



# Fundamentos de Programação

Estruturas de Controle Repetição

# Repetição

- Chamadas de estruturas iterativas, iterações, laços ou loops;
- Permitem repetir a execução de uma ação várias vezes;
- Podem ser:
  1. Repetição com Teste no Início
  2. Repetição com Teste no Fim
  3. Repetição com Variável de Controle

# Repetição com Teste no Início

```
while ( condicao )  
{  
    BlocoDeComandos1;  
}  
BlocoDeComandos2;
```

- Enquanto a condição for verdadeira, a seqüência de comandos será repetida.
- Quando a condição fornecer resultado falso, o controle sai da estrutura passando para o comando seguinte ao final do bloco.

# Repetição com Teste no Início

## Exemplo 1: Uso de FLAG

- **FLAG** é um valor específico fornecido após o último dado de entrada, que serve para indicar o fim dos dados de entrada.
- **FLAG** é somente uma marca de fim dos dados de entrada (não é um dado de entrada) e não pode ser processado.
- A leitura do **FLAG** informa ao programa que os dados de entrada terminaram e que ele deve partir para a execução da finalização de seu processamento (cálculos finais, impressões finais, etc.).

# Repetição com Teste no Início

## Exemplo 1: Uso de FLAG

```
...  
Lê o primeiro valor de entrada  
Enquanto o valor corrente não for FLAG  
{  
    Processa os dados  
    Lê o próximo valor de entrada  
}  
Finaliza o processamento  
...
```

# Repetição com Teste no Início

## Exemplo 1: Uso de FLAG

Desenvolva um algoritmo que leia uma sequência de números inteiros, calculando e imprimindo o quadrado de cada número lido. A sequência deve terminar quando o número 0 (zero) for lido.

# Repetição com Teste no Início

## Exemplo 1: Uso de FLAG

Desenvolva um algoritmo que leia uma sequência de números inteiros, calculando e imprimindo o quadrado de cada número lido. A sequência deve terminar quando o número 0 (zero) for lido.

***ZERO*** → indica o final da sequência de valores.  
→ **FLAG**

# Repetição com Teste no Início

## Exemplo 1: Uso de FLAG

Desenvolva um algoritmo que leia uma sequência de números inteiros, calculando e imprimindo o quadrado de cada número lido. A sequência deve terminar quando o número 0 (zero) for lido.

```
Lê o primeiro número
```

```
Enquanto o número lido não for zero
```

```
{
```

```
    Calcula e imprime o quadrado do número
```

```
    Lê o próximo número
```

```
}
```



# Repetição com Teste no Início

## Exemplo 1: Uso de FLAG

```
static void Main(string[] args)
{
    int num, quad;
    // imprime uma msg e le o 1o. inteiro
    Console.Write("Digite um numero inteiro: ");
    num = Convert.ToInt32(Console.ReadLine());

    while( num != 0 )
    {
        quad = num * num;
        Console.WriteLine("Quadrado de {0}: {1}", num, quad);
        Console.Write("Digite um numero inteiro: ");
        num = Convert.ToInt32(Console.ReadLine());
    }
}
```

# Exemplo 1: Uso de FLAG

## TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.Write("Digite um numero inteiro: ");
8 num = Convert.ToInt32(Console.ReadLine()); 9
10 while( num != 0 )
11 {
12     quad = num * num;
13     Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15     Console.Write("Digite um numero inteiro: ");
16 num = Convert.ToInt32(Console.ReadLine()); 17
18 }
```

linha	num	quad	teste
4	?	?	---

# Exemplo 1: Uso de FLAG

## TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.Write("Digite um numero inteiro: ");
8 num = Convert.ToInt32(Console.ReadLine()); 9
10 while( num != 0 )
11 {
12     quad = num * num;
13     Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15     Console.Write("Digite um numero inteiro: ");
16 num = Convert.ToInt32(Console.ReadLine()); 17
17 }
18 }
```

linha	num	quad	teste
4	?	?	---
8	5	?	---

```
Digite um numero inteiro: 5
```

# Exemplo 1: Uso de FLAG

## TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.Write("Digite um numero inteiro: ");
8 num = Convert.ToInt32(Console.ReadLine()); 9
10 while( num != 0 )
11 {
12     quad = num * num;
13     Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15     Console.Write("Digite um numero inteiro: ");
16 num = Convert.ToInt32(Console.ReadLine()); 17
17 }
18 }
```

linha	num	quad	teste
4	?	?	---
8	5	?	---
10	5	?	V

```
Digite um numero inteiro: 5
```

# Exemplo 1: Uso de FLAG

## TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.Write("Digite um numero inteiro: ");
8 num = Convert.ToInt32(Console.ReadLine()); 9
10 while( num != 0 )
11 {
12     quad = num * num;
13     Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15     Console.Write("Digite um numero inteiro: ");
16 num = Convert.ToInt32(Console.ReadLine()); 17
17 }
18 }
```

linha	num	quad	teste
4	?	?	---
8	5	?	---
10	5	?	V
12	5	25	---

```
Digite um numero inteiro: 5
```

# Exemplo 1: Uso de FLAG

## TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.Write("Digite um numero inteiro: ");
8 num = Convert.ToInt32(Console.ReadLine()); 9
10 while( num != 0 )
11 {
12     quad = num * num;
13     Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15     Console.Write("Digite um numero inteiro: ");
16     num = Convert.ToInt32(Console.ReadLine());
17 }
18 }
```

linha	num	quad	teste
4	?	?	---
8	5	?	---
10	5	?	V
12	5	25	---
16	2	25	---

