

- 3) Analise as afirmações a seguir sobre a estrutura de repetição while da linguagem Python e indique a alternativa correta:
- (I) Para que o bloco de instruções do laço seja executado, sua expressão de decisão, também conhecida como condição, deve resultar em True quando avaliada.
- (II) Uma forma de garantir que o laço seja encerrado é criar uma expressão de decisão que, em algum momento quando for avaliada, possa resultar em False.
- (III) Essa estrutura de repetição permite que uma sequência de instruções seja executada uma quantidade indeterminada de vezes, o que também pode criar loops infinitos.

- A) Todas as afirmações são verdadeiras.
B) Nenhuma das afirmações é verdadeira.
C) Apenas as afirmações I e II são verdadeiras.
D) Apenas as afirmações II e III são verdadeiras.
E) Apenas as afirmações I e III são verdadeiras.

- 2) Considerando o programa da figura a seguir, assinale a alternativa correta supondo que o usuário dê como entradas os valores 100 e 200, respectivamente.

```
a = int(input())
b = int(input())
c = 0
while c < a:
    print('bom dia!')
    d = 0
    while d < b:
        print('boa tarde!')
        d += 1
    c += 1
print('boa noite!')
```

- A) A string 'bom dia!' será exibida 100 vezes; a string 'boa tarde!' será exibida 20000 vezes; a string 'boa noite!' será exibida 1 vez.
B) A string 'bom dia!' será exibida 100 vezes; a string 'boa tarde!' será exibida 100 vezes; a string 'boa noite!' será exibida 1 vez.
C) A string 'bom dia!' será exibida 100 vezes; a string 'boa tarde!' será exibida 200 vezes; a string 'boa noite!' será exibida 100 vezes.
D) A string 'bom dia!' será exibida 100 vezes; a string 'boa tarde!' será exibida 200 vezes; a string 'boa noite!' será exibida 1 vez.
E) A string 'bom dia!' será exibida 100 vezes; a string 'boa tarde!' será exibida 20000 vezes; a string 'boa noite!' será exibida 100 vezes.

- 3) Vimos no material didático que a passagem de argumentos em Python é dita “passagem por objeto” e que isso funciona como uma mistura entre os conceitos de “passagem por referência” e “passagem por valor”. Marque a alternativa correta em relação a como é feita a passagem de argumentos em Python.

- A) Podemos escolher de qual forma iremos passar os argumentos, por padrão é feita a passagem por valor, e quando queremos fazer a passagem por referência, precedemos a variável por um cifrão.
B) Podemos escolher de qual forma iremos passar os argumentos, por padrão é feita a passagem por valor, e quando queremos fazer a passagem por referência, precedemos a variável por um asterisco.
C) O Python verifica o tipo da variável e então escolhe se irá fazer uma cópia do valor ou passar a referência para a função.
D) A passagem de argumentos é feita sempre por referência, mas observamos o efeito colateral, típico dessa forma de passagem de argumentos, apenas em objetos que sejam mutáveis. Os objetos imutáveis se comportam como se a passagem tivesse sido por valor.
E) A passagem de argumentos é feita sempre por valor, e uma cópia dos dados é feita sempre.

- 4) O código a seguir solicita ao usuário que seja digitado números inteiros e os armazene em uma lista de 3 posições. Substituindo todos os valores positivos e iguais a zero por 1 e todos os valores negativos por 0.

```
def troca(lista):
    for i in range(3):
        if lista[i]>= 0:
            lista[i] = 1
        else:
            lista[i] = 0
    return lista

lista = [0]*3
for i in range(3):
    lista[i] = int(input('Digite um valor: '))
print(lista)

print(lista)
```

Selecione a opção que completa o código corretamente.

- A) troca(lista)
B) troca(range)
C) troca(lista[])
D) lista(i)
E) lista(troca)

- 5) Marque a alternativa que contém apenas operações aplicáveis a todos os tipos de sequências estudadas durante o curso (listas, tuplas, strings e intervalos), também chamadas de “Operações comuns de sequências”.

- A) Pertencimento; concatenação e repetição; inclusão de itens na sequência; contagem de ocorrências.
B) Pertencimento; busca por valor; indexação e fatiamento; contagem de ocorrências.
C) Pertencimento; substituição de itens da sequência; indexação e fatiamento; contagem de ocorrências.
D) Pertencimento; concatenação e repetição; indexação e fatiamento; remoção de itens da sequência.
E) Pertencimento; substituição de itens da sequência; inclusão de itens na sequência; contagem de ocorrências.

- 6) Como visto na disciplina, existem pelo menos dois instrumentos necessários para iniciar a resolução de problemas visando a implementação automatizada da solução em um computador:

- (I) Um deles tem o papel de representar os aspectos importantes do problema alvo, descartando o que é irrelevante e objetivando criar um cenário que possa ser entendido pelo programador para ajudar na compreensão do problema e de como resolvê-lo, facilitando, por exemplo, a identificação do estado inicial do problema e qual o estado final desejável;

- (II) O outro é uma sequência de instruções logicamente ordenada que, se formulada e executada corretamente, resolve o problema, transformando o estado inicial em um estado final desejável.

