

Delegates , Lambda, LINQ

1- Considere as seguintes assertivas sobre delegates na linguagem C#. Indique se cada uma é verdadeira ou falsa:

- a-) Um delegate é um tipo de dado que pode ser usado para encapsular um método e chamá-lo posteriormente.
- b-) Para declarar um delegate em C#, é necessário especificar o tipo de retorno do método que ele encapsula, bem como os tipos de parâmetros.
- c-) É possível usar o operador += para adicionar vários métodos a um delegate, que serão chamados em sequência quando o delegate for invocado.
- d-) O delegate pode ser passado como parâmetro para outro método, permitindo que o método receptor invoque o método encapsulado no delegate.
- e-) O uso de delegates pode melhorar a modularidade e a flexibilidade do código, permitindo que um método seja substituído por outro sem que isso afete o restante do código.
- f-) Delegates só podem ser usados para encapsular métodos estáticos, não métodos de instância.
- g-) O delegate é um recurso exclusivo da linguagem C#, e não pode ser encontrado em outras linguagens de programação.
- h-) Delegates são frequentemente usados em eventos, para permitir que um objeto notifique outros objetos quando ocorre alguma ação.

2- Implemente um programa que use delegates para somar dois números inteiros. O programa deve seguir as seguintes especificações:

- a-Defina um delegate chamado OperacaoMatematica que recebe dois inteiros como parâmetros e retorna um inteiro.
- b-Defina um método que implementam a operação de soma: que recebe os dois inteiros como parâmetros
- c-Crie um objeto delegate do tipo OperacaoMatematica e atribua a ele ao método definido.
- d-Chame o delegate passando os parâmetros necessários e exiba o resultado da operação na tela.

Dica: Utilize o método Invoke do delegate para invocar a operação matemática.

3- Implemente um programa em C# que use multicast delegates para exibir uma mensagem de boas-vindas ao usuário. O programa deve seguir as seguintes especificações:

- a-) Defina um multicast delegate chamado MensagemBoasVindas que não recebe parâmetros e não retorna nenhum valor.
- b-) Defina duas funções que exibem uma mensagem de boas-vindas: uma em português e outra em inglês.
- c-) Crie um objeto delegate do tipo MensagemBoasVindas e adicione as duas funções criadas a ele usando o operador +=.
- d-) Chame o delegate para executar as duas funções e exibir as mensagens de boas-vindas.

4- Dado o código abaixo, assinale com verdadeiro ou falso cada afirmação sobre os delegates Action, Predicate e Func da linguagem C#:

```
int[] numeros = { 1, 2, 3, 4, 5 };

// Delegate Action
Action<int> exibirNumero = num => Console.WriteLine(num);
Array.ForEach(numeros, exibirNumero);

// Delegate Predicate
Predicate<int> ehPar = num => num % 2 == 0;
bool todosPares = Array.TrueForAll(numeros, ehPar);

// Delegate Func
```

Delegates , Lambda, LINQ

```
Func<int, int, int> somar = (a, b) => a + b;  
int resultado = somar(10, 20);  
  
Console.WriteLine(todosPares);  
Console.WriteLine(resultado);  
  
Console.ReadKey();
```

- a- O delegate Action é usado para encapsular um método que recebe um parâmetro e não retorna nenhum valor. (Verdadeiro ou Falso)
- b- O método Array.ForEach é usado para percorrer todos os elementos de um array e executar um delegate para cada um deles. (Verdadeiro ou Falso)
- c- O delegate Predicate é usado para encapsular um método que recebe um parâmetro e retorna um valor booleano. (Verdadeiro ou Falso)
- d- O método Array.TrueForAll retorna true se todos os elementos de um array satisfazem uma condição especificada por um delegate. (Verdadeiro ou Falso)
- e- O delegate Func é usado para encapsular um método que recebe um ou mais parâmetros e retorna um valor. (Verdadeiro ou Falso)
- f- O delegate Func pode ter até 16 parâmetros de entrada e um tipo de retorno genérico. (Verdadeiro ou Falso)
- g- No exemplo acima, o delegate Func é usado para definir um método que soma dois números inteiros. (Verdadeiro ou Falso)
- h- No exemplo acima, o método somar recebe dois parâmetros e retorna um valor. (Verdadeiro ou Falso)

5- Imagine que você precisa criar um método que imprima na tela os números pares de uma lista de inteiros de 1 a 20.

Para isso, você deve utilizar um delegate Action<int> que receba um número inteiro como parâmetro e imprima na tela apenas os números pares.

6- Crie um programa onde um método que receba dois números inteiros e retorne o resultado da operação de adição ou subtração desses números, a ser definida em tempo de execução por um delegate Func<int, int, int>.

7- Suponha que você possui uma lista de planetas com informações sobre o nome, diâmetro e massa de cada planeta.

Crie um programa e implemente um método que recebe essa lista e um delegate Predicate<Planeta> para filtrar os planetas de acordo com um critério específico.

8 - Qual é a definição correta de expressão lambda em C#?

- a) Uma função anônima que pode ser passada como argumento para outros métodos e que pode ser definida em apenas uma linha de código.
- b) Uma função que sempre retorna um valor booleano.
- c) Uma função que sempre tem um único parâmetro de entrada.
- d) Uma função que sempre retorna um valor inteiro.

9- Crie um método de extensão para a classe List<int> na linguagem C# que, dado uma lista de inteiros, retorne a soma de todos os elementos da lista que são ímpares.

Em seguida, crie um programa que instancie uma lista de inteiros com alguns números ímpares e alguns números pares e use o método de extensão criado para obter a soma dos números ímpares da lista.

Delegates , Lambda, LINQ

10 - Exercicios sobre LINQ

Exercício 1: Utilizando o método Where

Dado um array de strings, escreva um programa em C# que use a biblioteca LINQ para obter somente as strings que contenham a letra 'a'.

Exercício 2: Utilizando o método OrderBy

Dado um array de inteiros, escreva um programa em C# que use a biblioteca LINQ para ordenar os valores em ordem crescente.

Exercício 3: Utilizando o método GroupBy

Dado um array de strings, escreva um programa em C# que use a biblioteca LINQ para agrupar as strings por tamanho

Exercício 4: Utilizando o método FirstOrDefault

Dado um array de inteiros, escreva um programa em C# que use a biblioteca LINQ para obter o primeiro número par do array.