```
Digite um numero inteiro: 5
Quadrado de 5: 25
Digite um numero inteiro: 2
```

# Exemplo 1: Uso de FLAG

## TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.Write("Digite um numero inteiro: ");
8 num = Convert.ToInt32(Console.ReadLine()); 9
10 while( num != 0 )
11 {
12     quad = num * num;
13     Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15     Console.Write("Digite um numero inteiro: ");
16 num = Convert.ToInt32(Console.ReadLine()); 17
17 }
18 }
```

linha	num	quad	teste
4	?	?	---
8	5	?	---
10	5	?	V
12	5	25	---
16	2	25	---
10	2	25	V

```
Digite um numero inteiro: 5
Quadrado de 5: 25
Digite um numero inteiro: 2
```

# Exemplo 1: Uso de FLAG

## TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.Write("Digite um numero inteiro: ");
8 num = Convert.ToInt32(Console.ReadLine()); 9
10 while( num != 0 )
11 {
12     quad = num * num;
13     Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15     Console.Write("Digite um numero inteiro: ");
16 num = Convert.ToInt32(Console.ReadLine()); 17
17 }
18 }
```

linha	num	quad	teste
4	?	?	---
8	5	?	---
10	5	?	V
12	5	25	---
16	2	25	---
10	2	25	V
12	2	4	---

```
Digite um numero inteiro: 5
Quadrado de 5: 25
Digite um numero inteiro: 2
```



# Exemplo 1: Uso de FLAG

## TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.Write("Digite um numero inteiro: ");
8 num = Convert.ToInt32(Console.ReadLine()); 9
10 while( num != 0 )
11 {
12     quad = num * num;
13     Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15     Console.Write("Digite um numero inteiro: ");
16 num = Convert.ToInt32(Console.ReadLine());
17 }
18 }
```

linha	num	quad	teste
4	?	?	---
8	5	?	---
10	5	?	V
12	5	25	---
16	2	25	---
10	2	25	V
12	2	4	---
16	0	4	---

```
Digite um numero inteiro: 5
Quadrado de 5: 25
Digite um numero inteiro: 2
Quadrado de 2: 4
Digite um numero inteiro: 0
```

# Exemplo 1: Uso de FLAG

## TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.Write("Digite um numero inteiro: ");
8 num = Convert.ToInt32(Console.ReadLine()); 9
10 while( num != 0 )
11 {
12     quad = num * num;
13     Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15     Console.Write("Digite um numero inteiro: ");
16 num = Convert.ToInt32(Console.ReadLine()); 17
17 }
18 }
```

linha	num	quad	teste
4	?	?	---
8	5	?	---
10	5	?	V
12	5	25	---
16	2	25	---
10	2	25	V
12	2	4	---
16	0	4	---
10	0	4	F

```
Digite um numero inteiro: 5
Quadrado de 5: 25
Digite um numero inteiro: 2
Quadrado de 2: 4
Digite um numero inteiro: 0
```

# Exemplo 1: Uso de FLAG

## TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.Write("Digite um numero inteiro: ");
8 num = Convert.ToInt32(Console.ReadLine()); 9
10 while( num != 0 )
11 {
12     quad = num * num;
13     Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15     Console.Write("Digite um numero inteiro: ");
16 num = Convert.ToInt32(Console.ReadLine()); 17
17 }
18 }
```

linha	num	quad	teste
4	?	?	---
8	5	?	---
10	5	?	V
12	5	25	---
16	2	25	---
10	2	25	V
12	2	4	---
16	0	4	---
10	0	4	F
18	0	4	---

```
Digite um numero inteiro: 5
Quadrado de 5: 25
Digite um numero inteiro: 2
Quadrado de 2: 4
Digite um numero inteiro: 0
```

# Repetição com Teste no Início

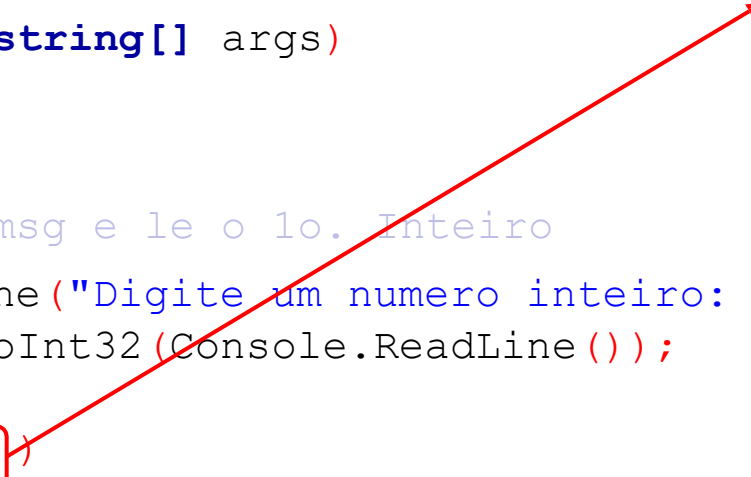
## Quesitos importantes em repetições

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.WriteLine("Digite um numero inteiro: ");
8     num = Convert.ToInt32(Console.ReadLine());
9
10    while( num != 0 )
11    {
12        quad = num * num;
13        Console.WriteLine("Quadrado de {0}: {1}", num, quad); 14
15        Console.WriteLine("Digite um numero inteiro: ");
16        num = Convert.ToInt32(Console.ReadLine());
17    }
18 }
```

# Repetição com Teste no Início

## Quesitos importantes em repetições

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.WriteLine("Digite um numero inteiro: ");
8     num = Convert.ToInt32(Console.ReadLine());
9
10    while (num != 0)
11    {
12        quad = num * num;
13        Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15        Console.WriteLine("Digite um numero inteiro: ");
16        num = Convert.ToInt32(Console.ReadLine());
17    }
18 }
```



### Condição

Teste normal envolve ao menos uma variável.

