

Curso: Computação

<b>Disciplina:</b>	Programação orientada por objetos			<b>Sem.</b>	<b>Ano</b>	<b>Turno</b>
<b>Aluno:</b>		<b>RA:</b>		1º	2025	
<b>Professor:</b>	Charlene e Jardel				<b>NOTA</b>	
<b>Data:</b>	___/___/___		<b>Valor:</b>			

QUESTÃO	Nº1	VALOR	1	NOTA	
---------	-----	-------	---	------	--

1 - Crie um programa que contenha o método principal *main* e uma classe chamada *Pessoa*. A classe *Pessoa* deve conter os atributos *nome* do tipo *string* com visibilidade pública e o atributo *matricula* do tipo *inteiro* também com visibilidade pública.

1.2 - Instancie um objeto denominado *aluno*, a partir da classe *Pessoa*.

1.3 - Instancie um objeto denominado *professor*, a partir da classe *Pessoa*.

1.4 - Exiba o nome do objeto *aluno*.

1.5 - Altere o nome para o seu nome.

1.6 - Exiba novamente.

2 - Altere a visibilidade dos atributos e tente acessá-los de dentro da *main*.

2.1 - O que é necessário alterar para acessá-los de dentro da *main*?

2.3 - Crie os métodos *get* e o *set* para cada variável.

3 - Vamos supor que precisamos saber quantos objetos do tipo *Pessoa* foram instanciados. Como podemos fazer isso?

4 - Crie uma variável *static* dentro da classe *Pessoa* que é incrementada toda vez que instanciamos um objeto do tipo *Pessoa*. Para isso crie um método chamado *incluirPessoa()* que incremente o número de objetos do tipo *Pessoa* instanciado.

5 - Toda a vez que instanciamos um objeto do tipo *Pessoa*, temos que chamar a método para incrementar o número de objetos do tipo *Pessoa*. Como podemos resolver esse problema?

6 - Para que não seja necessário a criação de um método que incremente a quantidade de objetos do tipo *Pessoa* (*incluirPessoa()*), crie um construtor para a Classe *pessoa* que realize esse incremento de forma automática.

QUESTÃO	Nº2	VALOR	1	NOTA	
---------	-----	-------	---	------	--

Complete com V (verdadeiro) ou F (Falso):

Curso: Computação

Disciplina:	Programação orientada por objetos			Sem.	Ano	Turno
Aluno:		RA:		1º	2025	
Professor:	Charlene e Jardel				NOTA	
Data:	__/__/__		Valor:			

1. (F) O escopo de namespace define que as variáveis declaradas dentro de um namespace podem ser acessadas apenas pela classe que as declarou.
  - Correção: As variáveis podem ser acessadas por outras classes dentro do mesmo namespace.
2. (V) Modificadores de acesso controlam a visibilidade dos atributos e métodos de uma classe.
3. (V) O encapsulamento permite que atributos e métodos fiquem ocultos de quem os utiliza, promovendo modularidade e segurança.
4. (F) Um atributo de instância pertence a todos os objetos de uma classe, ou seja, qualquer modificação em um objeto afeta os outros.
  - Correção: Cada objeto tem sua própria cópia do atributo de instância.
5. (V) Métodos estáticos só podem acessar atributos e métodos que também sejam estáticos.
6. (V) A palavra-chave `this` em C# refere-se ao objeto atual da classe.
7. (V) O método `Main` em C# pode ser chamado diretamente sem a necessidade de criar uma instância da classe que o contém.
8. (F) Uma classe declarada como `static` pode ser instanciada e herdada por outras classes.
  - Correção: Classes `static` não podem ser instanciadas nem herdadas.
9. (V) O Garbage Collector do .NET Framework é responsável por liberar objetos não mais referenciados na memória automaticamente.
10. (V) Construtores em C# são métodos especiais que possuem o mesmo nome da classe e têm a função de inicializar objetos.

