



Algoritmos e Linguagem de Programação

Capítulo 5

Programação com Repetição

Prof. Me. Renato Carioca Duarte



Programação com Laços - Objetivos

- Apresentar detalhes sobre o uso de laços para repetição de trechos de programas.
- São apresentadas as três formas populares de laço:
 - condicional pré-teste, condicional pós-teste e incondicional.



Laços e Detalhes Operacionais

- Sabemos que a capacidade operacional de um computador é executar programas e que os programas (obtidos a partir de algoritmos) são sequências de instruções que dão certa ordem de execução a um computador.
- Por vezes algumas dessas ordens devem ser repetidas , e para fazer esse tipo de ação temos a técnica de uso dos laços de repetição.
- Os laços que podemos usar com a linguagem C# são três:
 - um laço de ação condicional pré-teste (com a condição de controle no início do laço),
 - um laço de ação condicional pós-teste (com a condição de controle no fim do laço) e
 - um laço de ação incondicional operacionalizado por uma variável de controle.



Laço Condicional Pré-Teste

- Para a execução de laço condicional pré-teste temos na linguagem C# a instrução **while**, que permite executar um conjunto de instruções enquanto a condição verificada permanecer válida.
- No momento em que essa condição não for válida, o processamento da rotina será desviado para fora do laço. A sintaxe dessa estrutura de laço corresponde a:

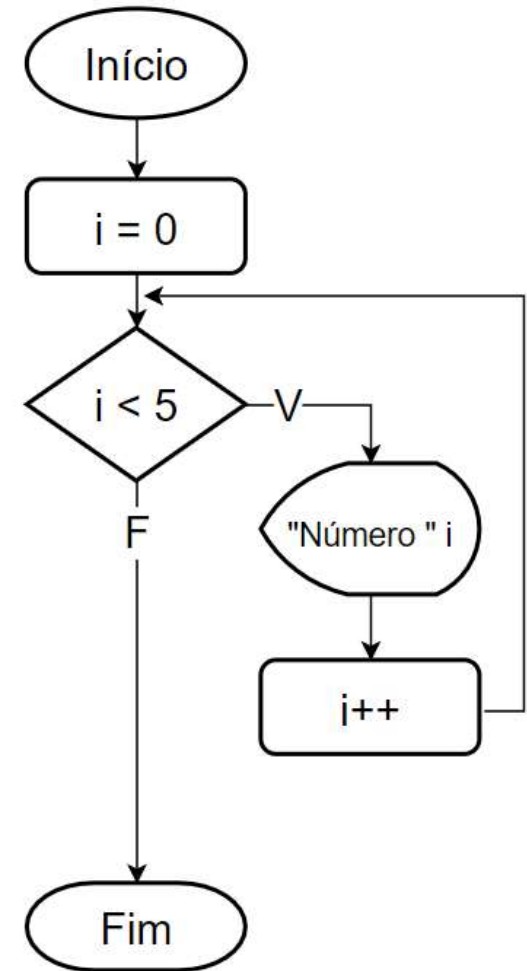
```
while <(condição)>
{
    <instrução 1 executada enquanto a condição for verdadeira>
    <instrução 2 executada enquanto a condição for verdadeira>
    <instrução 3 executada enquanto a condição for verdadeira>
    <instrução N executada enquanto a condição for verdadeira>
}
```



Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

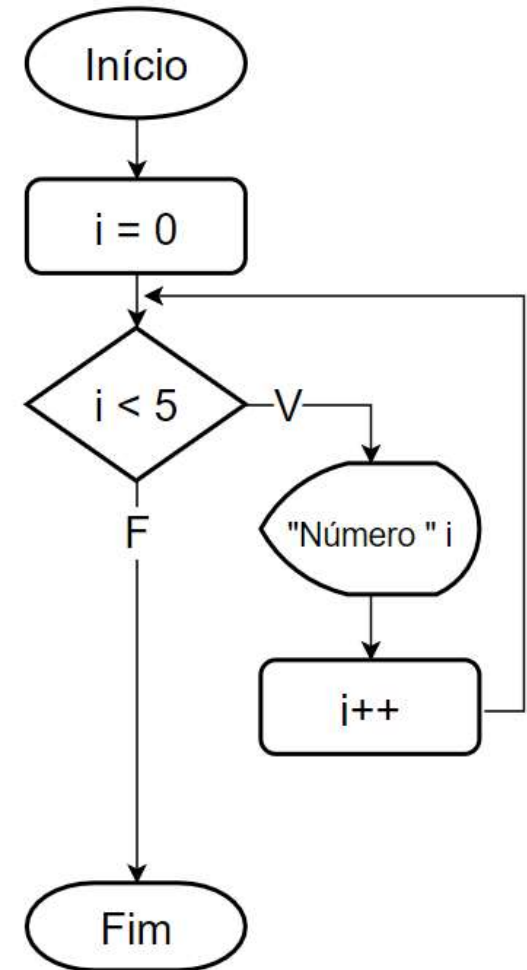
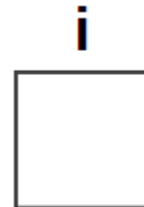




Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```





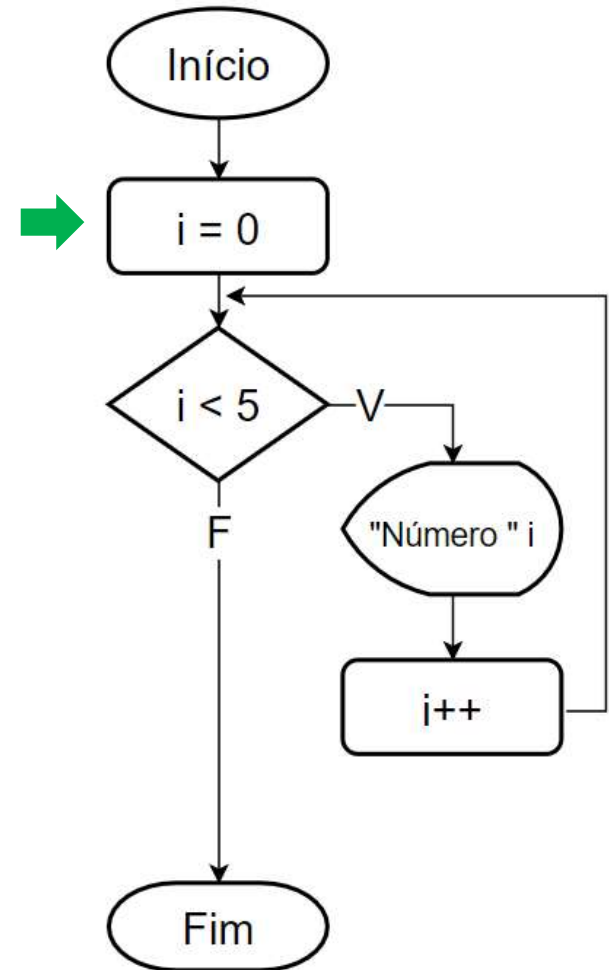
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```



i
0





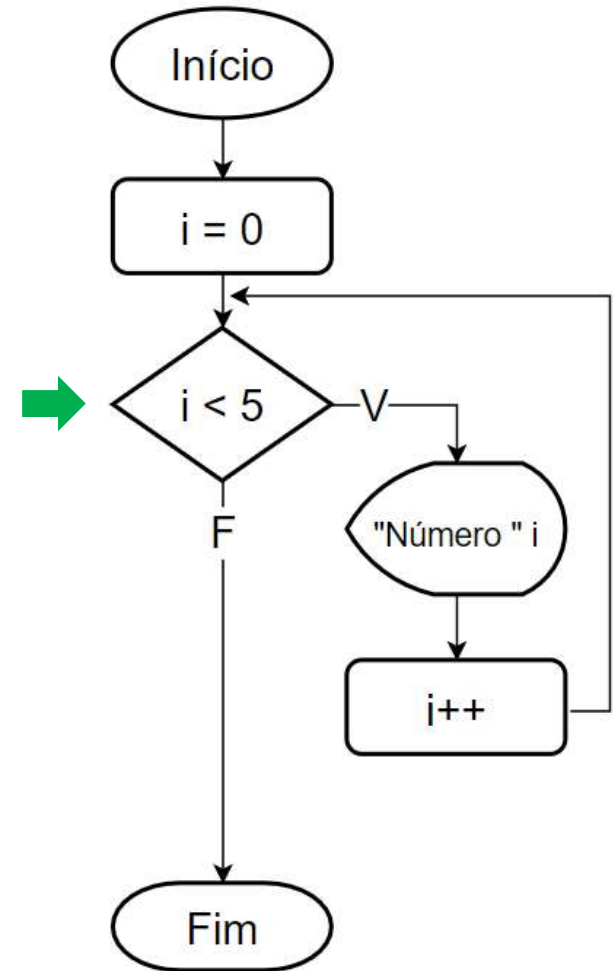
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    → while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```



i
0





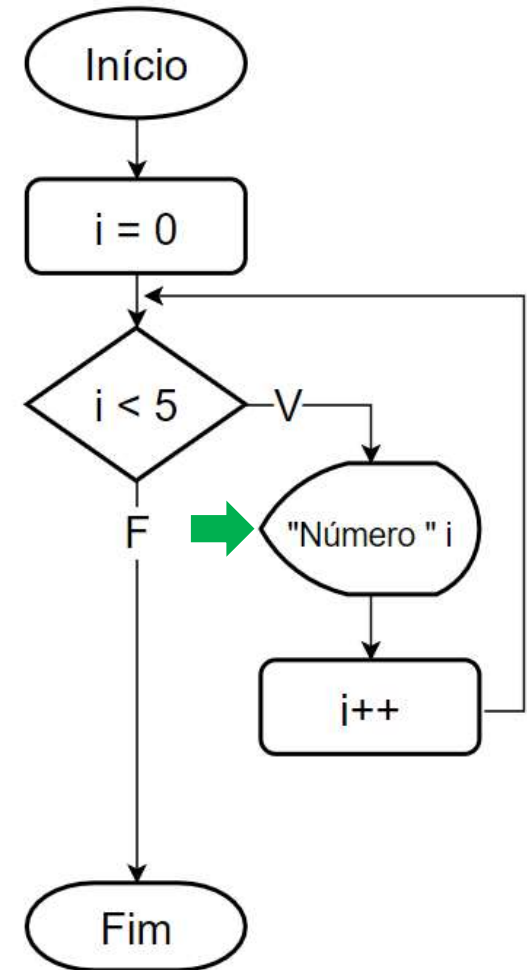
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

Numero 0

i
0





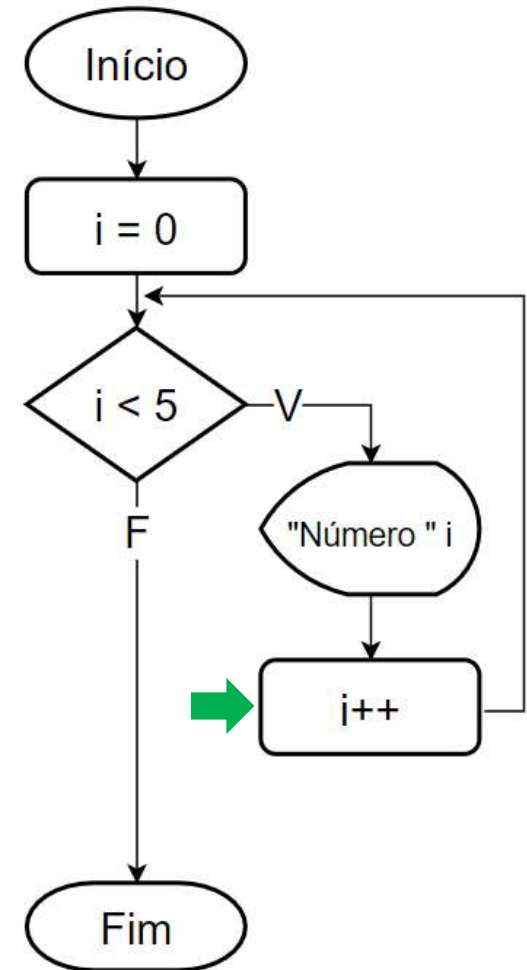
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