Imprime  
menos uma

# Repetição com Teste no Início

## Quesitos importantes em repetições

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.WriteLine("Digite um numero inteiro: ");
8 num = Convert.ToInt32(Console.ReadLine()); 9
9
10    while( num != 0 )
11    {
12        quad = num * num;
13        Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15        Console.WriteLine("Digite um numero inteiro: ");
16 num = Convert.ToInt32(Console.ReadLine()); 17
17    }
18 }
```

### Condição

Teste normalmente envolve ao menos uma variável.

### Inicialização

Toda variável da condição precisa ser inicializada antes do laço, através de atribuição ou leitura.

# Repetição com Teste no Início

## Quesitos importantes em repetições

```
1
2 static void Main(string[] args)
3 {
4     int num, quad;
5
6     // imprime uma msg e le o 1o. Inteiro
7     Console.WriteLine("Digite um numero inteiro:");
8     num = Convert.ToInt32(Console.ReadLine());
9
10    while( num != 0 )
11    {
12        quad = num * num;
13        Console.WriteLine("Quadrado de {0}: {1}", num, quad);
14
15        Console.WriteLine("Digite um numero inteiro:");
16        num = Convert.ToInt32(Console.ReadLine());
17    }
18 }
```

### Condição

Teste normalmente envolve ao menos uma variável.

### Inicialização

Toda variável da condição precisa ser inicializada antes do laço, através de atribuição ou leitura.

### Atualização

Ao menos uma variável da condição precisa ser atualizada no interior do laço.

# Repetição com Teste no Início

## Exemplo 2: Uso de acumulador

- **Acumulador** é uma variável utilizada para armazenar a soma (ou o produto) de uma sequência de valores.
- A variável usada como **acumulador** recebe um acréscimo a cada iteração, isto é, seu valor anterior é usado em sua atualização ao longo das iterações.

```
soma = soma + novo_valor;
```

- A variável usada como **acumulador** precisa ser inicializada antes do laço.

```
soma = 0;  
produto = 1;
```



# Repetição com Teste no Início

## Exemplo 2: Uso de acumulador

Desenvolva um algoritmo que leia uma sequência de números inteiros, calcule e imprima a soma dos valores lidos. A sequência deve terminar quando o número 0 (zero) for lido.

# Repetição com Teste no Início

## Exemplo 2: Uso de acumulador

Desenvolva um algoritmo que leia uma sequência de números inteiros, calcule e imprima a soma dos valores lidos. A sequência deve terminar quando o número 0 (zero) for lido.

- soma** → o valor da variável começa com zero (elemento neutro da adição) ;
- a cada iteração, conserva o valor anterior com acréscimo de um novo valor;
  - ao final de cada iteração, o valor da variável contém a soma parcial dos elementos ;
  - ao final do laço, o valor da variável contém a soma dos elementos.

# Repetição com Teste no Início

## Exemplo 2: Uso de acumulador

```
1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}",soma);
12         Console.Write("Digite um numero inteiro: ");
13         num = Convert.ToInt32(Console.ReadLine());
14     }
15     Console.WriteLine("Soma total: {0}",soma);
16 }
```

## Exemplo 2: Uso de acumulador

## TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8 while( num != 0 )    9
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}", soma);
12         Console.Write("Digite um numero inteiro: ");
13         num = Convert.ToInt32(Console.ReadLine());
14     }
15     Console.WriteLine("\nSoma total: {0}", soma);
16 }
```

[illegible]

## Exemplo 2: Uso de acumulador

## TESTE DE MESA

```

1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}",soma);
12         Console.Write("Digite um numero inteiro: ");
13 num = Convert.ToInt32(Console.ReadLine()); 14
14     }
15     Console.WriteLine("\nSoma total: {0}",soma);
16 }

```

[illegible]

## Exemplo 2: Uso de acumulador

## TESTE DE MESA

```

1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}",soma);
12         Console.Write("Digite um numero inteiro: ");
13 num = Convert.ToInt32(Console.ReadLine()); 14
14     }
15     Console.WriteLine("\nSoma total: {0}",soma);
16 }

```

[illegible]

Digite um numero inteiro: 9

## Exemplo 2: Uso de acumulador

## TESTE DE MESA

```

1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}",soma);
12         Console.Write("Digite um numero inteiro: ");
13 num = Convert.ToInt32(Console.ReadLine()); 14
14     }
15     Console.WriteLine("\nSoma total: {0}",soma);
16 }

```

[illegible]

Digite um numero inteiro: 9

## Exemplo 2: Uso de acumulador

## TESTE DE MESA

```

1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}",soma);
12         Console.Write("Digite um numero inteiro: ");
13 num = Convert.ToInt32(Console.ReadLine()); 14
14     }
15     Console.WriteLine("\nSoma total: {0}",soma);
16 }

```

[illegible]