**Curso: Computação**

<b>Disciplina:</b>	Programação orientada por objetos			<b>Sem.</b>	<b>Ano</b>	<b>Turno</b>
<b>Aluno:</b>		<b>RA:</b>		1º	2025	
<b>Professor:</b>	Charlene e Jardel				<b>NOTA</b>	
<b>Data:</b>	___/___/___		<b>Valor:</b>			

11. (V) Em C#, uma classe pode ter múltiplos destrutores, desde que tenham assinaturas diferentes.
12. (V) Métodos **get** e **set** são usados para manipular atributos privados, promovendo o encapsulamento na programação orientada a objetos.

<b>QUESTÃO</b>	<b>Nº3</b>	<b>VALOR</b>	1	<b>NOTA</b>	
----------------	------------	--------------	---	-------------	--

**Qual é o principal benefício do encapsulamento em programação orientada a objetos?**

- A) Aumentar o desempenho do programa.
- B) Tornar os atributos acessíveis diretamente por qualquer classe.
- C) Melhorar a segurança e modularidade do código.**
- D) Permitir que os métodos estáticos sejam acessíveis sem instância.

<b>QUESTÃO</b>	<b>Nº4</b>	<b>VALOR</b>	1	<b>NOTA</b>	
----------------	------------	--------------	---	-------------	--

**. Qual dos seguintes modificadores de acesso permite que um atributo seja acessado apenas dentro da classe que o declarou?**

- A) public
- B) private**
- C) protected
- D) internal

<b>QUESTÃO</b>	<b>Nº5</b>	<b>VALOR</b>	1	<b>NOTA</b>	
----------------	------------	--------------	---	-------------	--

**O que acontece se uma variável for declarada dentro de um bloco { } e tentarmos acessá-la fora desse bloco?**

- A) Ela pode ser acessada normalmente.
- B) O compilador exibe um erro informando que a variável não existe.**
- C) O valor da variável será substituído pelo valor padrão do seu tipo.
- D) O código continua executando normalmente, mas a variável recebe null.

Curso: Computação

<b>Disciplina:</b>	Programação orientada por objetos			<b>Sem.</b>	<b>Ano</b>	<b>Turno</b>
<b>Aluno:</b>		<b>RA:</b>		1º	2025	
<b>Professor:</b>	Charlene e Jardel				<b>NOTA</b>	
<b>Data:</b>	__/__/__		<b>Valor:</b>			

<b>QUESTÃO</b>	Nº 6	<b>VALOR</b>	1	<b>NOTA</b>	
----------------	------	--------------	---	-------------	--

Qual palavra-chave é usada para indicar que um atributo pertence a uma classe e não a um objeto específico?

- A) static
- B) instance
- C) shared
- D) global

<b>QUESTÃO</b>	Nº 7	<b>VALOR</b>	1	<b>NOTA</b>	
----------------	------	--------------	---	-------------	--

. Um método estático pode acessar diretamente quais tipos de membros da classe?

- A) Apenas atributos e métodos não estáticos.
- B) Apenas atributos e métodos estáticos.
- C) Apenas métodos, mas não atributos.
- D) Qualquer membro da classe, independentemente do escopo.

<b>QUESTÃO</b>	Nº 8	<b>VALOR</b>	1	<b>NOTA</b>	
----------------	------	--------------	---	-------------	--

O que acontece quando um objeto é desalocado da memória em C#?

- A) Ele é automaticamente destruído pelo Garbage Collector.
- B) O programador precisa chamar explicitamente o destrutor da classe.
- C) Ele continua existindo na memória até o fim da execução do programa.
- D) Ele é destruído imediatamente após a execução de seu último método.

Curso: Computação

Disciplina:	Programação orientada por objetos			Sem.	Ano	Turno
Aluno:		RA:		1º	2025	
Professor:	Charlene e Jardel				NOTA	
Data:	___/___/___		Valor:			

QUESTÃO	Nº 9	VALOR	1	NOTA	
---------	------	-------	---	------	--

Qual das seguintes opções NÃO é um tipo de escopo de variável em C#?

