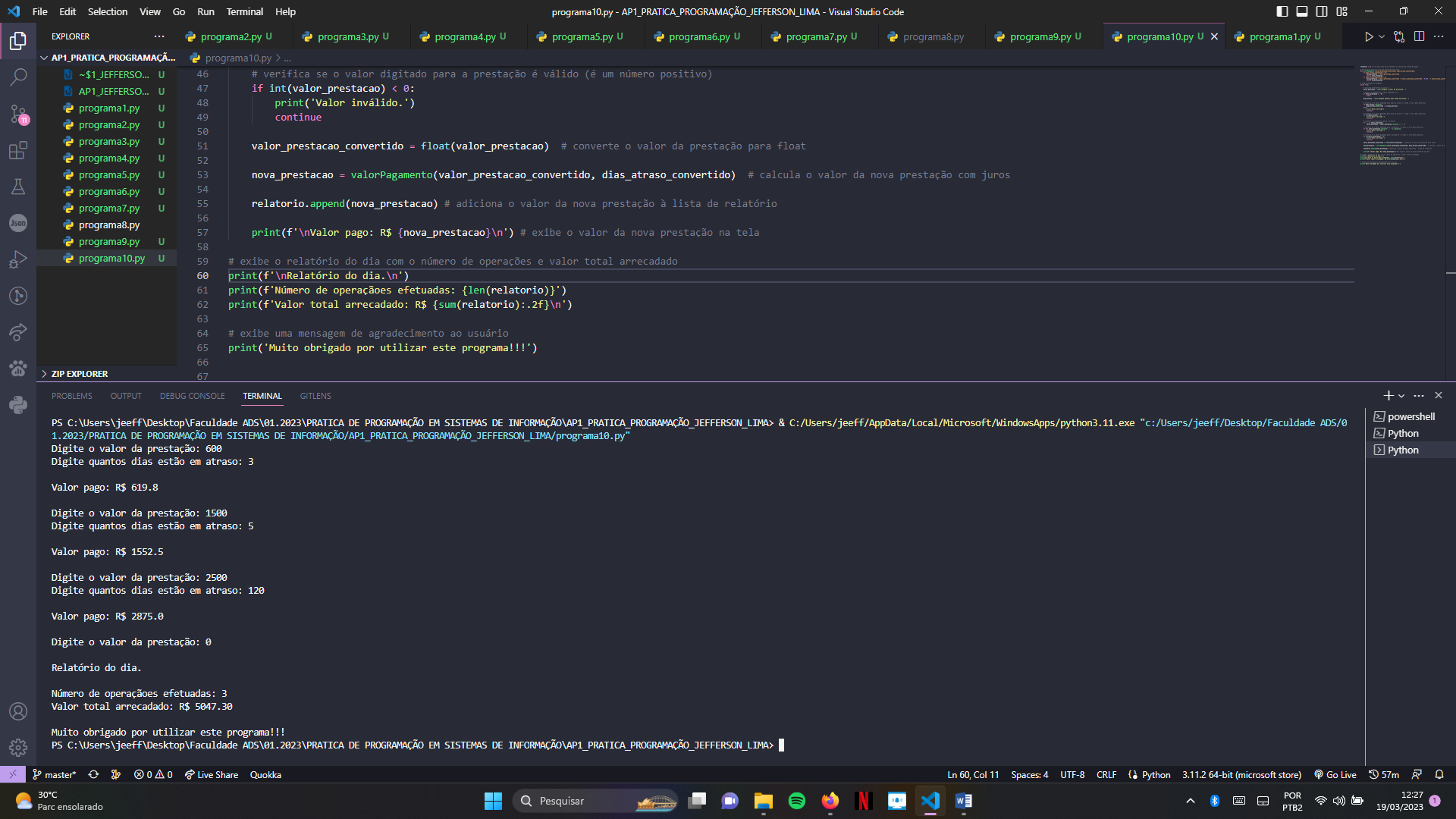
    if faixa[1] == float('inf'):

        print(f'Vendedores com salário a partir de R${faixa[0]}: {contadores[i]}')

    else:

        print(f'Vendedores com salário entre R${faixa[0]} e R${faixa[1]}: {contadores[i]}')