Digite um numero inteiro: 9



## Exemplo 2: Uso de acumulador

### TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8 while( num != 0 )
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}", soma);
12         Console.Write("Digite um numero inteiro: ");
13         num = Convert.ToInt32(Console.ReadLine());
14     }
15     Console.WriteLine("\nSoma total: {0}", soma);
16
17 }
```

linh	a	num	soma	teste
4		?	?	
5		?	0	
7		9	0	
8		9	0	V
10		9	9	
13		-2	9	

```
Digite um numero inteiro: 9
Soma parcial: 9
Digite um numero inteiro: -2
```

## Exemplo 2: Uso de acumulador

### TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}",soma);
12         Console.Write("Digite um numero inteiro: ");
13 num = Convert.ToInt32(Console.ReadLine()); 14
14     }
15     Console.WriteLine("\nSoma total: {0}",soma);
16 }
```

linh	a	num	soma	teste
4		?	?	
5		?	0	
7		9	0	
8		9	0	V
10		9	9	
13		-2	9	
8		-2	9	V

```
Digite um numero inteiro: 9
Soma parcial: 9
Digite um numero inteiro: -2
```

## Exemplo 2: Uso de acumulador

### TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}", soma);
12         Console.Write("Digite um numero inteiro: ");
13 num = Convert.ToInt32(Console.ReadLine()); 14
14     }
15     Console.WriteLine("\nSoma total: {0}", soma);
16 }
```

linh	a	num	soma	teste
4		?	?	
5		?	0	
7		9	0	
8		9	0	V
10		9	9	
13		-2	9	
8		-2	9	V
10		-2	7	

```
Digite um numero inteiro: 9
Soma parcial: 9
Digite um numero inteiro: -2
```

## Exemplo 2: Uso de acumulador

### TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}",soma);
12         Console.Write("Digite um numero inteiro: ");
13 num = Convert.ToInt32(Console.ReadLine()); 14
14     }
15     Console.WriteLine("\nSoma total: {0}",soma);
16 }
```

li	nha	num	soma	teste
	4	?	?	
	5	?	0	
	7	9	0	
	8	9	0	V
10		9	9	
13		-2	9	
	8	-2	9	V
10		-2	7	
13		0	7	

```
Digite um numero inteiro: 9
Soma parcial: 9
Digite um numero inteiro: -2
Soma parcial: 7
Digite um numero inteiro: 0
```

## Exemplo 2: Uso de acumulador

### TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}",soma);
12         Console.Write("Digite um numero inteiro: ");
13 num = Convert.ToInt32(Console.ReadLine()); 14
14     }
15     Console.WriteLine("\nSoma total: {0}",soma);
16 }
```

linha	num	soma	teste
4	?	?	
5	?	0	
7	9	0	
8	9	0	V
10	9	9	
13	-2	9	
8	-2	9	V
10	-2	7	
13	0	7	
8	0	7	F

```
Digite um numero inteiro: 9
Soma parcial: 9
Digite um numero inteiro: -2
Soma parcial: 7
Digite um numero inteiro: 0
```

## Exemplo 2: Uso de acumulador

### TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}",soma);
12         Console.Write ("Digite um numero inteiro: ")
13 num = Convert.ToInt32(Console.ReadLine());
14     }
15     Console.WriteLine("\nSoma total: {0}",soma);
16 }
```

```
Digite um numero inteiro: 9
Soma parcial: 9
Digite um numero inteiro: -2
Soma parcial: 7
Digite um numero inteiro: 0
Soma total: 7
```

li	nha	num	soma	teste
	4	?	?	
	5	?	0	
	7	9	0	
	8	9	0	V
10		9	9	
13		-2	9	
	8	-2	9	V
10		-2	7	
13		0	7	
	8	0	7	F
15		0	7	

# Repetição com Teste no Início

Os quesitos para o uso de laços são mantidos:

```
1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while (num != 0)
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}", soma);
12         Console.Write("Digite um numero inteiro: ");
13         num = Convert.ToInt32(Console.ReadLine());
14     }
15     Console.WriteLine("\nSoma total: {0}", soma);
16 }
```

## Condição

Teste normalmente envolve ao menos uma variável.

## Inicialização

Toda variável da condição precisa ser inicializada antes do laço, através de atribuição ou leitura.

## Atualização

Ao menos uma variável da condição precisa ser atualizada no interior do laço.

# Repetição com Teste no Início

Acumuladores precisam de:

## Inicialização

Variável precisa ser inicializada (normalmente com elemento neutro).

## Atualização

Variável precisa conservar o valor acumulado e acrescentar novo valor.

```
1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10        soma = soma + num; //atualiza acumulador
11        Console.WriteLine("Soma parcial: {0}", soma);
12        Console.Write("Digite um numero inteiro: ");
13        num = Convert.ToInt32(Console.ReadLine());
14    }
15    Console.WriteLine("\nSoma total: {0}", soma);
16 }
```



# Repetição com Teste no Início

## Impressão de acumuladores:

```
1
2 static void Main(string[] args)
3 {
4     int num, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         soma = soma + num; //atualiza acumulador
11         Console.WriteLine("Soma parcial: {0}", soma);
12         Console.Write("Digite um numero inteiro: ");
13         num = Convert.ToInt32(Console.ReadLine());
14     }
15     Console.WriteLine("\nSoma total: {0}", soma);
16 }
```

### Valor parcial

Impressão da variável é feita no laço, após sua atualização.

### Valor total

Impressão da variável é feita após o laço.

