**Segunda lista exercícios**

**Davi Ramos Ferreira 1MA**

**1** a) porque o passo está negativo e deveria ser positivo

**Justificação:** O laço está configurado para incrementar a variável **i** de 100 até 200, mas o passo é negativo, o que é incorreto. Para que o laço funcione corretamente, o passo deve ser positivo.

for i in range(100, 200, -3):

    print(i)

2. b) porque o passo está positivo e deveria ser negativo

**Justificação:** O laço tenta incrementar **i** de 300 até 100 com um passo positivo de 3, o que não é possível, pois 300 é maior que 100. O passo deveria ser negativo para que **i** decrescesse.

for i in range(300, 100, 3):

    print(i)

3. a) No visualg, o comando PARA usa uma variável de controle do tipo caracter, inteiro ou literal.

**Justificação:** A variável de controle no comando PARA deve ser do tipo inteiro. Não pode ser do tipo caracter ou literal.

4. b) o algoritmo exibe o texto “i” 10 vezes

**Justificação:** O comando **escreval(“i”)** imprime a letra "i" (caractere) e não o valor da variável **i**. A instrução **Fimse** está incorretamente colocada e não faz parte de nenhum bloco SE, mas isso não afeta a execução da impressão do caractere.

for i in range(1, 11):

    print("i")

7. a) acumulador

**Justificação:** Este comando soma o valor atual de **x** com **valor** e armazena o resultado em **x**, caracterizando um acumulador.

8. b) média

**Justificação:** Este comando divide o **total** pela **qtd**, calculando a média.

9. a) o programa emite a média da soma dos números entre 1 e 300

**Justificação:** O programa acumula a soma de **i** de 1 até 300 em **x** e depois imprime a média dividindo o total por 300.

10. b) o programa inicia uma contagem e termina rapidamente, pois o valor inicial é maior que o final - na hora em que for executado não vai mostrar nada

**Justificação:** O loop **Para i:=1000 ate 1 faça** não executa porque o valor inicial é maior que o final e o passo não foi especificado como negativo. Portanto, o programa não imprime nada.

11. b) o algoritmo lista os nomes, mas vai mostrar apenas o último nome cadastrado 5 vezes

**Justificação:** A variável **nome** armazena apenas o último valor lido e este valor será repetido 5 vezes na segunda parte do algoritmo.

12. **Explicação:** Este algoritmo preenche o vetor **Numero** com o dobro do valor do índice **CONT**.

Numero = [0] \* 10

for CONT in range(1, 11):

    Numero[CONT - 1] = 2 \* CONT

for CONT in range(1, 11):

    print(f"Numero[{CONT}] := {Numero[CONT - 1]}")

13 14 e 15.

Numero = [0] \* 10

for CONT in range(1, 11):

    Numero[CONT - 1] = 2 \* CONT

for CONT in range(1, 11):

    print(f"Numero[{CONT}] := {Numero[CONT - 1]}")

# Programa completo que inclui as partes 13, 14 e 15

# Parte 13: Armazenar dados dos seriados

NomeSeriado = [""] \* 3

QuantidadeEpisodios = [0] \* 3

TotalEpisodios = 0

MaxEpisodios = 0

MaxIndex = 0

# Recebendo os dados dos seriados

for i in range(3):

    NomeSeriado[i] = input("Informe o nome do seriado: ")

    QuantidadeEpisodios[i] = *int*(input("Informe a quantidade de episodios: "))

    TotalEpisodios += QuantidadeEpisodios[i]

    if QuantidadeEpisodios[i] > MaxEpisodios:

        MaxEpisodios = QuantidadeEpisodios[i]

        MaxIndex = i

# Calculando a média de episódios

MediaEpisodios = TotalEpisodios / 3

# Exibindo os resultados

print(f"Total de episodios: {TotalEpisodios}")

print(f"Media de episodios por serie: {MediaEpisodios:.2f}")

print(f"Seriado com mais episodios: {NomeSeriado[MaxIndex]}")

# Parte 14: Listar os vetores de trás para frente

print("\nListando os seriados de trás para frente:")

for i in range(2, -1, -1):

    print(f"Seriado: {NomeSeriado[i]}, Episodios: {QuantidadeEpisodios[i]}")

# Parte 15: Geração de números pares de 0 a 500

print("\nNúmeros pares de 0 a 500:")

for i in range(0, 501, 2):

    print(i)