1. **PROGRAMA 10**



Código do décimo e último programa

relatorio = [] # cria uma lista para armazenar os valores de prestação pagos

# função que calcula o valor da prestação com juros

def valorPagamento (*valor\_prestacao\_convertido*, *dias\_atraso\_convertido*):

    if *dias\_atraso\_convertido* == 0:

        nova\_prestacao = *valor\_prestacao\_convertido*

        return nova\_prestacao

    if *dias\_atraso\_convertido* > 0:

        nova\_prestacao = *valor\_prestacao\_convertido* + (*valor\_prestacao\_convertido* \* 0.03) + ( *dias\_atraso\_convertido* \* (*valor\_prestacao\_convertido* \* 0.001))

        return nova\_prestacao

# loop principal do programa

while True:

    # solicita informações ao usuário

    valor\_prestacao = input('Digite o valor da prestação: ')

    # encerra o programa se o valor digitado for 0

    if valor\_prestacao == '0':

        break

    dias\_atraso = input('Digite quantos dias estão em atraso: ')

    # verifica se o valor digitado para dias em atraso é válido (é um número positivo)

    if dias\_atraso.isdigit():

        dias\_atraso\_convertido = int(dias\_atraso)

    else:

        print('Valor inválido')

        continue

    # verifica se o valor digitado para dias em atraso é válido (é um número positivo)

    if int(dias\_atraso) < 0:

        print('Valor inválido.')

        continue

    # converte a vírgula para ponto, se houver

    if ',' in valor\_prestacao:

        valor\_prestacao = valor\_prestacao.replace(',', '.')

    # verifica se o valor digitado para a prestação é válido (é um número positivo)

    if not valor\_prestacao.replace('.', '').isdigit():

        print('Valor inválido.')

        continue

    # verifica se o valor digitado para a prestação é válido (é um número positivo)

    if int(valor\_prestacao) < 0:

        print('Valor inválido.')

        continue

    valor\_prestacao\_convertido = float(valor\_prestacao)  # converte o valor da prestação para float

    nova\_prestacao = valorPagamento(valor\_prestacao\_convertido, dias\_atraso\_convertido)  # calcula o valor da nova prestação com juros

    relatorio.append(nova\_prestacao) # adiciona o valor da nova prestação à lista de relatório

    print(f'\nValor pago: R$ {nova\_prestacao}\n') # exibe o valor da nova prestação na tela

# exibe o relatório do dia com o número de operações e valor total arrecadado

print(f'\nRelatório do dia.\n')

print(f'Número de operaçãoes efetuadas: {len(relatorio)}')

print(f'Valor total arrecadado: R$ {sum(relatorio):.2f}\n')

# exibe uma mensagem de agradecimento ao usuário

print('Muito obrigado por utilizar este programa!!!')

1. **CONCLUSÃO**

O trabalho propõe exercícios simples, porém desafiadores para os programadores de primeira viagem que não possuem conhecimento de lógica de programação e algortimos. Neste trabalho consegui praticar a sintaxe do Python, me tornei capaz de diferenciar a tipagem de outras linguagens, pude constatar que Python é uma linguagem de tipagem dinamica e forte. Devo sempre me preocupar com o tipo primitivo que atribui a determinada variavel.

Pratiquei as estruturas de loop e estruturas condicionais da linguagem, tive um pouco de dificuldade para me acostumar com a indentação da linguagem. Estou acostumado com o uso de chaves e pontuações para encerrar funções ou processos. Acredito que compreendi o básico de Python e estou preparado para os próximos desafios que virão.

Além do arquivo zipado e este documento, estarei disponibilizando os códigos em um repositório no GitHub.