# Repetição com Teste no Início

## Exemplo 3: Uso de contador

- **Contador** é uma variável utilizada para contar o número de ocorrências de determinado evento.
- A variável usada como **contador** recebe um acréscimo de uma unidade a cada iteração, isto é, é incrementada de 1 em 1.

```
contador = contador + 1;  
contador++;
```

- A variável usada como **acumulador** precisa ser inicializada antes do laço.

```
contador = 0;
```

# Repetição com Teste no Início

## Exemplo 3: Uso de contador

Desenvolva um algoritmo que leia uma sequência de números inteiros e imprima a quantidade de valores lidos. A sequência deve terminar quando o número 0 (zero) for lido.

# Repetição com Teste no Início

## Exemplo 3: Uso de contador

Desenvolva um algoritmo que leia uma sequência de números inteiros e imprima a quantidade de valores lidos. A sequência deve terminar quando o número 0 (zero) for lido.

***contador*** → o valor da variável começa com zero;  
→ após ler e testar cada novo valor da sequência, a variável é incrementada (aumenta em uma unidade);  
→ ao final do laço, o valor da variável contém o número de elementos.

# Repetição com Teste no Início

## Exemplo 3: Uso de contador

```
1
2 static void Main(string[] args)
3 {
4     int num, cont;
5     cont = 0; // inicializa contador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         cont++; //atualiza contador
11         Console.Write("Digite um numero inteiro: ");
12         num = Convert.ToInt32(Console.ReadLine());
13     }
14     Console.WriteLine("\nNumero de elementos: {0}", cont);
15 }
```

### Exemplo 3: Uso de contador

## TESTE DE MESA

```

1
2 static void Main(string[] args)
3 {
4     int num, cont;
5     cont = 0; // inicializa contador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         cont++; //atualiza contador
11         Console.Write("Digite um numero inteiro: ");
12 num = Convert.ToInt32(Console.ReadLine()); 13
13     }
14     Console.Write("Numero de elementos: {0}", cont);
15 }

```

[illegible]

### Exemplo 3: Uso de contador

## TESTE DE MESA

```

1
2 static void Main(string[] args)
3 {
4     int num, cont;
5     cont = 0; // inicializa contador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         cont++; //atualiza contador
11         Console.Write("Digite um numero inteiro: ");
12 num = Convert.ToInt32(Console.ReadLine()); 13
13     }
14     Console.Write("Numero de elementos: {0}", cont);
15 }

```

[illegible]

### Exemplo 3: Uso de contador

## TESTE DE MESA

```

1
2 static void Main(string[] args)
3 {
4     int num, cont;
5     cont = 0; // inicializa contador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         cont++; //atualiza contador
11         Console.Write("Digite um numero inteiro: ");
12 num = Convert.ToInt32(Console.ReadLine()); 13
13     }
14     Console.Write("Numero de elementos: {0}", cont);
15 }

```

Digite um numero inteiro: 22

[illegible]



### Exemplo 3: Uso de contador

## TESTE DE MESA

```

1
2 static void Main(string[] args)
3 {
4     int num, cont;
5     cont = 0; // inicializa contador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         cont++; //atualiza contador
11         Console.Write("Digite um numero inteiro: ");
12 num = Convert.ToInt32(Console.ReadLine()); 13
13     }
14     Console.Write("Numero de elementos: {0}", cont);
15 }

```

[illegible]

Digite um numero inteiro: 22

### Exemplo 3: Uso de contador

## TESTE DE MESA

```

1
2 static void Main(string[] args)
3 {
4     int num, cont;
5     cont = 0; // inicializa contador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         cont++; //atualiza contador
11         Console.Write("Digite um numero inteiro: ");
12 num = Convert.ToInt32(Console.ReadLine()); 13
13     }
14     Console.Write("Numero de elementos: {0}", cont);
15 }

```

Digite um numero inteiro: 22

[illegible]

### Exemplo 3: Uso de contador

## TESTE DE MESA

[illegible]

```

1
2 static void Main(string[] args)
3 {
4     int num, cont;
5     cont = 0; // inicializa contador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         cont++; //atualiza contador
11         Console.Write("Digite um numero inteiro: ");
12 num = Convert.ToInt32(Console.ReadLine()); 13
13     }
14     Console.Write("Numero de elementos: {0}", cont);
15 }

```

```

Digite um numero inteiro: 22
Digite um numero inteiro: 13

```

## Exemplo 3: Uso de contador

### TESTE DE MESA

linha	num	cont	teste
4	?	?	
5	?	0	
7	22	0	
8	22	0	V
10	22	1	
12	13	1	
8	13	1	V

```
1
2 static void Main(string[] args)
3 {
4     int num, cont;
5     cont = 0; // inicializa contador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         cont++; //atualiza contador
11         Console.Write("Digite um numero inteiro: ");
12 num = Convert.ToInt32(Console.ReadLine()); 13
13     }
14     Console.Write("Numero de elementos: {0}", cont);
15 }
```

```
Digite um numero inteiro: 22
Digite um numero inteiro: 13
```

## Exemplo 3: Uso de contador

### TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int num, cont;
5     cont = 0; // inicializa contador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         cont++; //atualiza contador
11         Console.Write("Digite um numero inteiro: ");
12         num = Convert.ToInt32(Console.ReadLine());
13     }
14     Console.Write("Numero de elementos: {0}", cont);
15 }
```

linha	num	cont	teste
4	?	?	
5	?	0	
7	22	0	
8	22	0	V
10	22	1	
12	13	1	
8	13	1	V
10	13	2	

