

## LINGUAGEM DE PROGRAMAÇÃO 2

NOME: \_\_\_\_\_ TURMA: \_\_\_\_\_

### EXERCÍCIO 01 – ANÁLISE DE ERROS

O Código a seguir apresenta 4 erros de compilação (aquele tipo de erro que não deixa o programa ser executado sem antes ter o código corrigido). É pedido que:

```
double bas;  
double alt;  
  
Console.WriteLine("Entre com a base do triângulo:")  
bas = Console.ReadLine();  
  
Console.WriteLine("Entre com a altura do triângulo:");  
alt = Convert.ToInt(Console.ReadLine());  
  
Console.WriteLine('O valor da área do triângulo vale:');  
Console.WriteLine(bas * alt / 2);  
  
Console.ReadLine();
```

- Sublinhe as 4 linhas de código que apresentam os erros de compilação.
- Indique/descreva o motivo de cada um desses erros.

---

---

---

---

**Para validar a sua resposta, escreva o programa no seu ambiente de desenvolvimento e observe os erros de compilação.**

- Corrija o código para que ele possa ser compilado e executado adequadamente.

## EXERCÍCIO 02

Dentre os tipos de dados usados pela linguagem C# listados abaixo, marque aquele que ocupa a menor quantidade de bits na memória RAM:

- a) int
- b) float.
- c) char
- d) bool.
- e) double

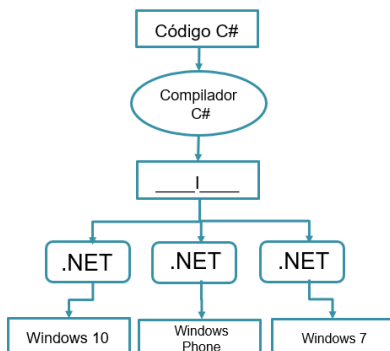
## EXERCÍCIO 03

Um ambiente de desenvolvimento integrado (IDE - *Integrated Development Environment*) é uma espécie de programa onde desenvolvedores podem implementar e testar programas. De acordo com o que a Microsoft preconiza, qual a principal IDE para se desenvolver software C#?

- a) Dev C++
- b) Visual Studio
- c) Eclipse
- d) Spyder
- e) Netbeans

## EXERCÍCIO 04

.NET é a plataforma que fornece o ambiente de execução, bibliotecas e ferramentas necessárias para desenvolver e executar aplicações em C#. Com base na imagem abaixo, responda:



O produto resultante do processo de compilação de código C#, que deve preencher a lacuna I, é conhecido como

- a) Hashcode.
- b) Fonte.
- c) CIL (*Common Intermediate Language*).
- d) Código Binário.
- e) Código de Máquina.

## EXERCÍCIO 05

Assinale a alternativa que associa os tipos de dados

string, double, char, int, bool

com os respectivos valores interpretados em tempo de execução:

- a) 'viviane', 5.19, 'v', 4, true
- b) bianca, 3.0, "b", 0, false
- c) "joao", 6, 'j', 7.0, 0
- d) "flavio", 7.8, 'f', 4, 1
- e) "claudio", 4.1, 'c', -3, false

## EXERCÍCIO 06

Observe o código C# abaixo:

```
double a = 4;
double b = 3;
double c = 2;

a = ((b * b) - (c + a));

b = c + 2;

c *= -1;

Console.ReadLine();
```

Imediatamente antes do primeiro comando de entrada (Console.ReadLine()), quais os valores que estão armazenados nas variáveis a, b e c, respectivamente?

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

---

## EXERCÍCIO 07 – IMPLEMENTAÇÃO DE CÓDIGO

---

Escreva um programa em C# que leia dois números reais (double) e imprima no console a soma, a subtração, a multiplicação, a divisão e o resto da divisão desses números. Observe um exemplo de entrada e saída:

```
Entrada e Saída:  
--- Calculadora ---  
Entre com dois valres:  
10  
2  
Soma: 12  
Subtração: 8  
Multiplicação: 20  
Divisão: 5  
Resto da Divisão: 0
```

---

## EXERCÍCIO 08 – IMPLEMENTAÇÃO DE CÓDIGO

---

Dentro do contexto da Física Mecânica, temos definições como velocidade, aceleração, deslocamento, tempo decorrido, entre outras. A expressão denominada **equação horária da velocidade** é dada por

$$v = v_0 + a \cdot t$$

e descreve como que a velocidade de um objeto em movimento uniformemente variado (ou seja, que apresenta uma aceleração não nula) se modifica em função do tempo. Nesta,  $v$  representa a velocidade final,  $v_0$  a velocidade inicial,  $a$  a aceleração e  $t$  o tempo decorrido.

Sabendo disso, implemente um programa em C# que dada a velocidade inicial de um objeto em movimento, sua aceleração e o tempo decorrido, imprima na tela qual a velocidade final. Observe um exemplo de entrada e saída:

```
Entrada:  
--- Calculadora de velocidade (m/s) ---  
Entre com a velocidade inicial (m/s):  
10  
Entre com a aceleração (m/s²):  
2  
Entre com o tempo decorrido(s):  
5
```

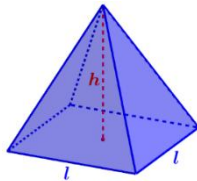
```
Saída:  
A velocidade final (m/s) vale:  
20
```

---

## EXERCÍCIO 09 – IMPLEMENTAÇÃO DE CÓDIGO

---

A Geometria é considerada uma das áreas mais importantes da Matemática. Conceitos como área e volume são vastamente explorados de acordo com diferentes formas geométricas. O volume da pirâmide depende diretamente da área da sua base e da sua altura. Considerando uma **pirâmide de base quadrada** escreva um programa que leia o lado e a altura da pirâmide e imprima na tela o seu volume. O seu volume é calculado através da seguinte relação:



$$\text{volume} = (\text{área da base}) * (\text{altura})/3$$

**Entrada:**

--- Calculadora de Volume (Pirâmide de Base Quadrada) ---  
Entre com o lado da base (m):  
3  
Entre com a altura (m):  
5

**Saída:**

O volume da pirâmide vale 15 m<sup>3</sup>.

---

## EXERCÍCIO 10 – IMPLEMENTAÇÃO DE CÓDIGO

---

Na Matemática Financeira, os **juros simples** são calculados utilizando a fórmula:

$$J = C . i . t$$

Onde:

- $J$  é o valor dos juros
- $C$  é o capital inicial
- $i$  é a taxa de juros (em %)
- $t$  é o tempo de aplicação (em meses ou anos, dependendo da taxa)



Sabendo disso, implemente um programa em C# que leia o capital inicial, a taxa de juros e o tempo de aplicação, e calcule o valor dos juros simples.

**Entrada:**

--- Calculadora de Juros Simples ---  
Entre com o capital inicial (R\$): 1000  
Entre com a taxa de juros mensal (em %): 2  
Entre com o tempo de aplicação (meses): 6

**Saída:**

O valor dos juros simples (R\$) é:  
120

Obs.: lembre-se de converter a taxa percentual para decimal antes de usar na fórmula. No exemplo acima, a taxa de 2% ao mês significa, portanto,  $2/100 = 0,02$ .