

Olá, terráqueos! Meu nome é Lucas Eduardo de Freitas, atualmente trabalho como Desenvolvedor Fullstack JR, e estou sempre buscando ampliar meus horizontes, para poder evoluir cada dia mais! Neste documento, estarei lhes apresentando a Documentação referente ao **primeiro exercício do Desafio de Lógica de Programação**. Então, sem mais delongas, vamos lá!

## # Questão 01

A **mediana** de uma lista de números é basicamente o elemento que se encontra no meio da lista após a ordenação. Dada uma lista de números com um número ímpar de elementos, desenvolva um algoritmo que encontre a **mediana**.

Conforme visualizamos no enunciado da primeira questão, teremos que criar um algoritmo que, após ordenar uma lista de números (obrigatoriamente ímpar), encontre a Mediana referente a mesma.

Com isso, criaremos um **Console Application**, que possibilitará este processo.

Após criar nossa aplicação, vamos criar a classe **Mediana.cs**, que possuirá os valores iniciais e o nosso resultado esperado, acompanhado de um método para validar se nossa lista contém um número ímpar/par de números. O código ficará da seguinte forma:

```
1
2 namespace Questao1
3 {
4     public class EncontraMediana
5     {
6         public int[] listaNumeros = new int[] { 9, 2, 1, 8, 4, 6, 3, 7, 5 };
7
8         public int mediana = 0;
9
10        public bool VerificaImparPar()
11        {
12            if (listaNumeros.Length % 2 == 0)
13            {
14                return false;
15            }
16            else
17            {
18                return true;
19            }
20        }
21    }
22 }
23
24
25
```

A variável *listaNumeros* será a responsável por apresentar a lista de valores que necessitamos.

Na classe **Program.cs**, será onde todas as nossas ações serão executadas, conforme abaixo:

```
4 namespace Questao1
5 {
6     O referências
7     class Program
8     {
9         O referências
10        static void Main(string[] args)
11        {
12            Console.WriteLine("QUESTÃO 1 - ENCONTRAR A MEDIANA:\n");
13
14            Console.Write("Calculando");
15            Thread.Sleep(600);
16            Console.Write(".");
17            Thread.Sleep(700);
18            Console.Write(".");
19            Thread.Sleep(800);
20            Console.WriteLine("\n");
21            Thread.Sleep(900);
22
23            EncontraMediana encontraMediana = new();
24
25            bool impar_par = encontraMediana.VerificaImparPar();
26
27            if(impar_par == true)
28            {
29                Array.Sort(encontraMediana.listaNumeros);
30
31                int posicao_valor = (encontraMediana.listaNumeros.Length + 1) / 2;
32                encontraMediana.mediana = encontraMediana.listaNumeros[posicao_valor - 1];
33
34                Console.WriteLine("A mediana é: {0} \n", encontraMediana.mediana);
35                Thread.Sleep(1000);
36            }
37            else
38            {
39                Console.WriteLine("Esta lista contém um total de números pares, por favor, tente novamente.\n");
40                Thread.Sleep(2000);
41            }
42        }
43    }
44 }
```

Como podemos visualizar, no início, há uma apresentação simples da nossa aplicação, seguida pela instanciação de uma variável para que seja possível utilizar os métodos da classe **Mediana.cs**, onde, logo em seguida, já efetuaremos a validação de nossa lista, que, conforme visualizamos anteriormente, possui 9 números, então, a mesma seguirá rodando a aplicação.

Após a validação, ordenaremos nossa lista de forma crescente, e então, aplicar a quantidade de números + 1 e depois dividi-la por 2, tudo isso guardando este resultado na variável *posicao\_valor*.

Após isso, atribuiremos nossa lista de números com base em *posicao\_valor* - 1 para a variável *mediana*, pertencente a classe **Mediana.cs**.

Com isso, basta escrever esta mesma variável em tela, para possibilitar nossa visualização:

```
Console de Depuração do Microsoft Visual Studio

QUESTÃO 1 - ENCONTRAR A MEDIANA:

Calculando...

A mediana é: 5

FIM DA QUESTÃO 1.

=====
```

## PROVA REAL:

Temos nossa lista: “9, 2, 1, 8, 4, 6, 3, 7, 5”, então, inicialmente, podemos visualizar que ela possui 9 dígitos, que é ímpar, então seguiremos com a análise.

Após isso, vamos ordená-la, ficará da seguinte maneira: “1, 2, 3, 4, 5, 6, 7, 8, 9”.

Agora, pegaremos esta quantidade total de números e adicionar 1, e logo após isso, dividir este total por 2. Que ficará:  $(9 + 1) / 2 = 5$ .

Ou seja, nossa mediana se encontrará na Quinta posição de nossa lista, sendo por coincidência o próprio número “5”.

=====

Agora, para termos certeza sobre nossos resultados, vamos utilizar a lista de exemplo dada no próprio enunciado, onde a saída deverá ser “4”:

Entrada:

```
arr = [9, 2, 1, 4, 6]
```

Saída:

```
4
```

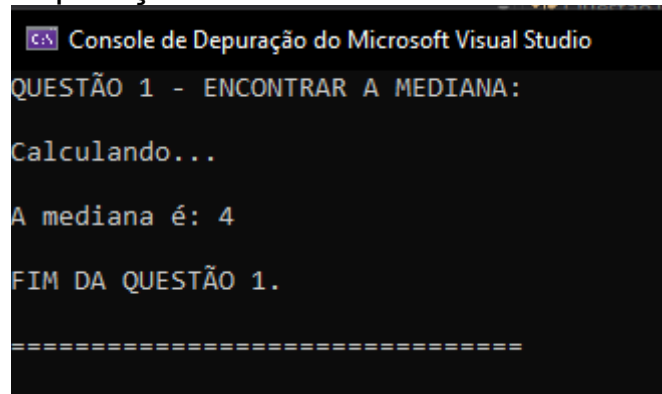
Alteramos nosso código:

```
namespace Questao1
{
    2 referências
    public class EncontraMediana
    {
        public int[] listaNumeros = new int[] { 9, 2, 1, 4, 6 };

        public int mediana = 0;

        1 referência
        public bool VerificaImparPar()
        {
            if (listaNumeros.Length % 2 == 0)
            {
                return false;
            }
            else
            {
                return true;
            }
        }
    }
}
```

E rodaremos nossa aplicação novamente:

A screenshot of the 'Console de Depuração do Microsoft Visual Studio' window. The text inside the console is as follows:

```
QUESTÃO 1 - ENCONTRAR A MEDIANA:  
  
Calculando...  
  
A mediana é: 4  
  
FIM DA QUESTÃO 1.  
  
=====
```

Maravilha! Os resultados se equivalem, seguindo a mesma explicação de nossa prova real.

Agradeço pela atenção até o fim desta documentação, decidi escrevê-la de forma mais pessoa, pois assim facilitaria a interpretação de quaisquer grupos de pessoas!

Atenciosamente, Lucas Eduardo de Freitas

**</end>**