```
Digite um numero inteiro: 22
Digite um numero inteiro: 13
```

## Exemplo 3: Uso de contador

### TESTE DE MESA

linha	num	cont	teste
4	?	?	
5	?	0	
7	22	0	
8	22	0	V
10	22	1	
12	13	1	
8	13	1	V
10	13	2	
12	0	2	

```
1
2 static void Main(string[] args)
3 {
4     int num, cont;
5     cont = 0; // inicializa contador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         cont++; //atualiza contador
11         Console.Write("Digite um numero inteiro: ");
12 num = Convert.ToInt32(Console.ReadLine()); 13
13     }
14     Console.Write("Numero de elementos: {0}", cont);
15 }
```

```
Digite um numero inteiro: 22
Digite um numero inteiro: 13
Digite um numero inteiro: 0
```

## Exemplo 3: Uso de contador

### TESTE DE MESA

linha	num	cont	teste
4	?	?	
5	?	0	
7	22	0	
8	22	0	V
10	22	1	
12	13	1	
8	13	1	V
10	13	2	
12	0	2	
8	0	2	F

```
1
2 static void Main(string[] args)
3 {
4     int num, cont;
5     cont = 0; // inicializa contador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         cont++; //atualiza contador
11         Console.Write("Digite um numero inteiro: ");
12 num = Convert.ToInt32(Console.ReadLine()); 13
13     }
14     Console.Write("Numero de elementos: {0}", cont);
15 }
```

```
Digite um numero inteiro: 22
Digite um numero inteiro: 13
Digite um numero inteiro: 0
```

## Exemplo 3: Uso de contador

### TESTE DE MESA

linha	num	cont	teste
4	?	?	
5	?	0	
7	22	0	
8	22	0	V
10	22	1	
12	13	1	
8	13	1	V
10	13	2	
12	0	2	
8	0	2	F
14	0	2	

```
1
2 static void Main(string[] args)
3 {
4     int num, cont;
5     cont = 0; // inicializa contador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         cont++; //atualiza contador
11         Console.Write("Digite um numero inteiro: ");
12 num = Convert.ToInt32(Console.ReadLine()); 13
13     }
14 Console.Write("Numero de elementos: {0}", cont);
15 }
```

```
Digite um numero inteiro: 22
Digite um numero inteiro: 13
Digite um numero inteiro: 0
Numero de elementos: 2
```



# Repetição com Teste no Início

## Exemplo 4: Cálculo de média

Desenvolva um algoritmo que leia uma sequência de números inteiros e imprima a média aritmética dos valores lidos. A sequência deve terminar quando o número 0 (zero) for lido.

# Repetição com Teste no Início

## Exemplo 4: Cálculo de média

Desenvolva um algoritmo que leia uma sequência de números inteiros e imprima a média aritmética dos valores lidos. A sequência deve terminar quando o número 0 (zero) for lido.

**media** → a média aritmética corresponde à soma dos valores dividida pelo número de valores;

- um acumulador e um contador serão necessários neste cálculo;
- o resultado da divisão deve ser um valor real, mesmo se os valores envolvidos são inteiros;
- a divisão deve ser realizada após o fim do laço, quando os valores do contador e do acumulador já não sofrerão alterações.

# Repetição com Teste no Início

## Exemplo 4: Cálculo de média

```
1
2 static void Main(string[] args)
3 {
4     int num, soma, cont;
5     double media;
6     soma = 0;           // inicializa acumulador
7     cont = 0;          // inicializa contador
8     Console.Write("Digite um numero inteiro: ");
9     num = Convert.ToInt32(Console.ReadLine());
10    while( num != 0 )
11    {
12        soma = soma + num; //atualiza acumulador
13        cont++;           //atualiza contador
14        Console.Write("Digite um numero inteiro: ");
15        num = Convert.ToInt32(Console.ReadLine());
16    }
17    media = soma / (double) cont;
18    Console.WriteLine("\nMedia dos elementos: {0}", media);
19 }
```

# Repetição com Teste no Início

Exemplo 5: Soma dos dígitos de um número

Desenvolva um algoritmo que leia um número inteiro positivo e imprima a soma de seus dígitos.

# Repetição com Teste no Início

## Exemplo 5: Soma dos dígitos de um número

Desenvolva um algoritmo que leia um número inteiro positivo e imprima a soma de seus dígitos.

→ a separação dos dígitos de um número pode ser feita através das seguintes operações:

- o resto da divisão por 10 permite que o dígito menos significativo seja obtido:

$$530479 \% 10 \rightarrow 9$$

- a divisão inteira por 10 permite que o dígito menos significativo seja descartado e o dígito seguinte assuma esta posição:

$$530479 / 10 \rightarrow 53047$$

# Repetição com Teste no Início

## Exemplo 5: Soma dos dígitos de um número

Desenvolva um algoritmo que leia um número inteiro positivo e imprima a soma de seus dígitos.

→ a separação dos dígitos de um número pode ser feita através das seguintes operações:

$$530479 \% 10 \rightarrow 9$$

$$\underline{530479 / 10 \rightarrow 53047}$$

$$53047 \% 10 \rightarrow 7$$

$$\underline{53047 / 10 \rightarrow 5304}$$

$$5304 \% 10 \rightarrow 4$$

$$\underline{5304 / 10 \rightarrow 530}$$

$$530 \% 10 \rightarrow 0$$

$$\underline{530 / 10 \rightarrow 53}$$

$$53 \% 10 \rightarrow 3$$

$$\underline{53 / 10 \rightarrow 5}$$

$$5 \% 10 \rightarrow 5$$

$$\underline{5 / 10 \rightarrow 0}$$

# Repetição com Teste no Início

## Exemplo 5: Soma dos dígitos de um número

Desenvolva um algoritmo que leia um número inteiro positivo e imprima a soma de seus dígitos.

→ a separação dos dígitos de um número pode ser feita através das seguintes operações:

$530479 \% 10 \rightarrow 9$

$530479 / 10 \rightarrow 53047$

$53047 \% 10 \rightarrow 7$

$53047 / 10 \rightarrow 5304$

$5304 \% 10 \rightarrow 4$

$5304 / 10 \rightarrow 530$

...

$\text{digito} = \text{num} \% 10;$

$\text{num} = \text{num} / 10;$

$\text{digito} = \text{num} \% 10;$

$\text{num} = \text{num} / 10;$

$\text{digito} = \text{num} \% 10;$

$\text{num} = \text{num} / 10;$

...

# Repetição com Teste no Início

## Exemplo 5: Soma dos dígitos de um número