Número 0

i
1





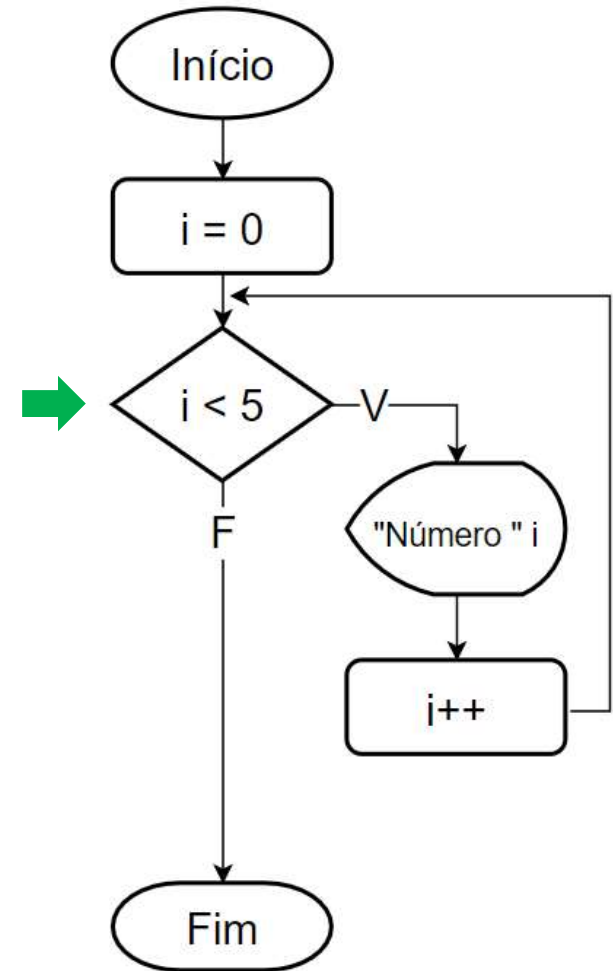
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    → while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

Numero 0

i
1





Exemplo - while

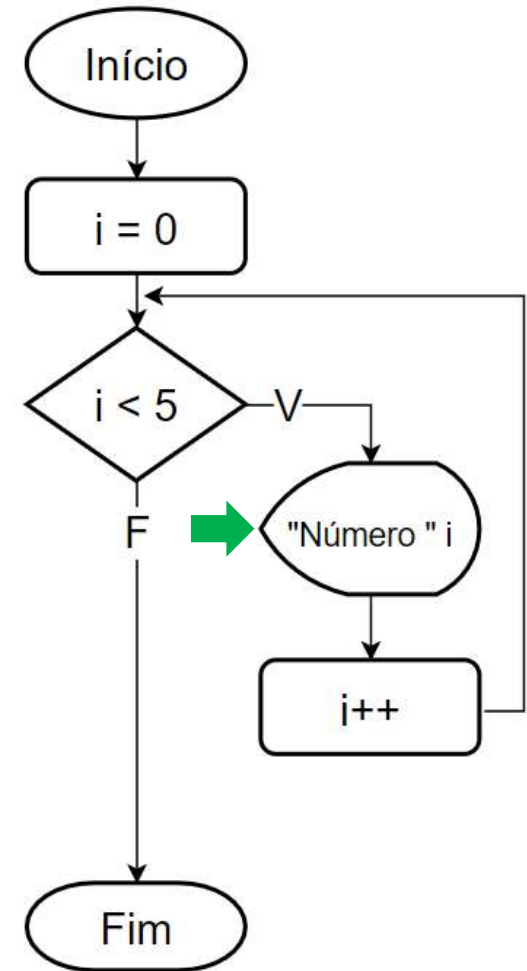
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

```
Numero 0
Numero 1
```

i

1





Exemplo - while

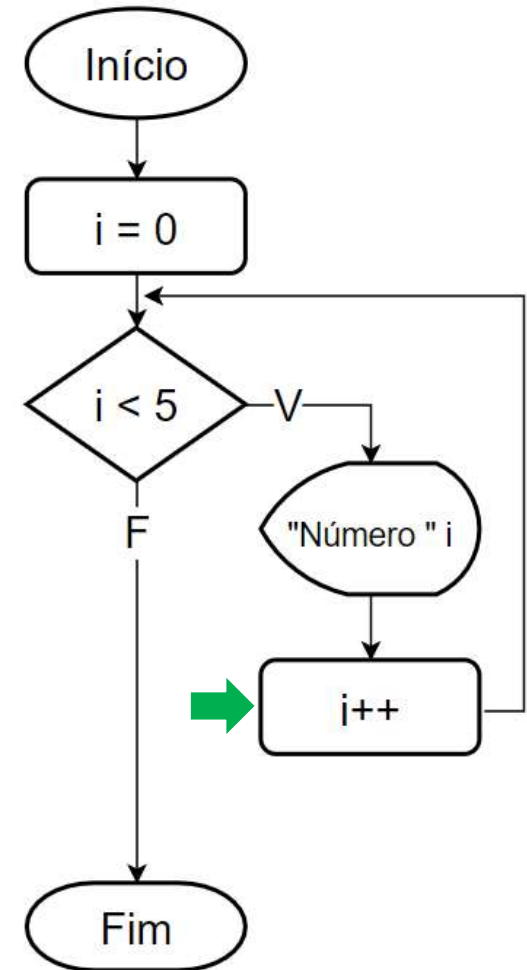
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

```
Numero 0
Numero 1
```

i

2





Exemplo - while

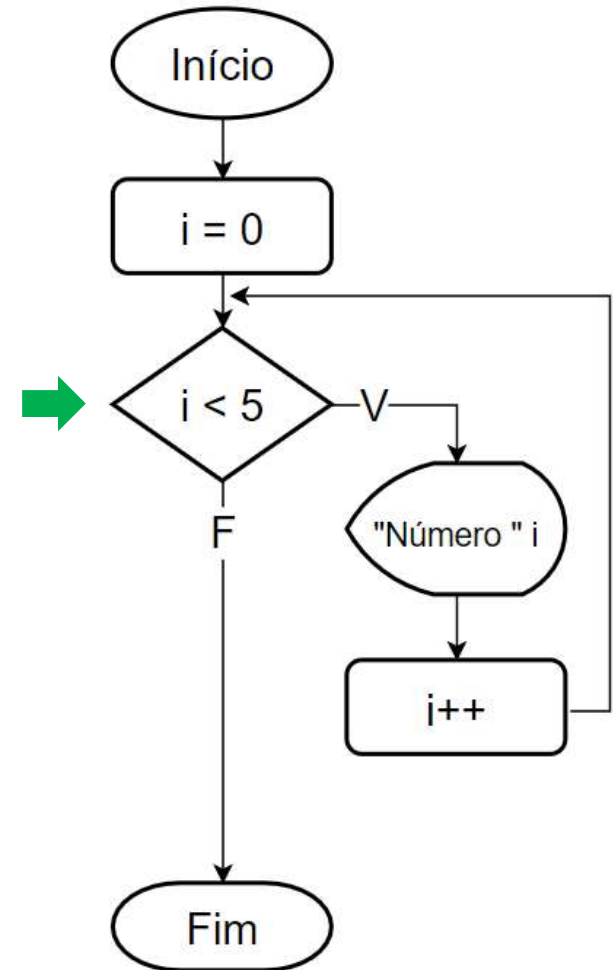
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    → while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

```
Numero 0
Numero 1
```

i

2





Exemplo - while

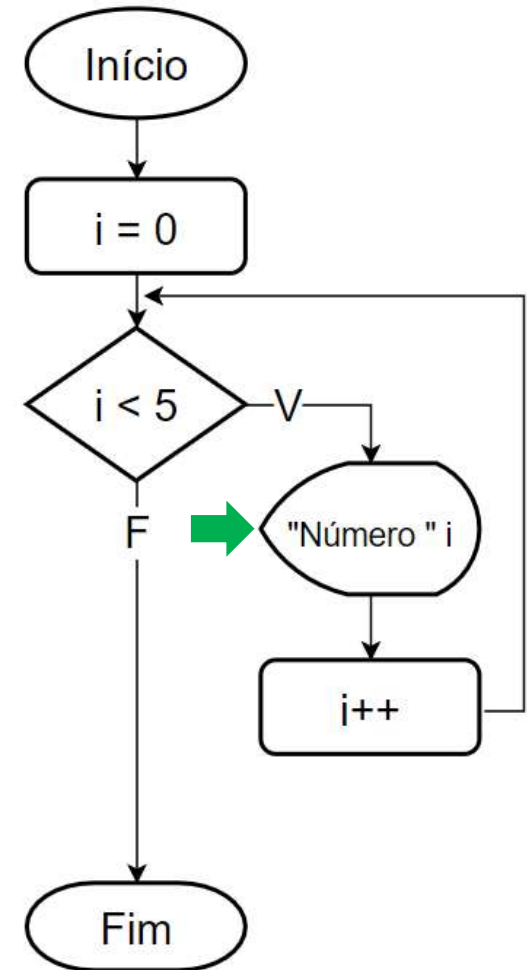
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

```
Numero 0
Numero 1
Numero 2
```

i

2





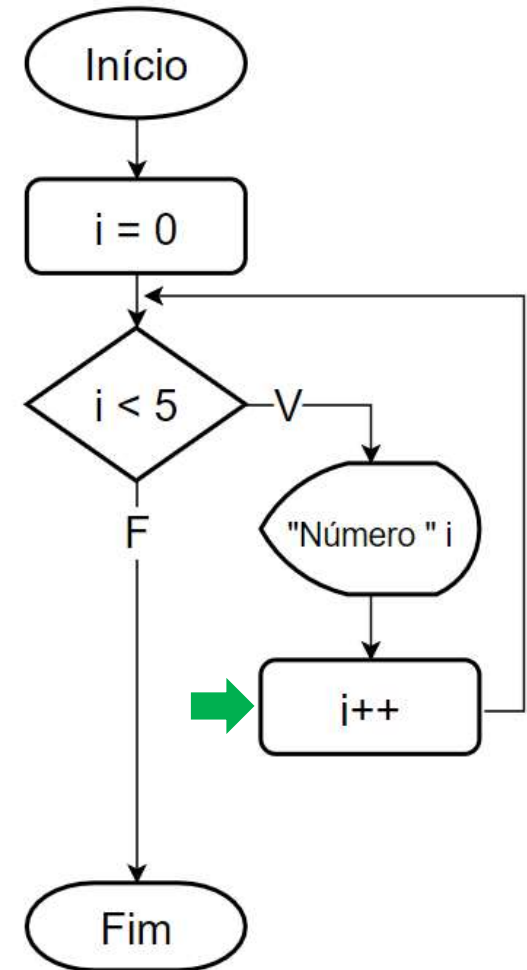
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

```
Numero 0
Numero 1
Numero 2
```

i
3





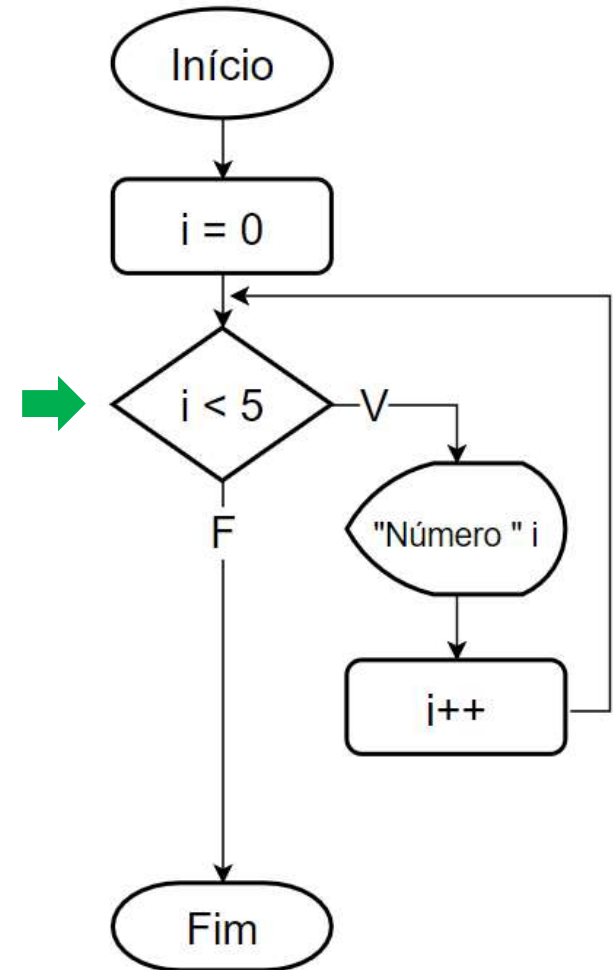
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    → while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

```
Numero 0
Numero 1
Numero 2
```

i
3





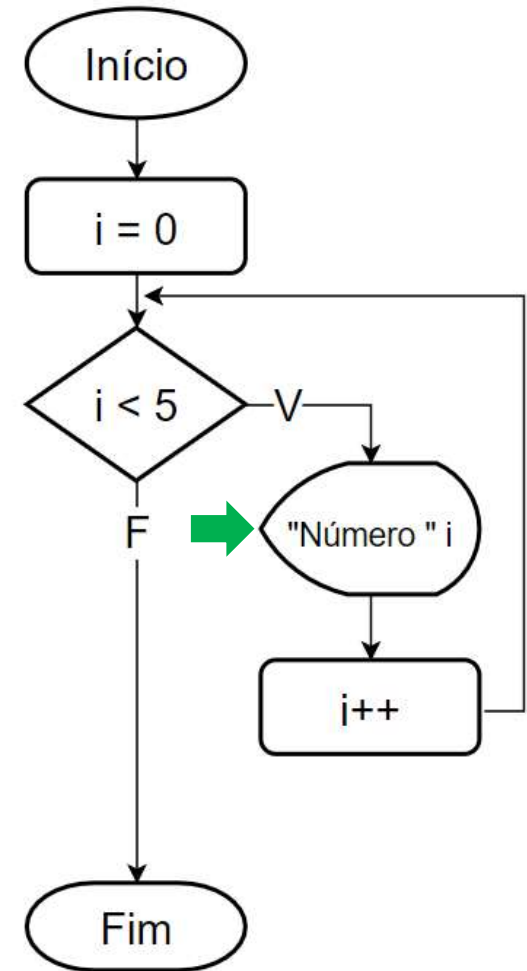
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

```
Número 0
Número 1
Número 2
Número 3
```

