

AVALIAÇÃO 01 - (LINGUAGEM DE PROGRAMAÇÃO 2)

NOME: _____ NÚMERO: _____ TURMA: _____ DATA: ____/____/____

QUESTÃO 01 (0,1 PT)

Um paradigma de programação é uma forma pela qual linguagens de programação são concebidas além de definir como o programador deve organizar o seu código em termos da modularização de seus componentes.

Marque a opção que apresenta os paradigmas de programação seguidos pela linguagem C e C#, respectivamente.

- a) Código de Máquina e byte code.
- b) Lógico e Funcional.
- c) Estruturado e Orientação a Objetos.
- d) Funções e Métodos.
- e) Orientação a Estruturas e Orientação a Classes.

QUESTÃO 02 (0,1 PT)

A programação orientada a objetos trouxe como uma de suas principais novidades, a capacidade de aproximar o mundo real do mundo prático da programação. Isso é feito essencialmente através do conceito de classes. Tratando-se de classes, é correto afirmar que:

- a) São exatamente como *structs*, conceito esse presente em linguagens como C.
- b) Para que um objeto seja criado, é necessário fazer uso da palavra reservada *new* e chamar um método construtor daquela classe.
- c) Sempre apresentam em seu escopo atributos e métodos.

d) Não podem conter variáveis locais em nenhum de seus métodos.

e) Se seus atributos forem estáticos, então esses atributos são chamados de variáveis de instância.

QUESTÃO 03 (0,1 PT)

Sobre a linguagem C#, é correto afirmar que:

- a) por padrão, uma aplicação de Console C# cria uma classe Main contendo um método Program.
- b) a extensão padrão de seus arquivos costuma ser .csharp.
- c) os comandos de entrada e saída no console podem ser encontrados entre os métodos da classe Console.
- d) o valor 3.1415 é considerado do tipo float enquanto que o valor 18 do tipo double.
- e) a classe Math contém um conjunto de métodos para converter valores. Por exemplo, podemos citar os métodos *ToIn32*, *ToChar*, *ToString* e *ToDouble*.

QUESTÃO 04 (0,1 PT)

Em C#, o processo de instanciação, ou seja, de criação de novos objetos, é realizado através do comando cuja sintaxe adequada é:

- a) `nome_Classe = new nome_Objeto();`
- b) `nome_Classe nome_Objeto = new nome_Classe();`
- c) `nome_Instancia nome_Classe = new nome_Objeto();`

- d) nome_Objeto nome_Instancia = new
nome_Classe ();
- e) nome_Classe nome_Objeto = new
nome_Instancia();

QUESTÃO 05 (0,1 PT)

Sobre os métodos construtores, é correto afirmar que:

- f) Só pode existir um método construtor na classe.
- g) Não podem receber mais que dois parâmetros.
- h) Asseguram o encapsulamento.
- i) Não apresentam tipo de retorno.
- j) São sempre públicos.

QUESTÃO 06 (0,1 PT)

Em um sistema de livraria, foi necessário criar uma classe Livro. Nesta classe, os atributos Preço e Quantidade não podem assumir valores negativos, uma vez que vai contra a regra de negócio do contexto. Acerca do código C# abaixo, analise as seguintes alternativas e marque a correta

```
class Livro
{
    private double Preço;
    private int Quantidade;
    public string Nome;

    public Livro(double p, int q, string n)
    { Preço = p; Quantidade = q; Nome = n; }
}
```

- a) A classe implementou adequadamente o conceito de encapsulamento.
- b) Não é possível criar objetos inconsistentes.
- c) Como o atributo Nome não é privado, o encapsulamento não está sendo aplicado na íntegra.
- d) O construtor é utilizado para criar objetos íntegros.
- e) A visibilidade privada dos atributos não garante por si só o encapsulamento.