```
1
2 static void Main(string[] args)
3 {
4     int num, digito, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         digito = num % 10; // obtem digito menos significativo
11         soma = soma + digito; // acrescenta o valor do digito a soma
12         num = num / 10; // descarta o digito armazenado
13     }
14     Console.WriteLine("\nSoma dos digitos: {0}", soma);
15
16 }
```



# Repetição com Teste no Início

## Exemplo 5: Soma dos dígitos de um número

```
1
2 static void Main(string[] args)
3 {
4     int num, digito, soma;
5     soma = 0; // inicializa acumulador
6     Console.Write("Digite um numero inteiro: ");
7     num = Convert.ToInt32(Console.ReadLine());
8     while( num != 0 )
9     {
10         digito = num % 10; // obtem digito menos significativo
11         soma = soma + digito; // acrescenta o valor do digito a soma
12         num = num / 10; // descarta o digito armazenado
13     }
14     Console.WriteLine("\nSoma dos digitos: {0}", soma);
15
16 }
```

Atualização da variável da condição

# Repetição com variável de controle

```
for ( inicializacao; condicao; atualizacao )  
{  
    blocoDeComandos1;  
}  
blocoDeComandos2;
```

O uso do comando **for** é muito similar ao do comando **while**.

# Repetição com variável de controle

```
for ( inicializacao; condicao; atualizacao )  
{  
    blocoDeComandos1;  
}  
blocoDeComandos2;
```

```
inicializacao;  
while ( condicao )  
{  
    blocoDeComandos1;  
    atualizacao;  
}  
blocoDeComandos2;
```

O **for** concentra os comandos de inicialização, condição e atualização entre parênteses, separados por ponto e vírgula.

# Repetição com variável de controle

```
for ( inicializacao; condicao; atualizacao )  
{  
    blocoDeComandos1;  
}  
blocoDeComandos2;
```

```
inicializacao;  
while ( condicao )  
{  
    blocoDeComandos1;  
    atualizacao;  
}  
blocoDeComandos2;
```

## inicialização

O comando de inicialização do **for** é realizado uma única vez, antes de sua primeira execução.

# Repetição com variável de controle

```
for ( inicializacao; condicao; atualizacao )  
{  
    blocoDeComandos1;  
}  
blocoDeComandos2;
```

```
inicializacao;  
while ( condicao )  
{  
    blocoDeComandos1;  
    atualizacao;  
}  
blocoDeComandos2;
```

## condição

A condição do **for** é testada no início de toda iteração. O bloco de comandos interno só é executado se a condição for VERDADEIRA.

# Repetição com variável de controle

```
for ( inicializacao; condicao; atualizacao )  
{  
    blocoDeComandos1;  
}  
blocoDeComandos2;
```

```
inicializacao;  
while ( condicao )  
{  
    blocoDeComandos1;  
    atualizacao;  
}  
blocoDeComandos2;
```

## atualização

A atualização é realizada após a execução do bloco de comandos interno e antes do teste da condição da iteração seguinte.

# Repetição com variável de controle

```
for ( inicializacao; condicao; atualizacao )  
{  
    blocoDeComandos1;  
}  
blocoDeComandos2;
```

```
inicializacao;  
while ( condicao )  
{  
    blocoDeComandos1;  
    atualizacao;  
}  
blocoDeComandos2;
```

Funcionalmente, **for** e **while** são idênticos, apenas a sintaxe dos comandos muda.

# Repetição com variável de controle

```
for ( contador=0; contador<5; contador++ )  
{  
    blocoDeComandos1;  
}  
blocoDeComandos2;
```

```
contador = 0;  
while ( contador < 5 )  
{  
    blocoDeComandos1;  
    contador++;  
}  
blocoDeComandos2;
```

Em geral, **for** é mais usado quando há uma variável de controle, como um contador na condição.



# Repetição com variável de controle

```
for ( contador=0; contador<5; contador++ )  
{  
    blocoDeComandos1;  
}  
blocoDeComandos2;
```

```
contador = 0;  
while ( contador < 5 )  
{  
    blocoDeComandos1;  
    contador++;  
}  
blocoDeComandos2;
```

A vantagem do **for** neste caso é concentrar inicialização, condição e atualização, evitando que algum destes comandos seja esquecido.

# Repetição com variável de controle