i
3





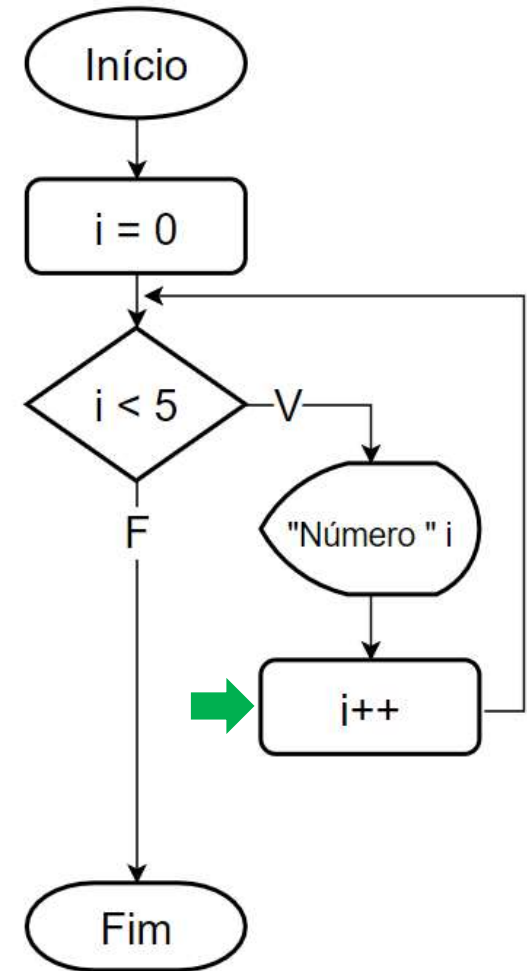
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

```
Número 0
Número 1
Número 2
Número 3
```

i
4





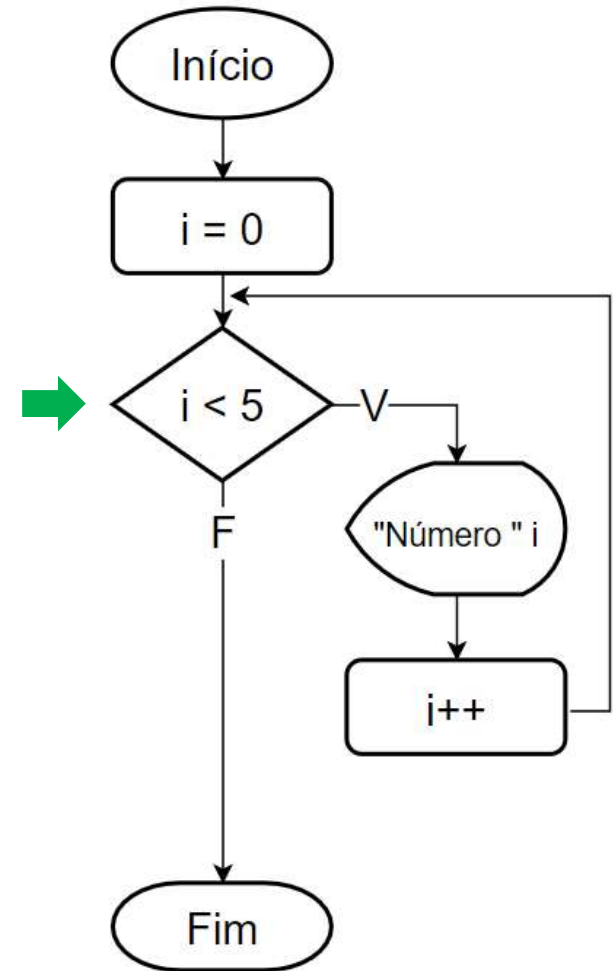
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    → while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

```
Número 0
Número 1
Número 2
Número 3
```

i
4





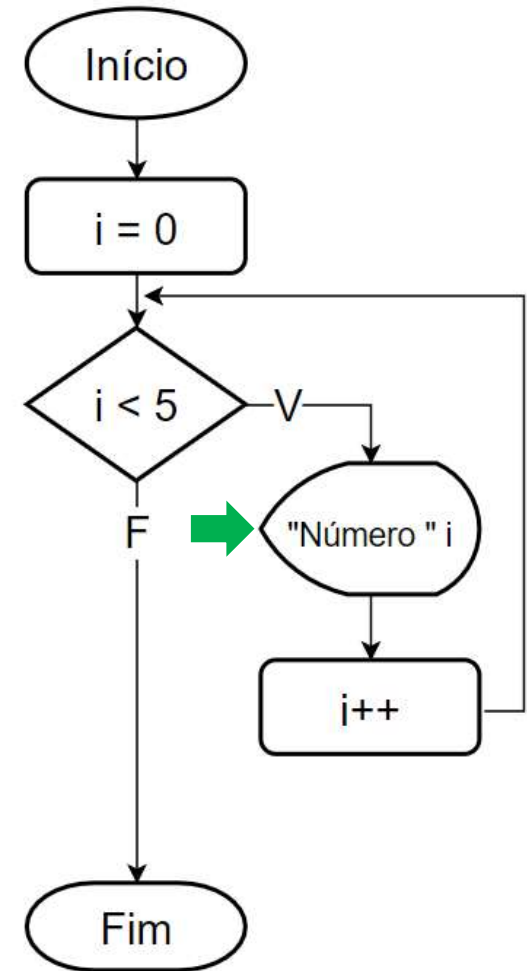
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

```
Número 0
Número 1
Número 2
Número 3
Número 4
```

i
4





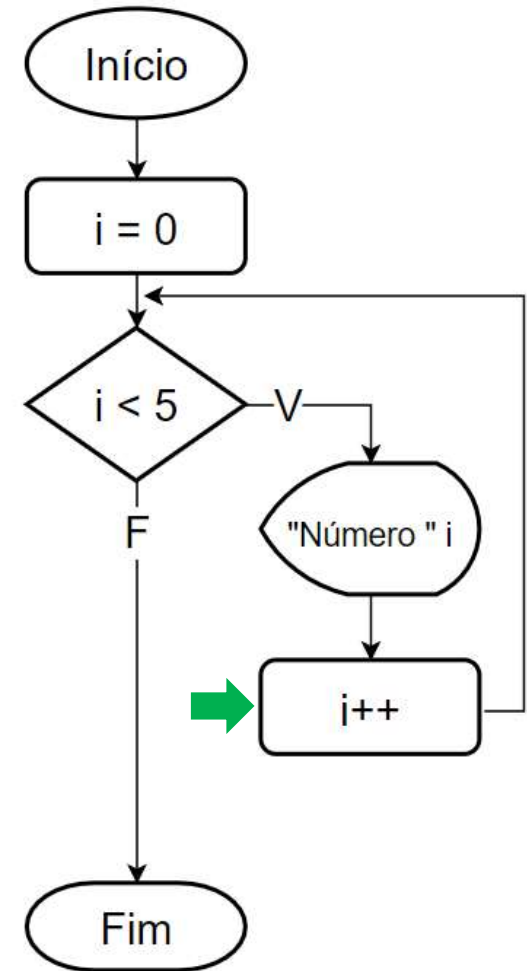
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

```
Número 0
Número 1
Número 2
Número 3
Número 4
```

i
5





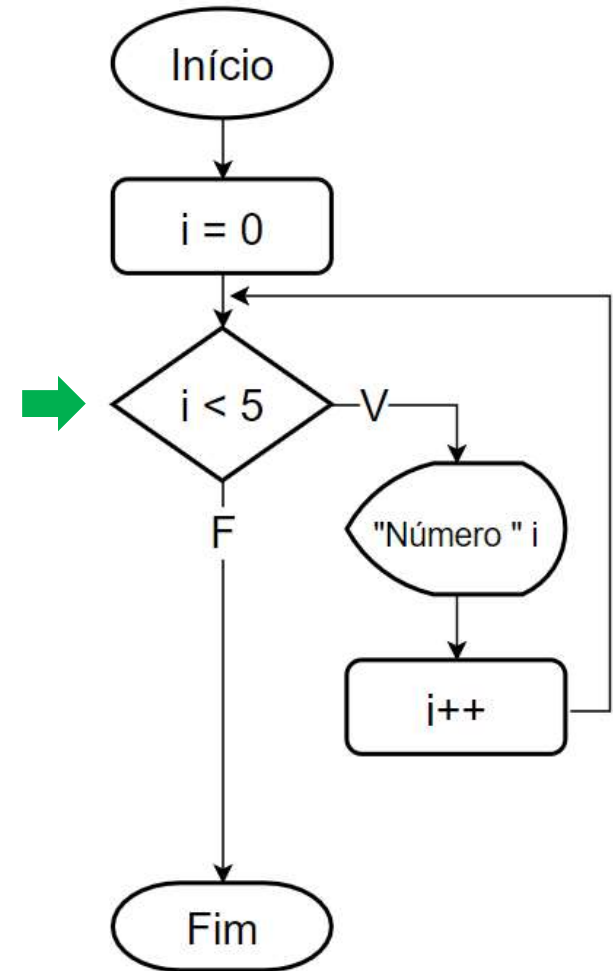
Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    → while (i<5)
    {
        Console.WriteLine ("Número " + i);
        i++;
    }
}
```

```
Numero 0
Numero 1
Numero 2
Numero 3
Numero 4
```

i
5






Exemplo - while

- Mostrar os números 0, 1, 2, 3 e 4.

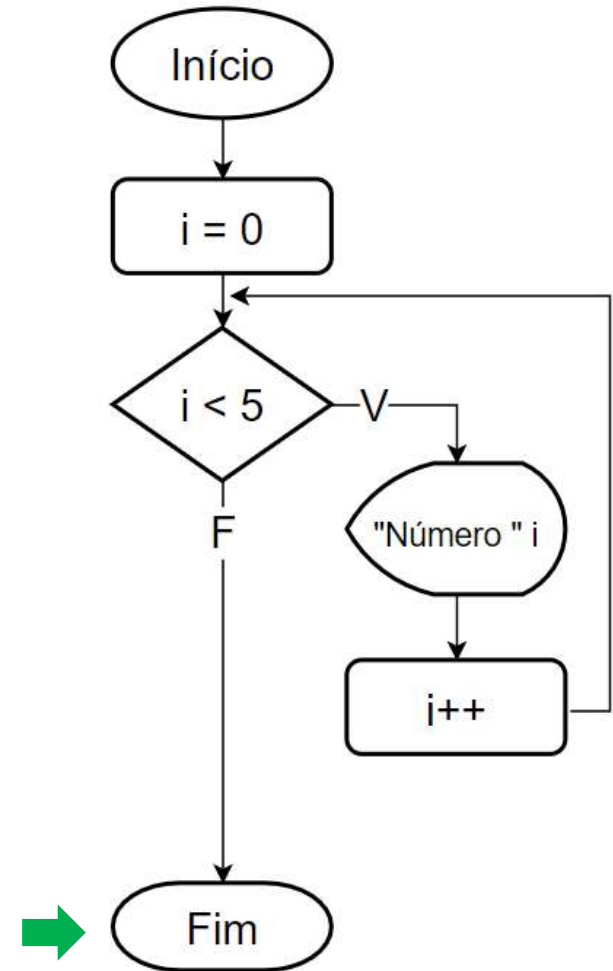
```
public static void Main (string[] args)
{
    int i;
    i=0;
    while (i<5)
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
}
```