- A) Escopo de método
- B) Escopo de instância
- C) Escopo de projeto
- D) Escopo de namespace

QUESTÃO	Nº 10	VALOR	1	NOTA	
---------	-------	-------	---	------	--

Em relação ao escopo de atributos de classe (**static**), qual das afirmações é verdadeira?

- A) Cada objeto da classe tem uma cópia independente dos atributos estáticos.
- B) Os atributos estáticos são compartilhados entre todas as instâncias da classe.
- C) Os atributos estáticos são inicializados com valores aleatórios.
- D) Um atributo **static** pode ser acessado apenas dentro da classe onde foi declarado.

QUESTÃO	Nº 11	VALOR	1	NOTA	
---------	-------	-------	---	------	--

Qual é a principal característica de um método construtor?

- A) Possui o mesmo nome da classe e inicializa os atributos do objeto.
- B) Pode ter qualquer nome e é responsável por destruir objetos.
- C) Sempre retorna um valor do tipo **void**.
- D) Só pode ser chamado uma única vez na execução do programa.

QUESTÃO	Nº 12	VALOR	1	NOTA	
---------	-------	-------	---	------	--

Curso: Computação

<b>Disciplina:</b>	Programação orientada por objetos			<b>Sem.</b>	<b>Ano</b>	<b>Turno</b>
<b>Aluno:</b>		<b>RA:</b>		1º	2025	
<b>Professor:</b>	Charlene e Jardel				<b>NOTA</b>	
<b>Data:</b>	__/__/__		<b>Valor:</b>			

Qual é a forma correta de chamar um método estático de uma classe Calculadora que possui um método Somar?

- A) Calculadora calc = new Calculadora(); calc.Somar(5, 3);
- B) calc.Somar(5, 3);
- C) Calculadora.Somar(5, 3);**
- D) new Calculadora().Somar(5, 3);

<b>QUESTÃO</b>	Nº 13	<b>VALOR</b>	1	<b>NOTA</b>	
----------------	-------	--------------	---	-------------	--

. O que acontece se uma classe for declarada como **static**?

- A) Ela pode ser instanciada, mas não pode ser herdada.
- B) Ela não pode ser instanciada nem herdada.**
- C) Ela pode ser herdada, mas não pode ser instanciada.
- D) Ela funciona como qualquer outra classe comum.

<b>QUESTÃO</b>	Nº 14	<b>VALOR</b>	1	<b>NOTA</b>	
----------------	-------	--------------	---	-------------	--

O que caracteriza a sobrecarga de construtores em C#?

- A) Múltiplos construtores com nomes diferentes na mesma classe.
- B) Construtores que possuem a mesma assinatura.
- C) Construtores com o mesmo nome da classe, mas assinaturas diferentes.**
- D) Apenas um único construtor pode existir em uma classe.

<b>QUESTÃO</b>	Nº 14	<b>VALOR</b>	1	<b>NOTA</b>	
----------------	-------	--------------	---	-------------	--

Curso: Computação

Disciplina:	Programação orientada por objetos			Sem.	Ano	Turno
Aluno:		RA:		1º	2025	
Professor:	Charlene e Jardel				NOTA	
Data:	___/___/___		Valor:			

Qual dos seguintes exemplos mostra a sintaxe correta de um destrutor em C#?

- A) ~ClasseNome { // Código }
- B) void ~ClasseNome() { // Código }
- C) public ~ClasseNome() { // Código }
- D) delete ClasseNome { // Código }

QUESTÃO	Nº 15	VALOR	1	NOTA	
---------	-------	-------	---	------	--

Como podemos garantir que um atributo **private** seja acessado de forma segura?

- A) Utilizando métodos get e set.
- B) Alterando o modificador para public.
- C) Criando uma variável global que armazene os valores do atributo.
- D) Usando diretamente this.atributo fora da classe.

QUESTÃO	Nº 16	VALOR	1	NOTA	
---------	-------	-------	---	------	--

Se um método **static** tentar acessar um atributo não estático diretamente, o que acontece?