QUESTÃO 07 (0,1 PT)

Observe os códigos que são mostrados abaixo e marque a opção que indica o que será apresentado no Console.

```
class Teste
{
    public static int A = 1;
    public int B = 0;

    public Teste(int x)
    {
        B = x + A;
        A++;
    }
}
```

```
class Program
{
    static void Main(string[] args)
    {
        Teste t1 = new Teste(2);
        Teste t2 = new Teste(4);
        Teste t3 = new Teste(8);

        Console.WriteLine(t1.B);
        Console.WriteLine(t2.B);
        Console.WriteLine(t3.B);
        Console.WriteLine(Teste.A);

        Console.ReadLine();
    }
}
```

- a) 3 5 9 | b) 2 4 8 | c) 3 5 9 4 d) 3 6 | | 4 e) 3 6 | | |

QUESTÃO 08 (0,1 PT)

Ainda nas duas classes da questão anterior, quantas variáveis locais foram declaradas no total?

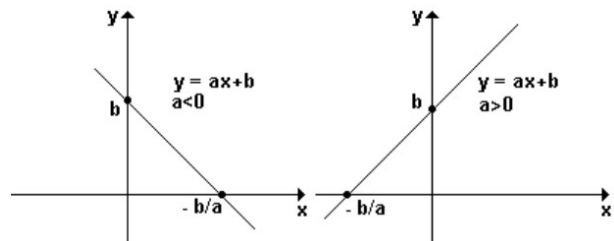
- a) 5 b) 4 c) 3 d) 2 e) 1

QUESTÃO 09 – BASE (LEIA COM ATENÇÃO)

Uma Função Polinomial do 1º Grau ou Função Linear é a função real definida por:

$$f(x) = ax + b$$

onde a e b são coeficientes reais. O gráfico obtido a partir de uma Função Linear no plano cartesiano é uma reta como mostra a figura ao lado. Se $a = 0$, então a função linear é dita constante e representa uma reta paralela ao eixo x .



Várias são as características que podem ser extraídas de Funções Lineares. Vejamos:

I. Inclinação da reta

Uma função linear é dita crescente se o valor de $a > 0$ e decrescente caso $a < 0$.

II. Raiz

A reta intercepta o eixo x (eixo das abscissas) no ponto $(x, 0)$, ou seja, sempre que $f(x) = 0$. O valor de x que anula a função é chamado de Raiz. Uma Função Linear tem uma única raiz que pode ser calculada como

$$x_{raiz} = \frac{-b}{a}$$

QUESTÃO 09 – CLASSE FUNÇÃO LINEAR (0,6 PT)

De acordo com o que foi dito, implemente uma classe que representa uma Função Linear que não seja constante seguindo as instruções abaixo:

- Implemente uma classe `FuncaoLinear`. Como sabemos que uma Função Linear é definida pelos seus dois coeficientes, adicione nesta classe dois atributos reais denominados por A e B .
- Determine se há necessidade de aplicar o encapsulamento em algum ou todos os atributos da classe de acordo com a descrição dada no exercício. Se houver a necessidade, crie os métodos Getters e Setters para o(s) atributo(s) e justifique com um comentário no código.
- Implemente um construtor para esta classe que inicializa os seus atributos mantendo a integridade dos mesmos.
- Implemente o método `Raiz` que retorna o valor da raiz de um objeto `FuncaoLinear`.
- Implemente o método `Inclinação` que retorna um texto indicando “crescente” ou “decrescente” de acordo com o objeto em questão.
- Implemente o método `Descrever` que retorna o formato da função linear de acordo com seus atributos.
Exemplo: caso $A=2$ e $B=1$, esse método deverá retornar o texto “ $f(x)=2x+1$ ”.

QUESTÃO 10 – CLASSE PROGRAMA (0,6PT)

Sabemos que os Projetos ou Soluções em C# são compostos por diversas classes. Dentre elas, a classe Program.cs se mostra importante por conter o método Main(), ponto de partida na execução de qualquer programa.

Sabendo disso, e considerando que no seu projeto as classes Program.cs e FuncaoLinear.cs (como implementado na questão anterior) estão presentes, implemente o método Main() da classe Program.cs.

Esse método deverá perguntar através do Console a um usuário, quais são os parâmetros A e B da função linear que ele deseja estudar. Após a leitura dos dados, instancie um objeto da classe FuncaoLinear e a utilize na resolução da sua implementação. Observe um exemplo de entrada e saída logo abaixo.

```
Estudando uma Função Linear!
Favor, digite os valores de A e B:
2
1
Sua função:  $f(x)=2x+1$ 
Inclinação da reta: crescente
Raiz: -0.5
```