```
Numero 0
Numero 1
Numero 2
Numero 3
Numero 4
```

i

5





Laço Condicional Pós-Teste

- Para a execução de laço condicional pós-teste temos na linguagem C# a instrução **do ... while**, que faz um teste lógico no final do laço, verificando se é possível executar novamente o trecho de instruções subordinadas a ele.
- Esse laço tem como característica operacional sempre executar o bloco subordinado ao laço no mínimo uma vez.
- A estrutura de laço **do ... while** tem seu funcionamento controlado por condição, mas esse tipo de laço, como dito, **executa o bloco de instruções subordinadas ao laço pelo menos uma vez antes de verificar a validade da condição estabelecida**, diferentemente do laço pré-teste while, que executa somente um conjunto de instruções quando a condição for favorável.



Laço Condicional Pós-Teste

- Dessa forma, do ... while funciona em sentido contrário ao while, **pois sempre processa um conjunto de instruções no mínimo uma vez**, mesmo que a condição não seja válida.
- A instrução **do ... while** pode ser escrita com a sintaxe:

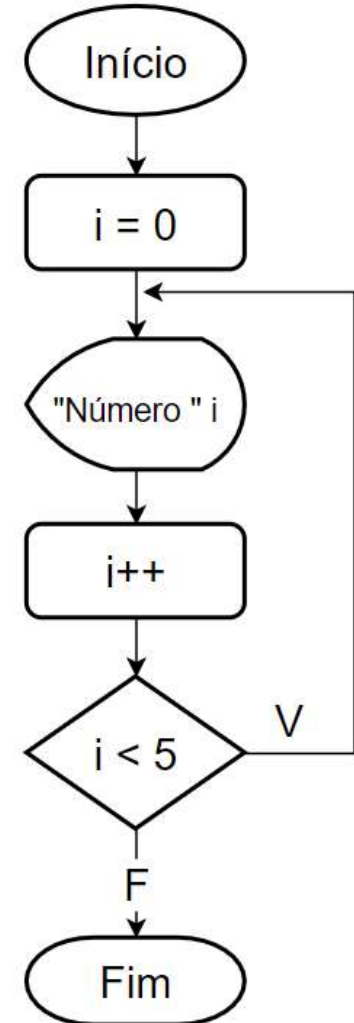
```
do
{
    <perita instrução 1 até condição ser verdadeira>;
    <perita instrução 2 até condição ser verdadeira>;
    <perita instrução 3 até condição ser verdadeira>;
    <perita instrução N até condição ser verdadeira>;
}
while <(condição)>;
```



Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

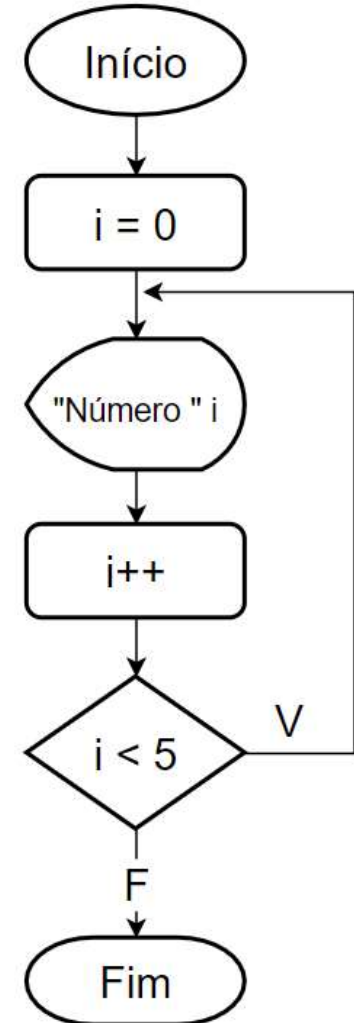
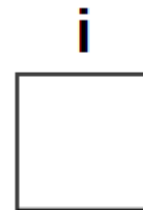




Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```





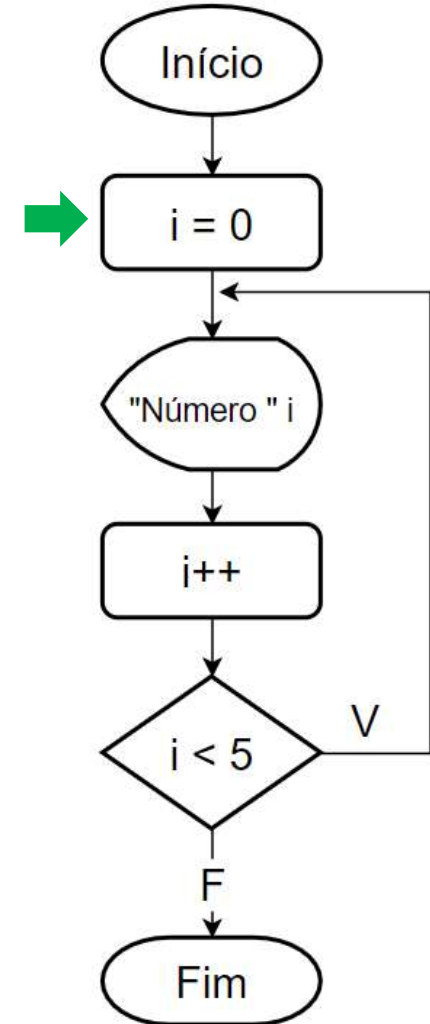
Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    → i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```



i
0





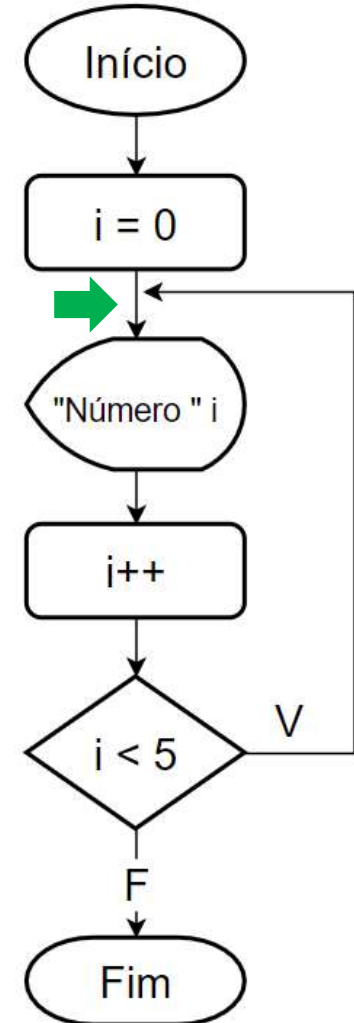
Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```



i
0

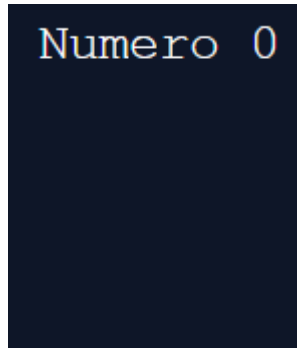




Exemplo – do ... while

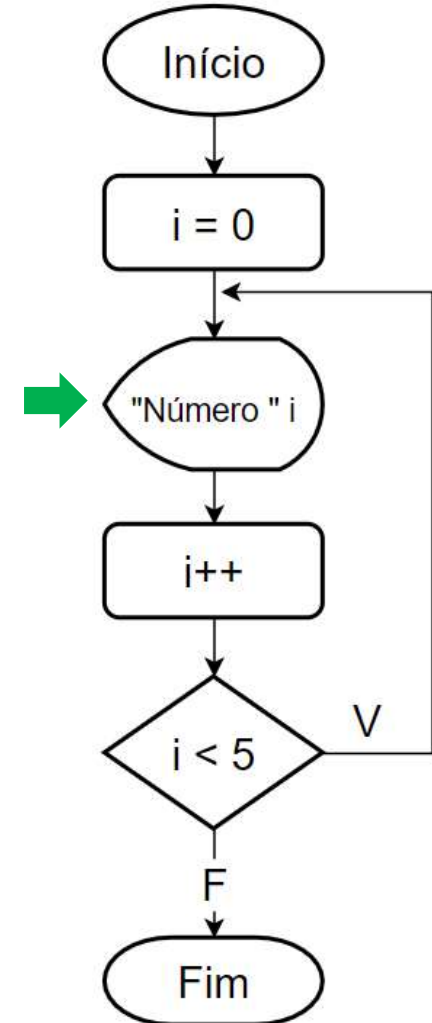
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```



Numero 0

i
0





Exemplo – do ... while

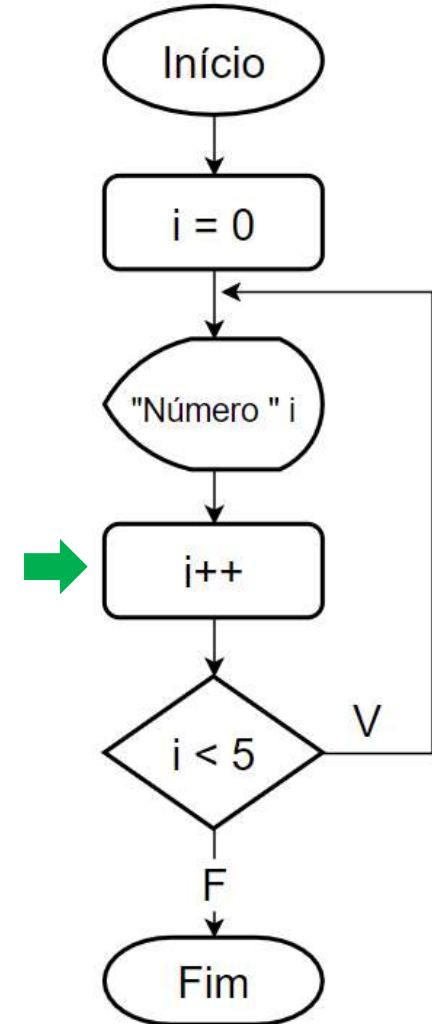
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

Numero 0

i

1





Exemplo – do ... while

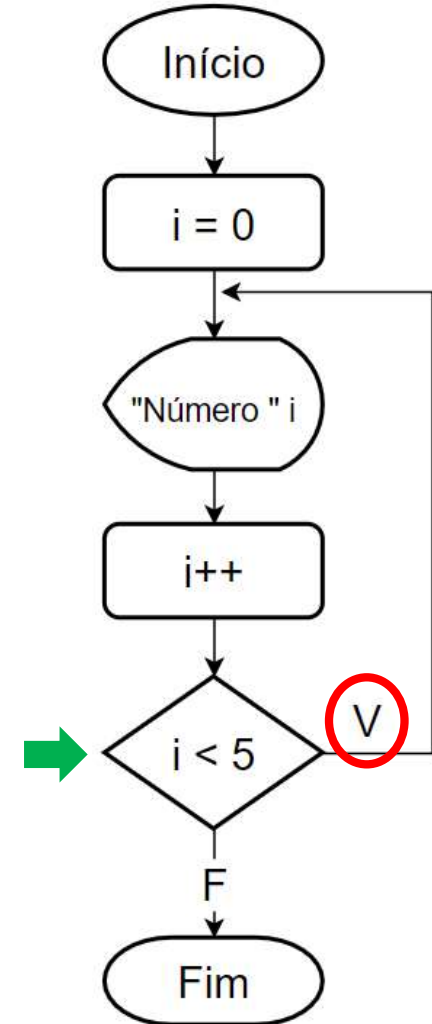
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

Numero 0

i

1





Exemplo – do ... while

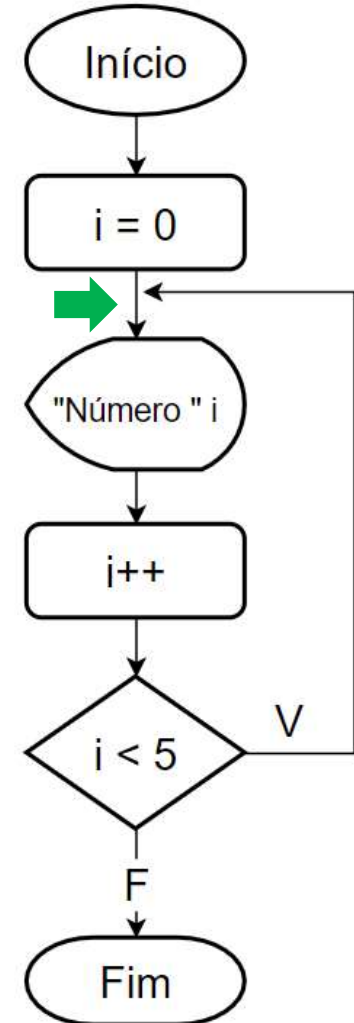
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

Numero 0

i

1





Exemplo – do ... while

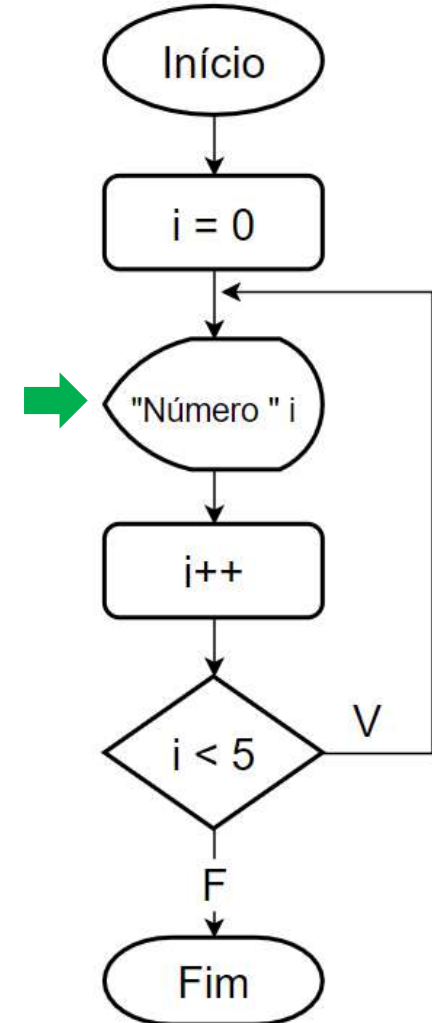
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
```