Com base no exposto, indique a alternativa que contém os termos mais adequados aos quais (I) e (II) se referem, respectivamente:

- A) A linguagem de programação e o computador.
B) O computador simplificado e o processamento.
C) O algoritmo e a representação.
D) A entrada e a saída.
E) A representação e o algoritmo.

- 7) Avalie o código da imagem a seguir e marque a alternativa que apresenta as saídas corretas, geradas pelo programa caso o usuário execute-o 3 vezes, com as respectivas entradas:

1)

vertebrado

mamífero

2)

invertebrado

anelídeo

3)

invertebrado

aracnídeo

```
entrada_1 = input()
entrada_2 = input()

if entrada_1 == "vertebrado":
    if entrada_2 == "ave":
        print("é uma ave")
    elif entrada_2 == "mamífero":
        print("é um mamífero")
    else:
        print("vertebrado desconhecido")
elif entrada_1 == "invertebrado":
    if entrada_2 == "inseto":
        print("é um inseto")
    elif entrada_2 == "anelídeo":
        print("é um anelídeo")
    else:
        print("invertebrado desconhecido")
else:
    print("animal desconhecido")
```

- A) é um mamíferoé um anelídeoinvertebrado desconhecido
B) é uma aveé um anelídeoinvertebrado desconhecido
C) é uma aveé um insetoé um aracnídeo
D) é um mamíferoé um anelídeoé um aracnídeo
E) é um mamíferoé um insetoé um anelídeo

- 8) Considerando um programa que lê dois valores inteiros e exibe o resultado do primeiro número dividido pelo segundo número, escolha a alternativa que o complete:

```
n1 = _____
n2 = int(input("Informe outro número: "))

print("0 resultado da divisão é: {:.2f}".format(n1/n2))
```

- A) int("Informe um número: ")
B) float(int("Informe um número: "))
C) input("Informe um número: ")
D) print(input("Informe um número: "))
E) int(input("Informe um número: "))

- 9) O programa da figura a seguir recebe como entrada um número natural n e deve exibir a sequência de valores naturais de zero até n-1 e de n-1 até zero. Por exemplo, se n=3 o programa deverá exibir 0 1 2 e, na próxima linha, 2 1 0. Assinale a alternativa que contém os valores que substituem adequadamente os símbolos ? (interrogação), e na ordem correta, de modo que o programa funcione conforme a especificação. Observação: Desconsidere as |' das alternativas, são apenas separadores da sequência de valores.

```
n = int(input('número: '))

inicio = ?
fim      = ?
passo    = ?

for i in range(inicio, fim):
    print(i, end=' ')

print()

for i in range(?, ?, passo):
    print(i, end=' ')

A) 0 | n      | -1 | fim - 1 | início - 1
B) 0 | n      | -1 | fim + 1 | início + 1
C) 0 | n + 1 | -1 | fim - 1 | início + 1
D) 0 | n      | -1 | fim      | início + 1
E) 0 | n      | 1  | fim + 1 | início - 1
```

- 10) Python tem regras de precedência e associatividade para definir como os operandos são relacionados aos operadores em uma expressão, evidentemente isso influencia no resultado. No início da aprendizagem, é comum inserirmos pares de parênteses desnecessários nas expressões, o que eventualmente compromete a legibilidade do código. Por isso, em geral, recomenda-se utilizar apenas os parênteses essenciais. São essenciais os pares de parênteses que, se retirados, podem alterar o resultado da expressão. Por exemplo, em 5+(3*2), podemos retirar os parênteses sem alterar o resultado, mas na expressão 4*(3+5) os parênteses influenciam o resultado da expressão. Entretanto, nas expressões 2+3-4 e 2+(3-4) não há diferença no resultado, mas há alteração na associação dos operandos com os operadores, pois 3 foi associado à subtração por causa dos parênteses.

Você, que estudou o material didático, já conhece bem as regras e sabe como deixar as expressões mais legíveis apenas com os parênteses essenciais e que mantêm a mesma associação entre operandos e operadores. Sabendo disso, veja a expressão a seguir:

((11%((5+4)**2))-(12/(2*3)))

Indique a alternativa que contém uma expressão semelhante à da figura, com o *mesmo resultado* e *mesma associação entre operandos e operadores*, além de conter apenas os parênteses essenciais.

- A) (11%(5+4)**2)-(12/(2*3))
B) 11%(5+4)**2-12/2*3
C) 11%(5+4)**2-(12/(2*3))
D) 11%(5+4**2-12/2*3
E) 11%(5+4)**2-12/(2*3)