

## Jagged Arrays

### Transcrição

Neste vídeo, veremos um problema que acontece bastante na prática, que é a impressão de nomes de pessoas de certas famílias.

Temos as famílias Flinstones, Simpsons e Dona Florinda.

Vamos inserir seus nomes na aplicação.

Primeiro, criaremos um array de famílias:

```
namespace A2._2_JaggedArrays
{
    class Program
    {
        static void Main(string[] args)
        {
            //família 1: Família Flinstones
            //família 2: Família Simpsons
            //família 3: Família Dona Florinda

            string[] familias = new string[]
            {
                "Fred", "Wilma", "Pedrita",
                "Homer", "Marge", "Lisa", "Bart", "Maggie",
                "Florinda", "Kiko"
            };
        }
    }
}
```

Para imprimir os nomes, criaremos um laço foreach :

```
namespace A2._2_JaggedArrays
{
    class Program
    {
        static void Main(string[] args)
        {
            //família 1: Família Flinstones
            //família 2: Família Simpsons
            //família 3: Família Dona Florinda

            string[] familias = new string[]
            {
                "Fred", "Wilma", "Pedrita",
                "Homer", "Marge", "Lisa", "Bart", "Maggie",
                "Florinda", "Kiko"
            };

            foreach (var pessoa in familias)
            {
            }
        }
    }
}
```

```
        Console.WriteLine(pessoa);  
    }  
}
```

Executaremos o programa. Veremos que serão impressos todos os nomes, de todas as famílias.

Nosso objetivo é fazer com que sejam impressos em linhas horizontais, dentro de suas respectivas famílias.

Para isso, faremos algumas alterações.

Colocamos os nomes dos membros das famílias entre chaves, e declaramos o array como multidimensional.

```
string[,] familias = new string[3, ]  
{  
    {"Fred", "Wilma", "Pedrita"},  
    {"Homer", "Marge", "Lisa", "Bart", "Maggie"},  
    {"Florinda", "Kiko"}  
};
```

Ao declararmos o tamanho do array, o número de famílias é 3, mas o número de nomes em cada uma varia entre 2 e 5.

Isto significa que o array não é retangular, tem uma quantidade diferente de elementos por linha.

Neste caso, teremos que utilizar um tipo diferente de array, o `JaggedArray`.

É um tipo de array denteado, ou serrilhado. Ele conterá as três famílias.

Como a segunda dimensão não é a mesma para todos os itens, criaremos mais um par de colchetes, ao lado de cada um dos pares já existentes.

```
string [][] familias = new string [3][]
```

Outra mudança será no modo como listamos todos os nomes, não podemos iniciar um array denteado da forma que está.

Para isso, faremos:

```
string [][] familias = new string [3][]  
  
familias[0] = new string[] {"Fred", "Wilma", "Pedrita"};  
familias[1] = new string[] {"Homer", "Marge", "Lisa", "Bart", "Maggie"};  
familias[2] = new string[] {"Florinda", "Kiko"};
```

Resultando no seguinte código:

```
namespace A2._2_JaggedArrays  
{  
    class Program
```

```
{
    static void Main(string[] args)
    {
        //família 1: Família Flinstones
        //família 2: Família Simpsons
        //família 3: Família Dona Florinda

        //JAGGED ARRAY = ARRAY DENTEADO = ARRAY SERRILHADO

        string[][] familias = new string[3][]
        //{
        //    {"Fred", "Wilma", "Pedrita"},
        //    {"Homer", "Marge", "Lisa", "Bart", "Maggie"},
        //    {"Florinda", "Kiko"}
        //};

        familias[0] = new string[] {"Fred", "Wilma", "Pedrita"};
        familias[1] = new string[] {"Homer", "Marge", "Lisa", "Bart", "Maggie"};
        familias[2] = new string[] {"Florinda", "Kiko"};

        foreach (var pessoa in familias)
        {
            Console.WriteLine(pessoa);
        }
    }
}
```

Agora executaremos a aplicação, utilizando o comando "Ctrl + F5".

Ainda não conseguimos enxergar os nomes das famílias, o que significa que há um problema no programa.

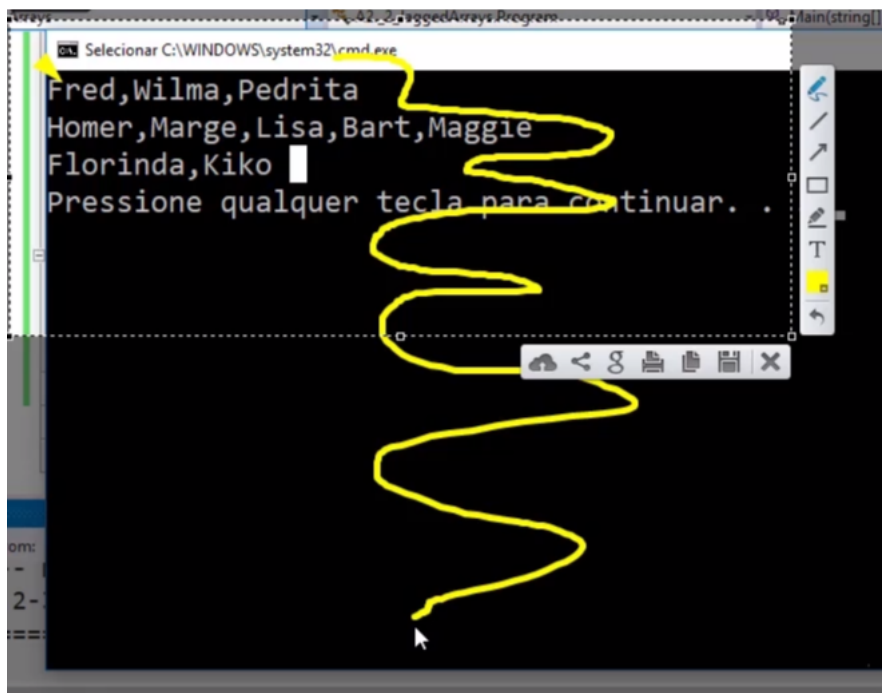
Para que isso não aconteça, precisaremos realizar a impressão de outra forma.

```
foreach (var familia in familias)
{
    Console.WriteLine(string.Join(", ", familia));
}
```

Executaremos o programa, e agora vemos que temos os nomes, em linhas horizontais, separados por famílias.

Mas por que chamamos de array denteado, ou serrilhado?

Este nome é devido ao fato de que, por conterem números diferentes de elementos, quando listados, parecem fazer uma imagem de dentes, ou serras, vide figura abaixo:



Não é nem um array simples, nem um multidimensional.

Ele também é chamado de arrays de arrays, porque cada elemento de um array denteado, também é um array.

Com isso, fechamos este capítulo, no qual trabalhamos com arrays multidimensionais.