i

1





Exemplo – do ... while

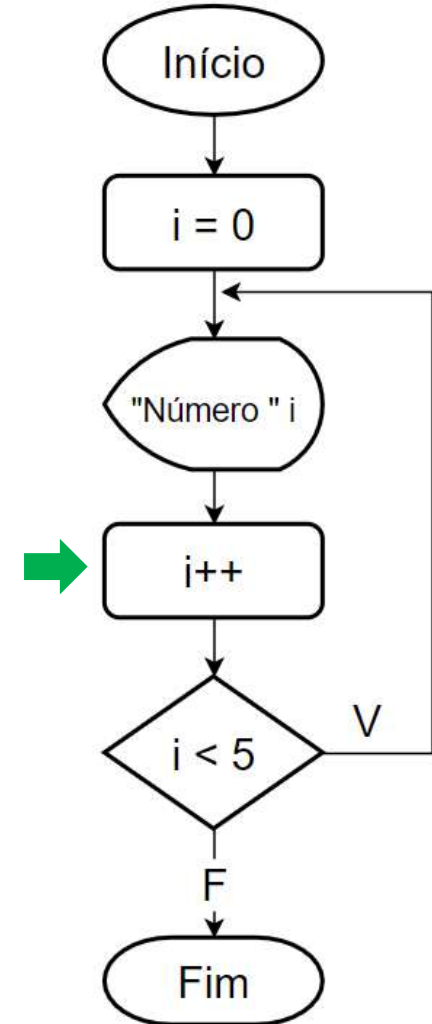
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
```

i

2





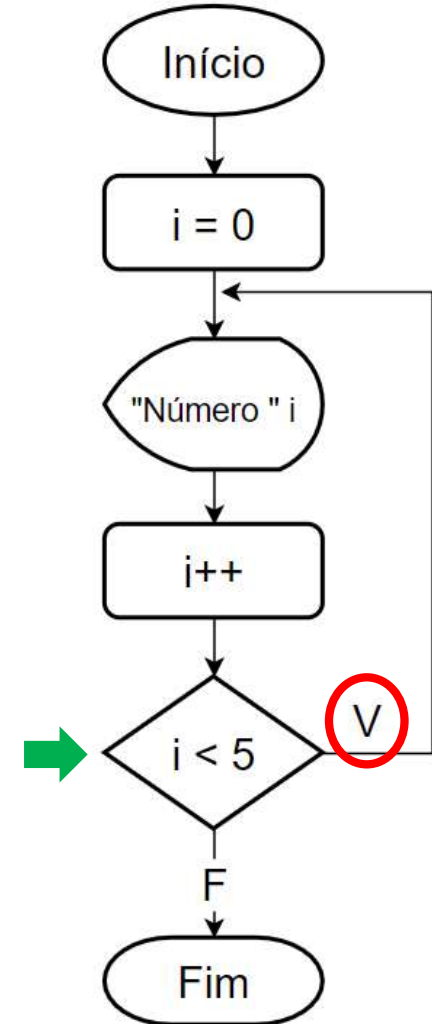
Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
```

i
2





Exemplo – do ... while

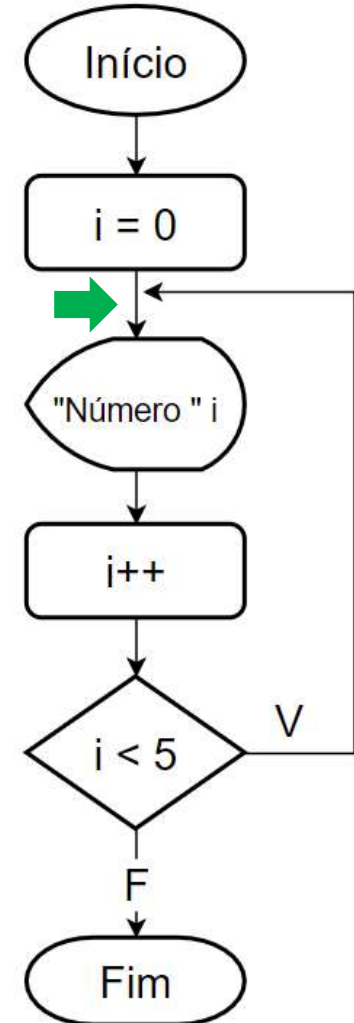
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
```

i

2





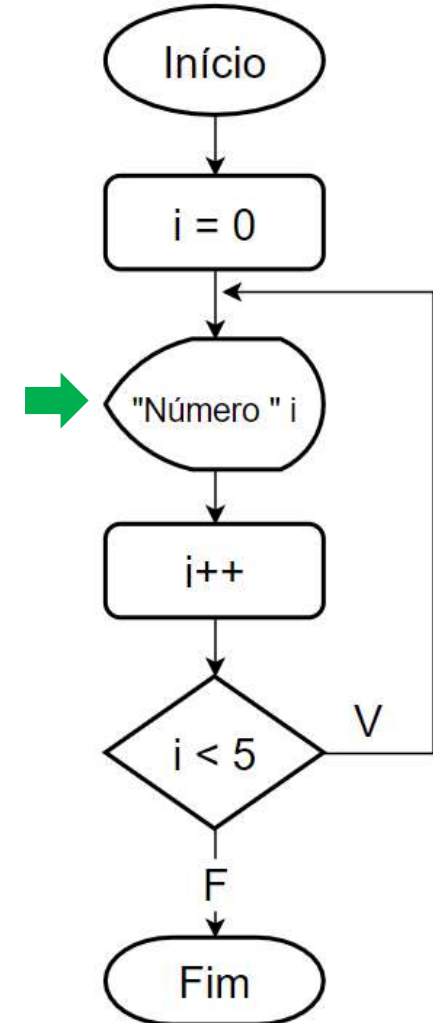
Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
Numero 2
```

i
2





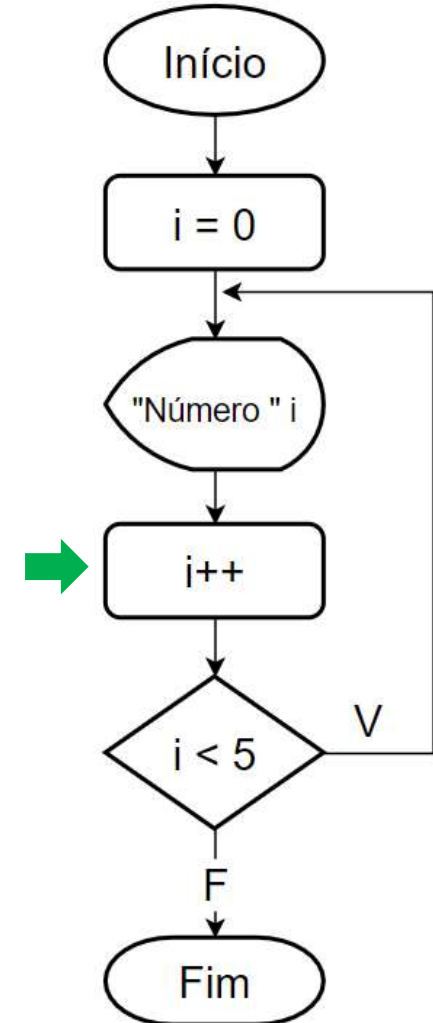
Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
Numero 2
```

i
3





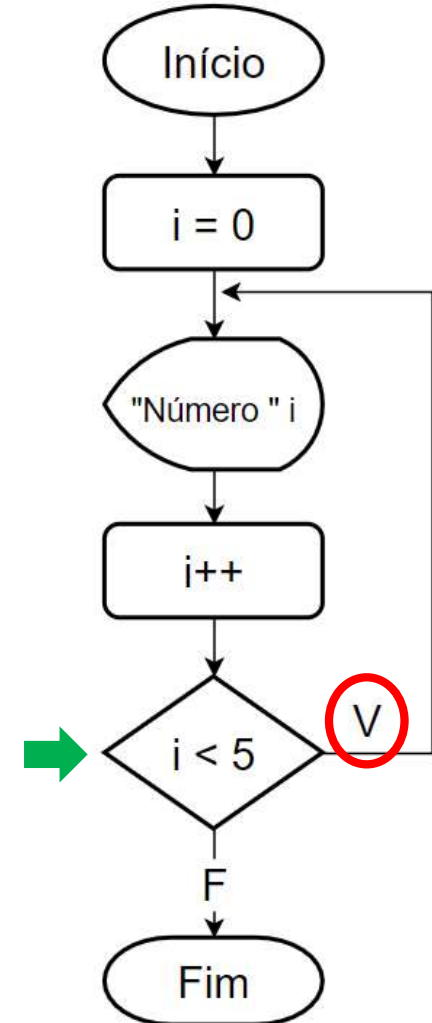
Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
Numero 2
```

i
3





Exemplo – do ... while

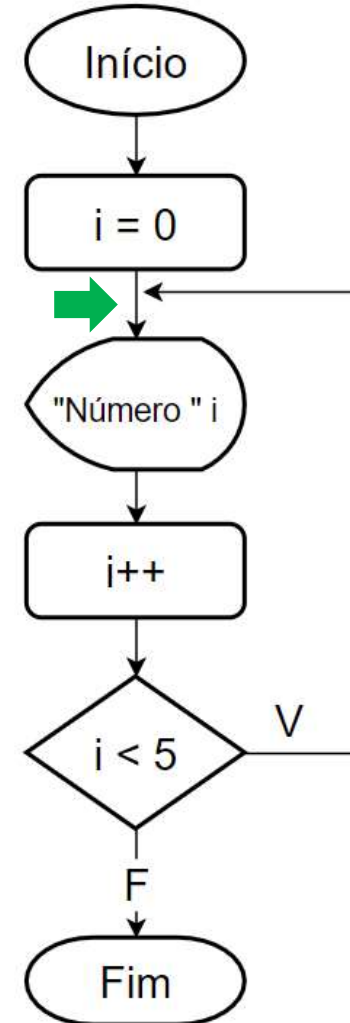
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
Numero 2
```

i

3





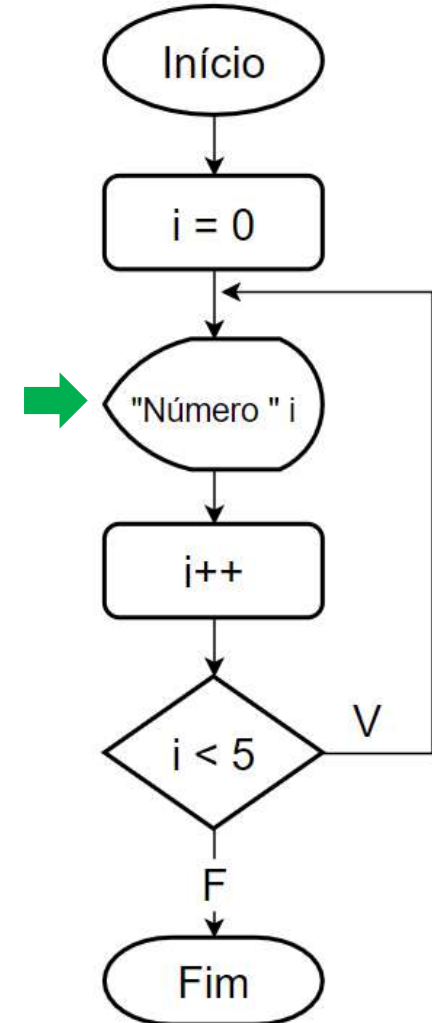
Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
Numero 2
Numero 3
```

i
3





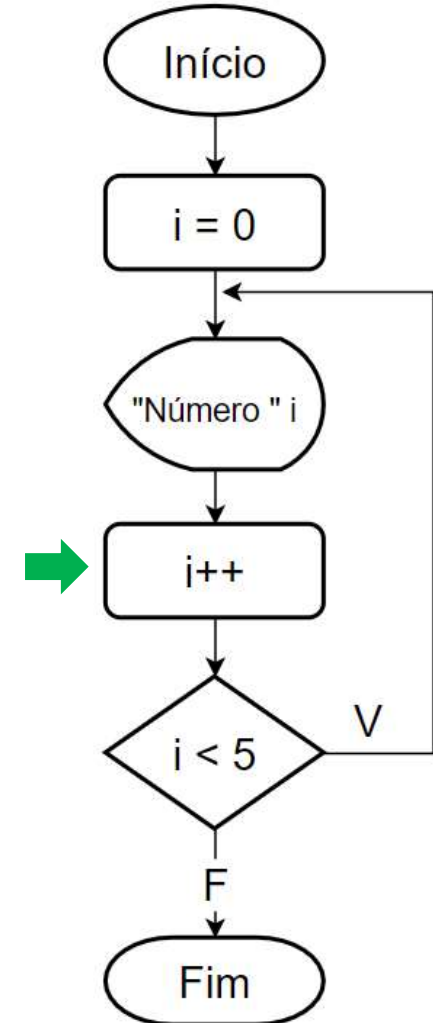
Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
Numero 2
Numero 3
```

i
4





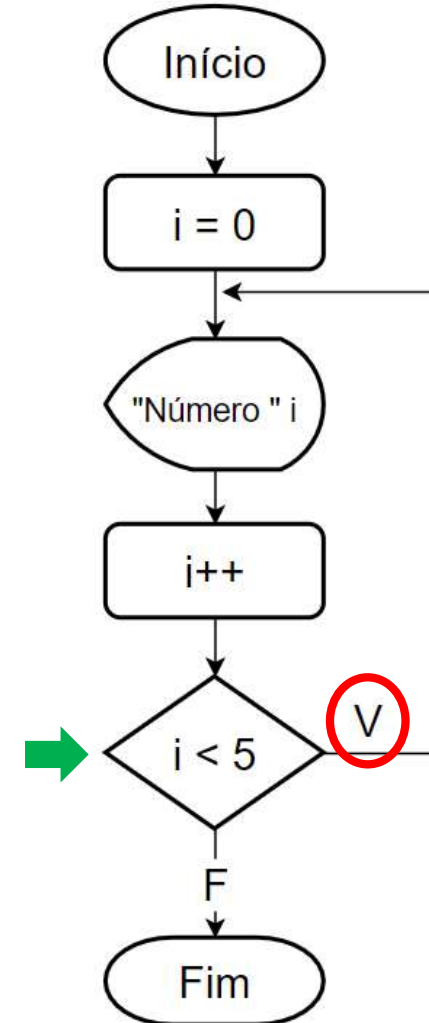
Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
Numero 2
Numero 3
```