- A) O código compila sem erros.
- B) O compilador gera um erro, pois métodos estáticos só podem acessar membros estáticos.
- C) O código entra em um loop infinito.
- D) O método converte automaticamente o atributo para estático.

QUESTÃO	Nº 17	VALOR	1	NOTA	
---------	-------	-------	---	------	--

Como um destrutor é chamado em C#?

- A) Pelo programador, usando **delete**.

Curso: Computação

Disciplina:	Programação orientada por objetos			Sem.	Ano	Turno
Aluno:		RA:		1º	2025	
Professor:	Charlene e Jardel				NOTA	
Data:	__/__/__		Valor:			

B) Automaticamente pelo Garbage Collector.

C) Chamando **Dispose()** no objeto.

D) Invocando **Destroy()** manualmente.

QUESTÃO	Nº 18	VALOR	1	NOTA	
---------	-------	-------	---	------	--

Qual é a principal diferença entre um atributo de instância e um atributo estático?

A) O atributo de instância pertence à classe, enquanto o estático pertence ao objeto.

B) O atributo estático é compartilhado por todas as instâncias da classe.

C) O atributo de instância pode ser acessado sem instanciar um objeto.

D) Ambos têm comportamento idêntico, mas o estático ocupa menos memória.

QUESTÃO	Nº 19	VALOR	1	NOTA	
---------	-------	-------	---	------	--

Qual palavra-chave permite que uma classe filha acesse atributos protegidos da classe pai?

A) private

B) public

C) internal

D) protected

QUESTÃO	Nº 20	VALOR	1	NOTA	
---------	-------	-------	---	------	--



Curso: Computação

Disciplina:	Programação orientada por objetos			Sem.	Ano	Turno
Aluno:		RA:		1º	2025	
Professor:	Charlene e Jardel				NOTA	
Data:	___/___/___	Valor:				

csharp

```
class Pessoa {
    private string nome;

    public void SetNome(string novoNome) {
        nome = novoNome;
    }

    public string GetNome() {
        return nome;
    }
}
```

Dado o código em C#, o que acontece se tentarmos acessar pessoa.nome diretamente na função Main()?

- A) O código compila e nome pode ser acessado normalmente.
- B) O código não compila porque nome é privado e só pode ser acessado por métodos da classe.
- C) O código gera um erro em tempo de execução porque nome não foi inicializado.**
- D) O código compila, mas nome retorna sempre null porque não foi inicializado corretamente.

QUESTÃO	Nº 21	VALOR	1	NOTA	
---------	-------	-------	---	------	--

Curso: Computação

Disciplina:	Programação orientada por objetos			Sem.	Ano	Turno
Aluno:		RA:		1º	2025	
Professor:	Charlene e Jardel				NOTA	
Data:	___/___/___		Valor:			

```
csharp

class Carro {
    public string modelo;

    public Carro(string modelo) {
        this.modelo = modelo;
    }
}

class Program {
    static void Main() {
        Carro c = new Carro("Sedan");
        Console.WriteLine(c.modelo);
    }
}
```

Qual a saída do código?

- A) Sedan
- B) null
- C) Erro de compilação, pois o construtor está mal definido
- D) Erro de execução, pois a variável modelo não foi inicializada

QUESTÃO	Nº 22	VALOR	1	NOTA	
---------	-------	-------	---	------	--

Curso: Computação

Disciplina:	Programação orientada por objetos			Sem.	Ano	Turno
Aluno:		RA:		1º	2025	
Professor:	Charlene e Jardel				NOTA	
Data:	__/__/__		Valor:			

```
class Contador {
    private static int total = 0;

    public Contador() {
        total++;
    }

    public static int GetTotal() {
        return total;
    }
}

class Program {
    static void Main() {
        Contador c1 = new Contador();
        Contador c2 = new Contador();
        Contador c3 = new Contador();

        Console.WriteLine(Contador.GetTotal());
    }
}
```

O que será impresso ao executar o código?

- A) 1
- B) 2
- C) 3
- D) Erro de compilação