

# Teste de desenvolvimento de sistemas

Rodrigo Schmidt Lucchesi

October 2019

1. Nenhum corpo que não seja sólido é cristal. Conclui-se logicamente que:

r.: Todo cristal é corpo sólido

---

2. A soma de dois números é 16 e a diferença dos seus quadrados é 128. O produto desses dois números é:

$$x_1 + x_2 = 16 \quad (1)$$

$$x_1^2 - x_2^2 = 128 \quad (2)$$

A partir da equação (1):

$$x_1 = 16 - x_2 \quad (3)$$

Substituindo (3) em (2):

$$(16 - x_2)^2 - x_2^2 = 128$$

$$(256 - 32x_2 + x_2^2) - x_2^2 = 128$$

$$256 - 32x_2 = 128$$

$$-32x_2 = 128 - 256$$

$$-32x_2 = -128$$

$$x_2 = 4 \quad (4)$$

Substituindo (4) em (3):

$$x_1 = 16 - 4$$

$$x_1 = 12 \quad (5)$$

Sendo assim:

$$x_1 \cdot x_2 = 48 \rightarrow \text{Resposta}$$

---

3. Em determinada rua, há 5 residências que estão numeradas em forma de progressão geométrica crescente, de modo que a primeira residência recebeu numeração 3, e a segunda, numeração 12. Com base nessas informações, assinalar a alternativa CORRETA:

$$a(n) = a_1 \cdot q^{(n-1)}$$

$$a_1 = 3$$

$$a_2 = 12$$

$$a_2 = a_1 \cdot q$$

$$12 = 3 \cdot q$$

$$q = 4$$

Sendo assim:

$$a_3 = 3 \cdot 4^2 = 48$$

$$a_4 = 3 \cdot 4^3 = 192$$

$$a_5 = 3 \cdot 4^4 = 768$$

Correta:

(X) O resultado da soma da numeração da terceira e da quarta casa é igual a 240.

$$a_3 + a_4 = 48 + 192 = 240$$

---

5. Um método se torna multithread quando é declarado como Async e esperado com await?

(X) Não, Async/Await não é uma garantia que o método irá ser executado em uma nova thread.

---

6. Usar programação assíncrona (Async/Await) garante que a execução do código seja mais rápida:

(X) A programação assíncrona executa tarefas em paralelo, o código pode terminar a execução mais rápido mesmo sem mudança de performance.

---

7. Qual o resultado do programa a seguir? (Somente uma resposta correta)

```
internal class Program
{
    private static void Main(string[] args)
    {
        int? n1 = null;
        int? n2 = 45;
        int? n3 = null;
        Console.WriteLine(n1 > n2);
        Console.WriteLine(n1 < n2);
        Console.WriteLine(n1 != n2);
        Console.WriteLine(n1 == n2);
        Console.WriteLine(n1 == n3);
    }
}
```

Resposta:

Código em C#:

Output = [False, False, True, False, True]

- 
8. Escrever um código utilizando português estruturado ou Pseudocódigo, no qual o resultado final do Vetor deve conter os mesmos números ordenados crescentemente. Considerando o seguinte vetor = {5,3,2,4,7,1,0,6} (Não utilize funções de ordenação prontas)

Output = [0,1,2,3,4,5,6,7]

Input = [5,3,2,4,7,1,0,6]

Sorting Algorithm [Insertion]

8.1. Examina o valor atual com o da esquerda do vetor, ou seja, iterador sentido crescente positivo;

8.2. Troca o valor analisado se este é menor do que o anterior;

8.3. Pior cenário:

Se os valores estiverem em ordem decrescente:

$t = O(n^2)$  onde  $n$  = tamanho do vetor

#### 8.4. Pseudocódigo

```
Para cada item, iniciando do primeiro valor
até o último valor do vetor - 1 unidade:
    contador = 1 // (j)
    Enquanto contador for maior do que 0
    e o vetor (A) não tenha chegado a ultima posição:

        // Troque a posição do vetor do valor do contador
        A[j] pelo o da esquerda A[j-1]
        Troque A[j] por A[j-1]
        contador = contador -1
        j = j -1
```