i
4





Exemplo – do ... while

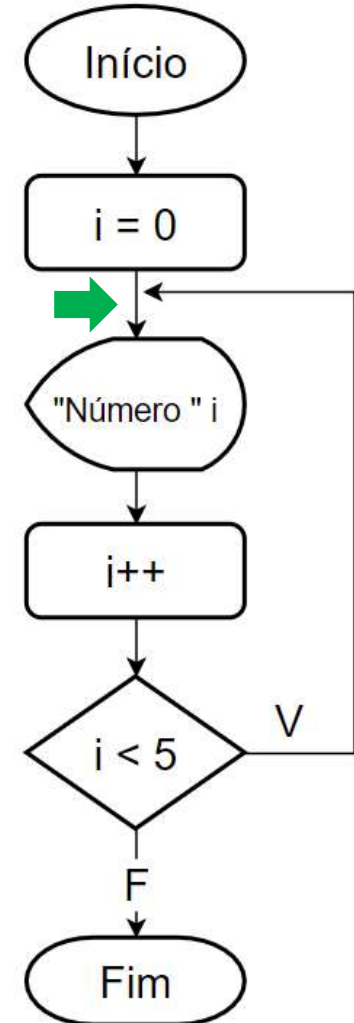
- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
Numero 2
Numero 3
```

i

4





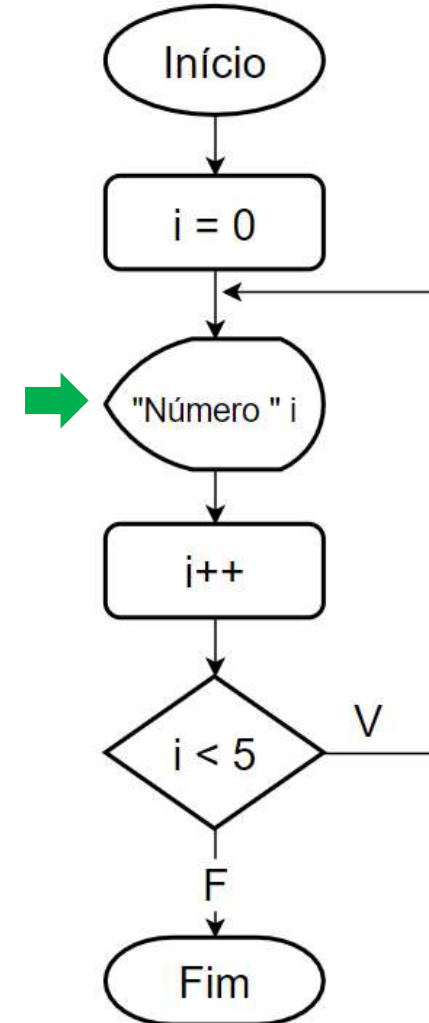
Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
Numero 2
Numero 3
Numero 4
```

i
4





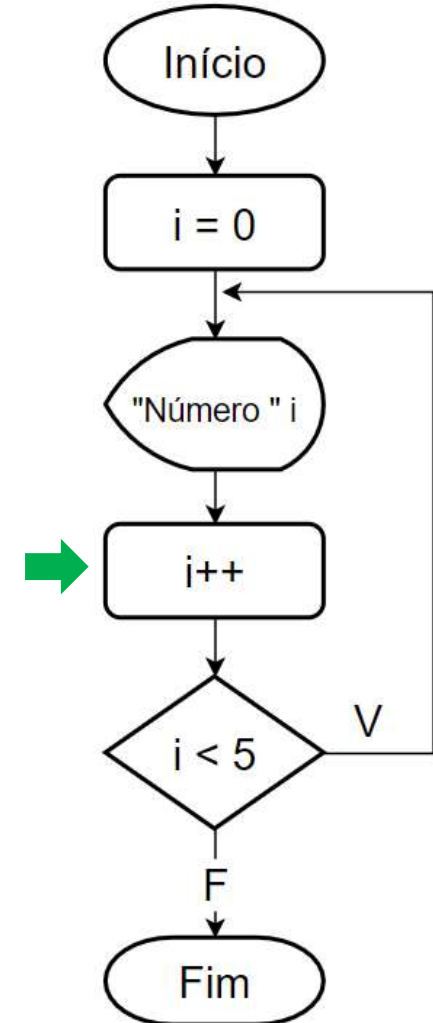
Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
Numero 2
Numero 3
Numero 4
```

i
5





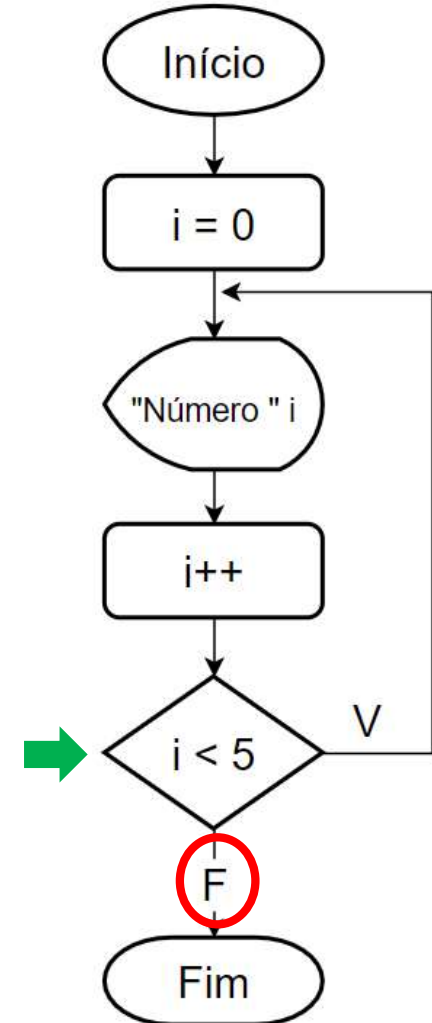
Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```

```
Numero 0
Numero 1
Numero 2
Numero 3
Numero 4
```

i
5






Exemplo – do ... while

- Mostrar os números 0, 1, 2, 3 e 4.

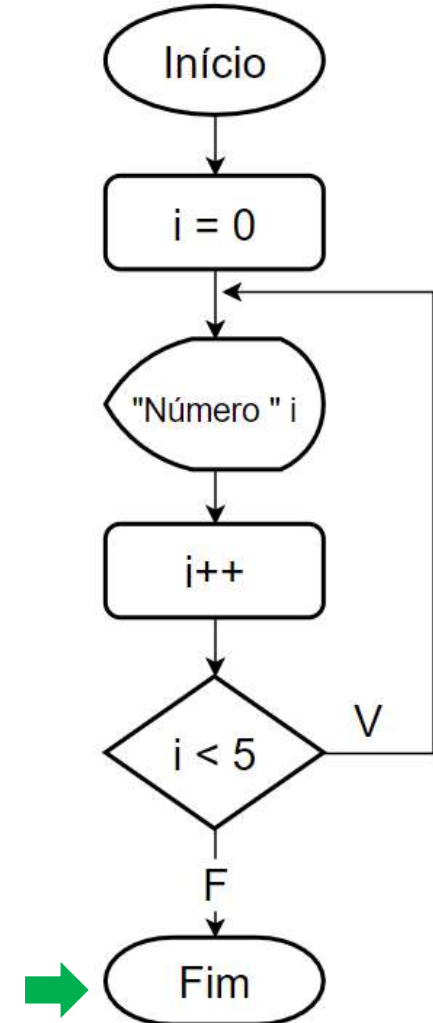
```
public static void Main (string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine ("Numero " + i);
        i++;
    }
    while (i<5);
}
```



```
Numero 0
Numero 1
Numero 2
Numero 3
Numero 4
```

i

5





Laço Incondicional

- Já vimos os conceitos de laços com **while** e **do ... while**.
- Nessas duas técnicas, vimos que é possível elaborar trechos que executem uma parte do programa um determinado número de vezes com a utilização de uma variável contador.
- Além das formas que já vimos, há outra forma de facilitar o uso de contadores por meio do laço **for**, que possui a sintaxe:

```
for (<início>; <fim>; <incremento>)
{
    <executa instrução 1>
    <executa instrução 2>
    <executa instrução 3>
    <executa instrução N>
}
```



Laço Incondicional

- O laço **for** é executado por uma instrução que recebe três parâmetros de trabalho, separados por ponto e vírgula.
- O primeiro parâmetro é representado por uma variável e seu valor inicial;
- O segundo parâmetro, pela condição de finalização da execução do laço;
- O terceiro e último parâmetro, pelo contador de incremento que ocorre entre os valores inicial e final da contagem.

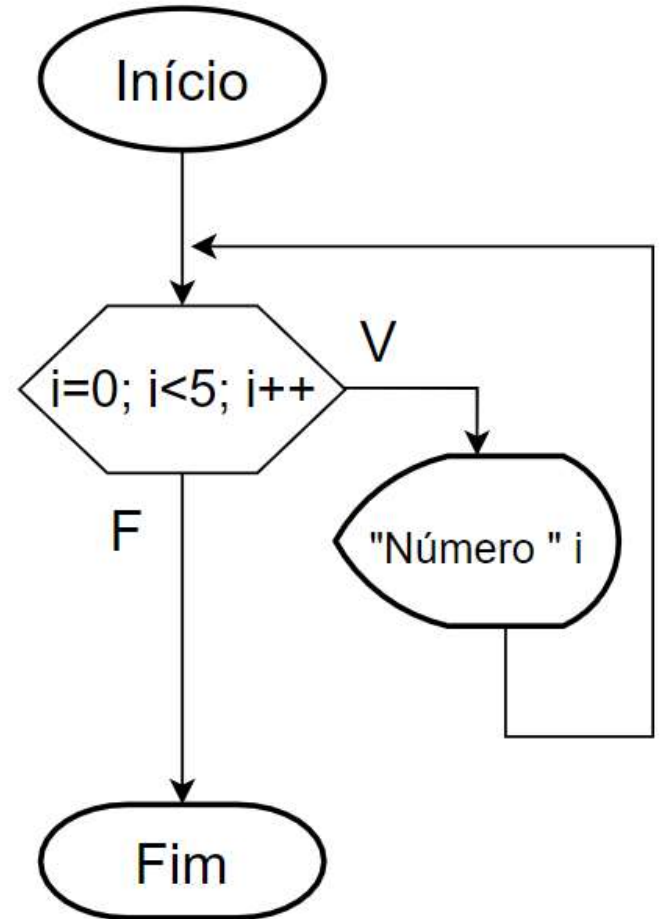
```
for (<início>; <fim>; <incremento>)  
{  
    <executa instrução 1>  
    <executa instrução 2>  
    <executa instrução 3>  
    <executa instrução N>  
}
```



Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    for (i=0;i<5;i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```

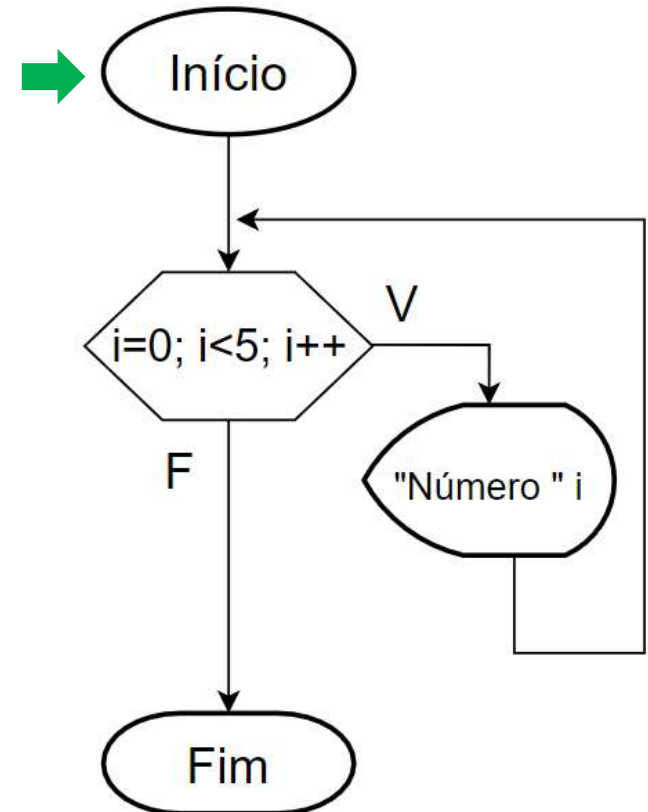




Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
➔ public static void Main (string[] args)
{
    int i;
    for (i=0;i<5;i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```

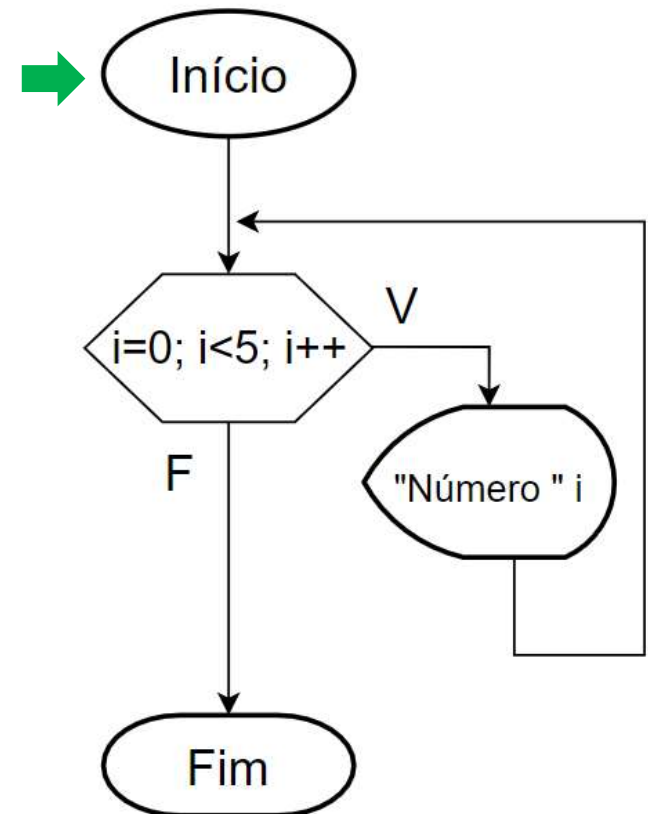




Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    for (i=0;i<5;i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```





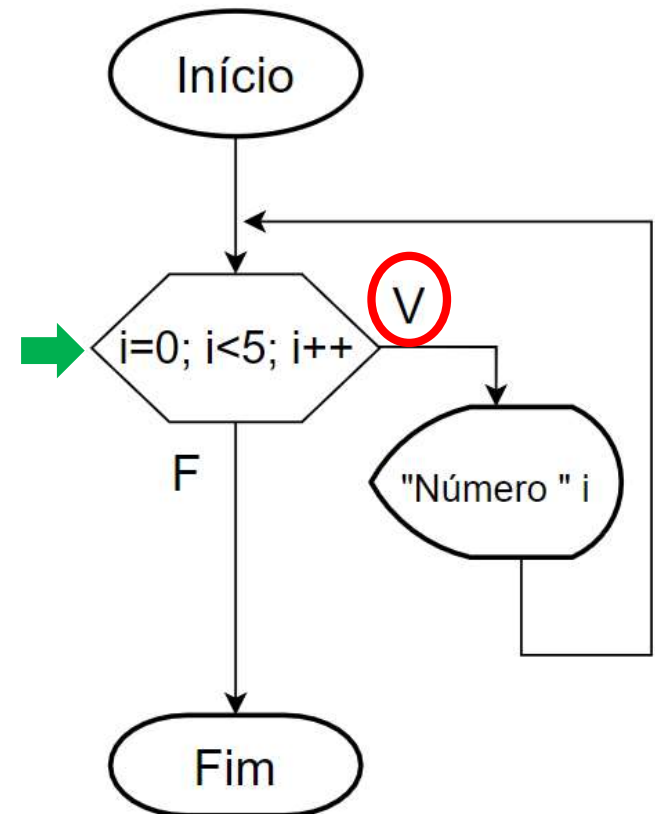
Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    → for (i=0; i<5; i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```



i
0

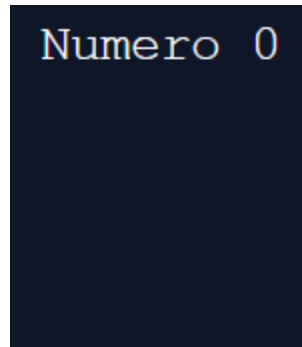




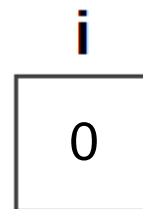
Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

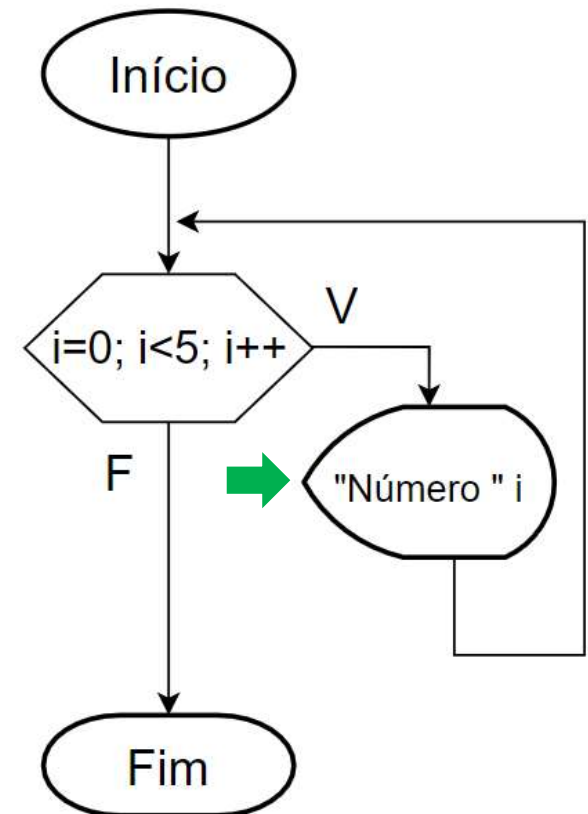
```
public static void Main (string[] args)
{
    int i;
    for (i=0;i<5;i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```



Numero 0



i
0





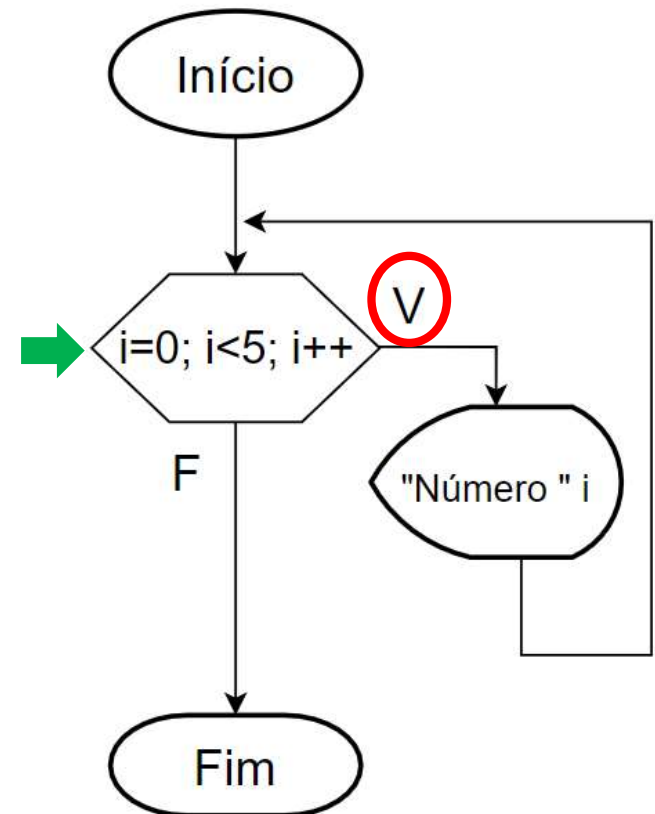
Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    → for (i=0; i<5; i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```

```
Numero 0
```

i
1





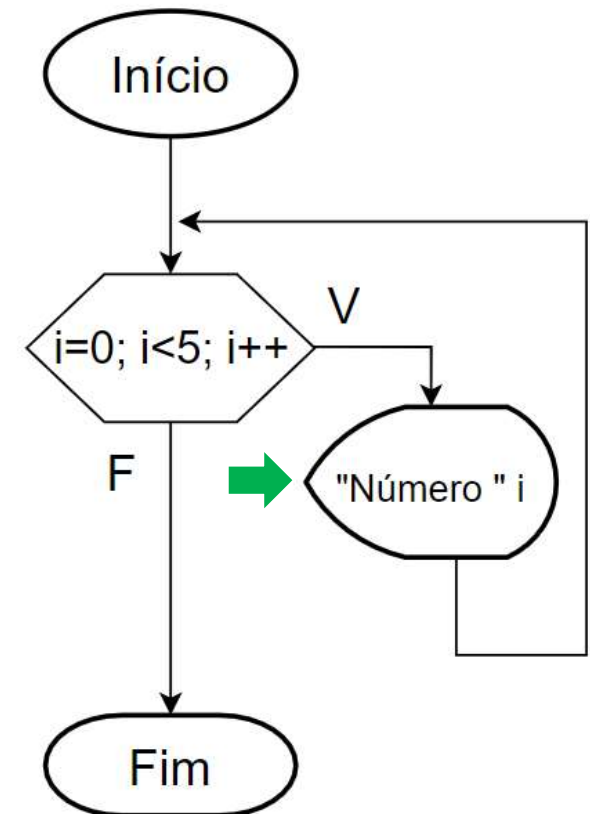
Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    for (i=0;i<5;i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```

```
Numero 0
Numero 1
```

i
1





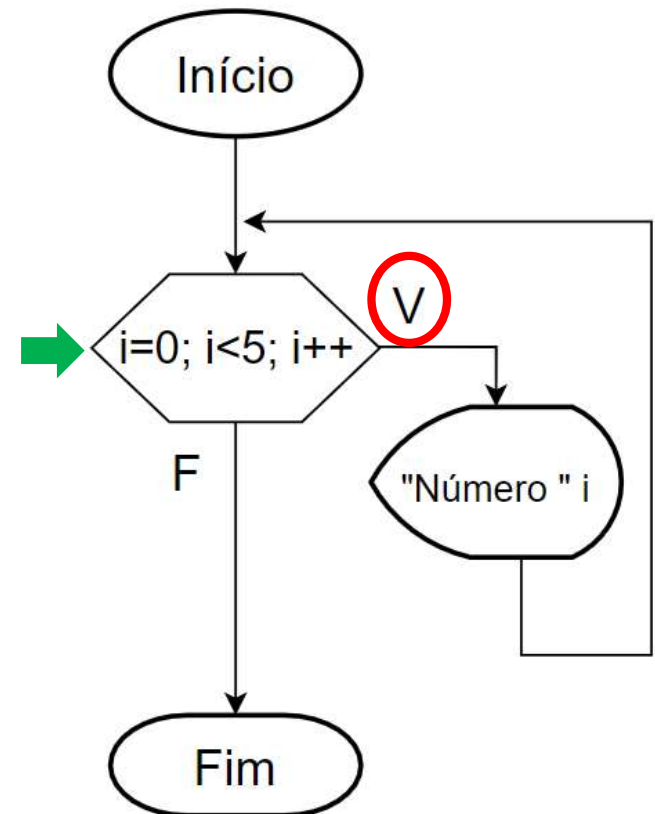
Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    → for (i=0; i<5; i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```

```
Numero 0
Numero 1
```

i
2





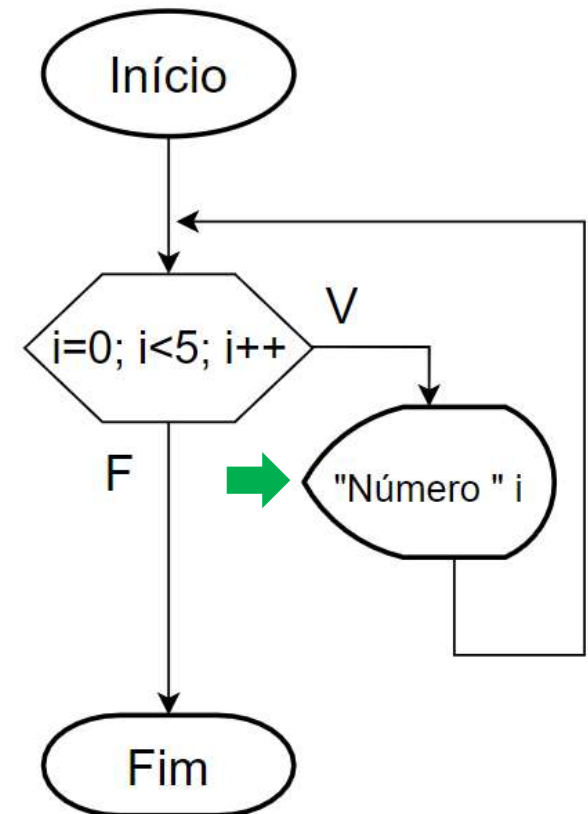
Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    for (i=0;i<5;i++)
    {
        ➡ Console.WriteLine ("Número " + i);
    }
}
```

```
Numero 0
Numero 1
Numero 2
```

i
2





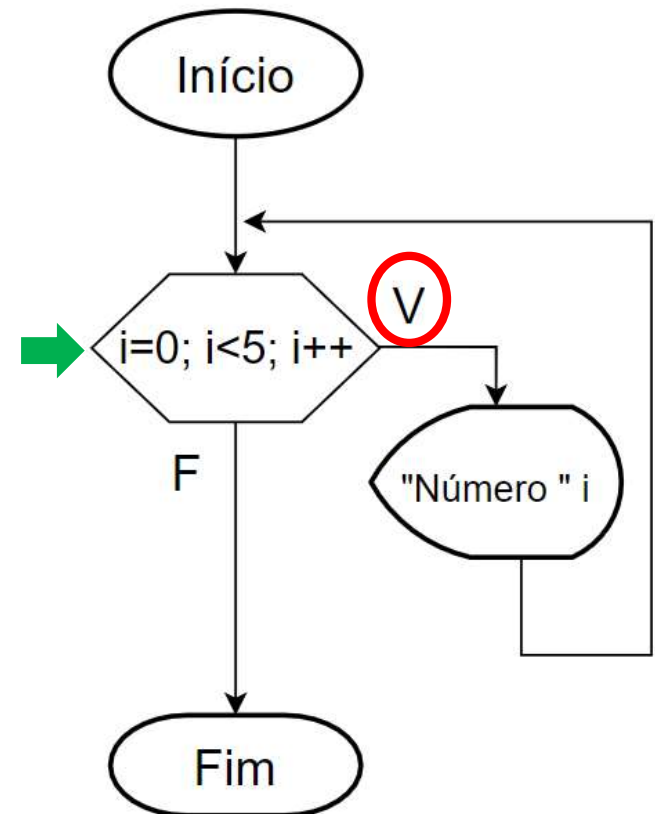
Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    → for (i=0; i<5; i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```

```
Numero 0
Numero 1
Numero 2
```

i
3





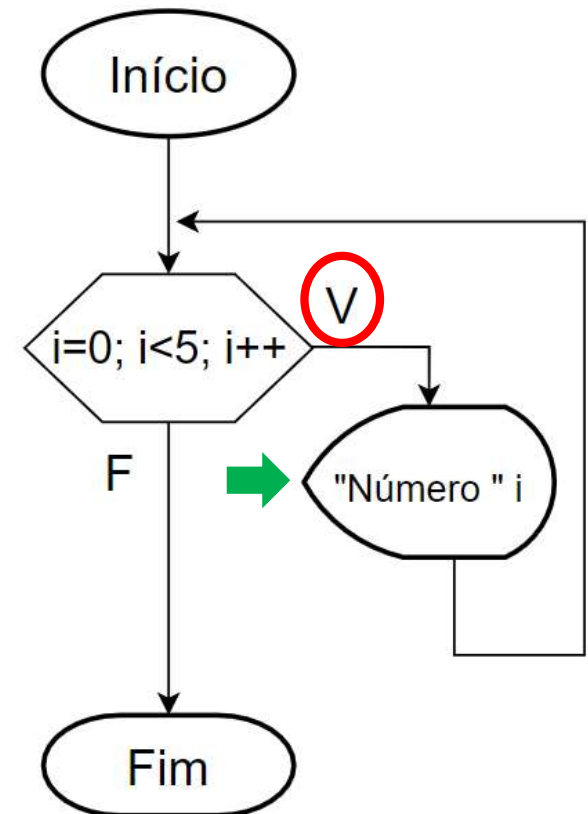
Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    for (i=0;i<5;i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```

```
Numero 0
Numero 1
Numero 2
Numero 3
```

i
3





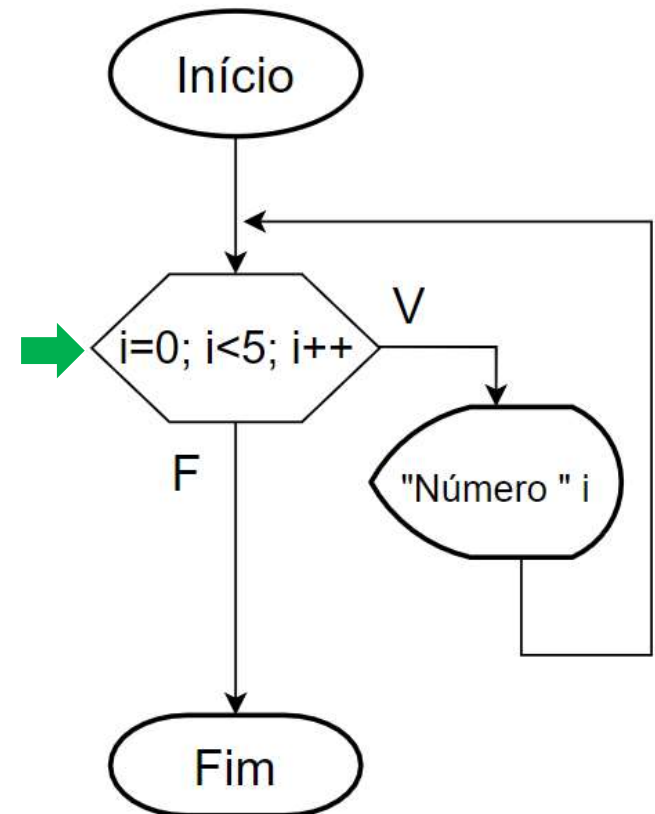
Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    → for (i=0; i<5; i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```

```
Numero 0
Numero 1
Numero 2
Numero 3
```

i
4



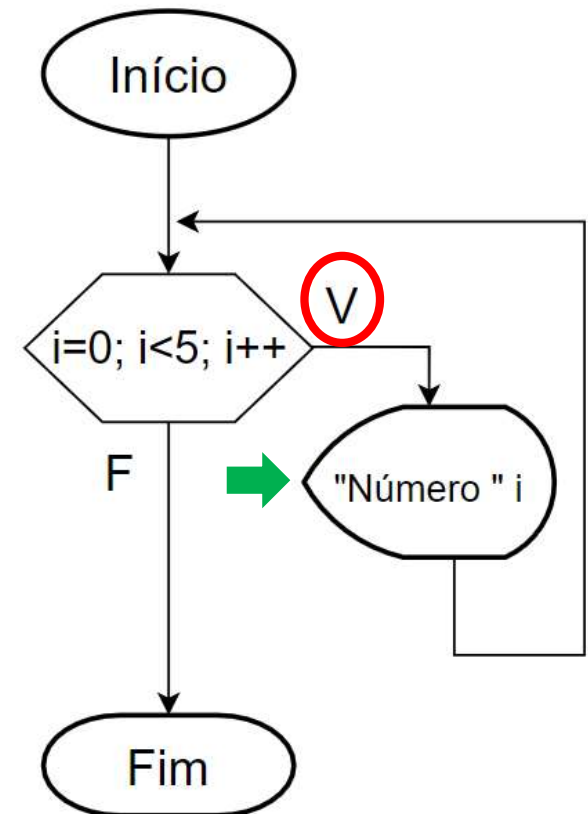


Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    for (i=0;i<5;i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```

```
Numero 0
Numero 1
Numero 2
Numero 3
Numero 4
```





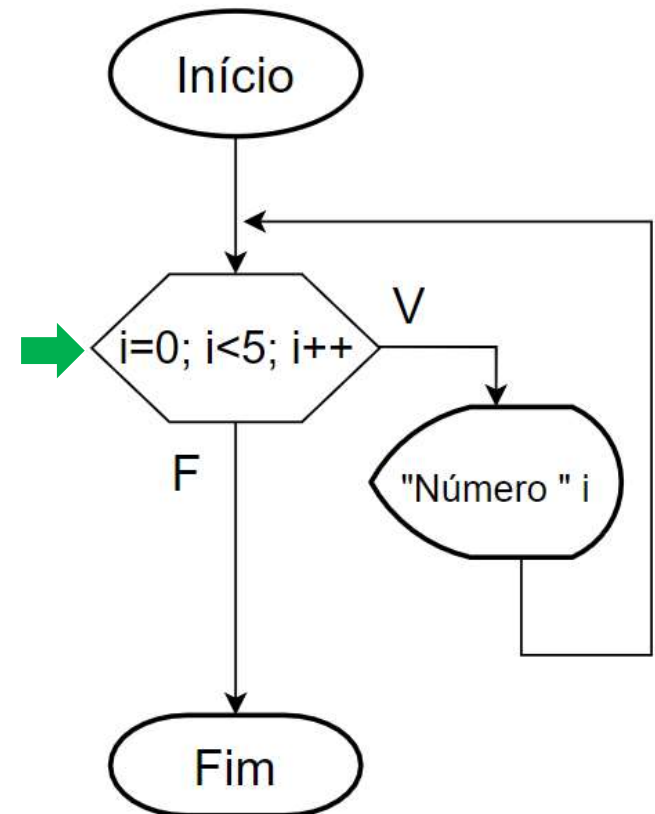
Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

```
public static void Main (string[] args)
{
    int i;
    → for (i=0; i<5; i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```

```
Numero 0
Numero 1
Numero 2
Numero 3
Numero 4
```

i
5






Exemplo - for

- Mostrar os números 0, 1, 2, 3 e 4.

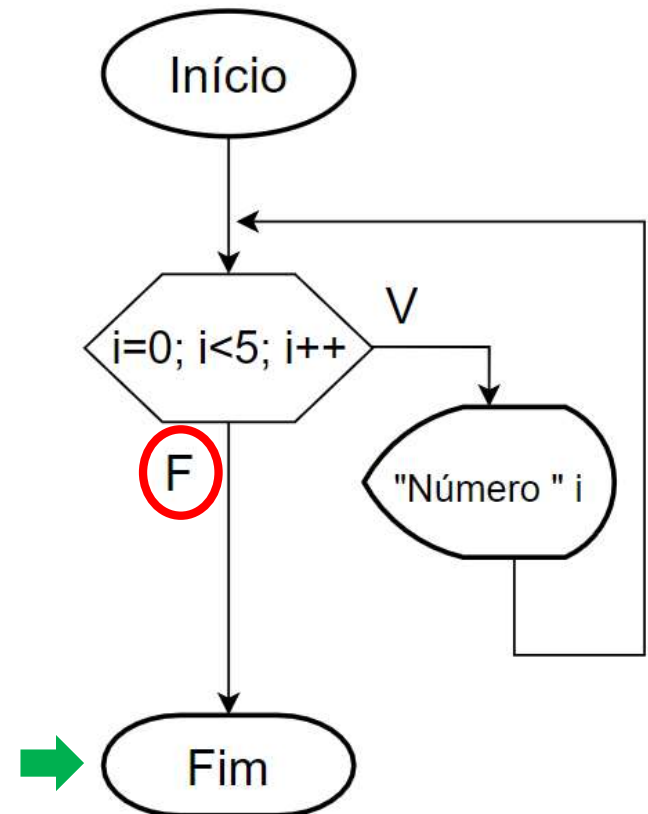
```
public static void Main (string[] args)
{
    int i;
    for (i=0;i<5;i++)
    {
        Console.WriteLine ("Número " + i);
    }
}
```



```
Numero 0
Numero 1
Numero 2
Numero 3
Numero 4
```

i

5





Exercícios

1. Desenvolver a codificação dos seguintes problemas a serem transformados em programas de computador. Utilize o laço **while**.
 - a) Apresentar todos os valores numéricos inteiros impares situados na faixa de 0 a 20.
 - b) Apresentar o total da soma dos cem primeiros números inteiros, representados pela sequencia $1+2+3+4+5+6+7+\dots+97+98+99+100$.
 - c) Apresentar todos os números divisíveis por 4 que sejam menores que 30. Iniciar a contagem com o valor 1 (um).
 - d) Apresentar os quadrados dos números inteiros de 15 a 200.
 - e) Apresentar o resultado da soma de todos os valores pares existentes na faixa numérica de 1 a 50.
2. Desenvolver a codificação dos problemas apresentados no exercício 1 com o laço **do...while**.
3. Desenvolver a codificação dos problemas apresentados no exercício 1 com o laço **for**

Conceitos desta aula

- - laços para repetição;
- - laço condicional pré-teste;
- - laço condicional pós-teste;
- - laço incondicional.