```
for ( contador=0; contador<5; contador++ )  
{  
    blocoDeComandos1;  
}  
blocoDeComandos2;
```

```
contador = 0;  
while ( contador < 5 )  
{  
    blocoDeComandos1;  
    contador++;  
}  
blocoDeComandos2;
```

O comando **while** é mais indicado quando o controle do laço envolve FLAG ou condições complexas, uma vez que o código fica mais claro de ser entendido.

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

Desenvolva um algoritmo que imprima todos os valores inteiros em um intervalo indicado pelo usuário.

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

Desenvolva um algoritmo que imprima todos os valores inteiros em um intervalo indicado pelo usuário.

- o usuário vai indicar o valor inicial e o valor final do intervalo;
- todos os valores do intervalo devem ser impressos, um a um;
- um contador pode ser utilizado para indicar cada valor a ser impresso.

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	

```
Digite o menor numero: 3
```

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	
8	3	5	?	

```
Digite o menor numero: 3
Digite o maior numero: 5
```



# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	
8	3	5	?	
10	3	5	3	

```
Digite o menor numero: 3
Digite o maior numero: 5
Intervalo:
```

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	
8	3	5	?	
10	3	5	3	
10	3	5	3	V

```
Digite o menor numero: 3
Digite o maior numero: 5
Intervalo:
```

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

```

Digite o menor numero: 3
Digite o maior numero: 5
Intervalo: 3

```

[illegible]

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	
8	3	5	?	
10	3	5	3	
10	3	5	3	V
10	3	5	4	

```
Digite o menor numero: 3
Digite o maior numero: 5
Intervalo: 3
```

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

```
Digite o menor numero: 3
Digite o maior numero: 5
Intervalo: 3
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	
8	3	5	?	
10	3	5	3	
10	3	5	3	V
10	3	5	4	
10	3	5	4	V

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

```
Digite o menor numero: 3
Digite o maior numero: 5
Intervalo: 3 4
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	
8	3	5	?	
10	3	5	3	
10	3	5	3	V
10	3	5	4	
10	3	5	4	V

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	
8	3	5	?	
10	3	5	3	
10	3	5	3	V
10	3	5	4	
10	3	5	4	V
10	3	5	5	

```
Digite o menor numero: 3
Digite o maior numero: 5
Intervalo: 3 4
```

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

```
Digite o menor numero: 3
Digite o maior numero: 5
Intervalo: 3 4
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	
8	3	5	?	
10	3	5	3	
10	3	5	3	V
10	3	5	4	
10	3	5	4	V
10	3	5	5	
10	3	5	5	V



# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

```
Digite o menor numero: 3
Digite o maior numero: 5
Intervalo: 3 4 5
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	
8	3	5	?	
10	3	5	3	
10	3	5	3	V
10	3	5	4	
10	3	5	4	V
10	3	5	5	
10	3	5	5	V

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

```
Digite o menor numero: 3
Digite o maior numero: 5
Intervalo: 3 4 5
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	
8	3	5	?	
10	3	5	3	
10	3	5	3	V
10	3	5	4	
10	3	5	4	V
10	3	5	5	
10	3	5	5	V
10	3	5	6	

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

```
Digite o menor numero: 3
Digite o maior numero: 5
Intervalo: 3 4 5
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	
8	3	5	?	
10	3	5	3	
10	3	5	3	V
10	3	5	4	
10	3	5	4	V
10	3	5	5	
10	3	5	5	V
10	3	5	6	
10	3	5	6	F

# Repetição com variável de controle

## Exemplo 1: Imprime valores em um intervalo

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     Console.Write("\nIntervalo: ");
10    for( cont=inicio; cont<=fim; cont++ )
11    {
12        Console.Write(" {0} ", cont);
13    }
14
15 }
```

```
Digite o menor numero: 3
Digite o maior numero: 5
Intervalo: 3 4 5
```

### TESTE DE MESA

linha	inicio	fim	cont	teste
4	?	?	?	
6	3	?	?	
8	3	5	?	
10	3	5	3	
10	3	5	3	V
10	3	5	4	
10	3	5	4	V
10	3	5	5	
10	3	5	5	V
10	3	5	6	
10	3	5	6	F

# Repetição com variável de controle

## Exemplo 2: Imprime tabuada

Desenvolva um algoritmo que leia dois números inteiros **tab** e **limite** e imprima a tabuada de **tab** desde 1 até **limite**.

# Repetição com variável de controle

## Exemplo 2: Imprime tabuada

Desenvolva um algoritmo que leia dois números inteiros **tab** e **limite** e imprima a tabuada de **tab** desde 1 até **limite**.

- o usuário vai indicar os valores **tab** e **limite**;
- todos os múltiplos de **tab** devem ser impressos, um a um, com multiplicadores variando de 1 a **limite**;
- um contador pode ser utilizado para armazenar os multiplicadores.

# Repetição com variável de controle

## Exemplo 2: Imprime tabuada

```
1
2  static void Main(string[] args)
3  {
4      int cont, tab, limite;
5      Console.Write("Tabuada de:");
6      tab = Convert.ToInt32(Console.ReadLine());
7      Console.Write("e o ultimo multiplicador:");
8      limite = Convert.ToInt32(Console.ReadLine());
9      Console.WriteLine("\nTabuada de {0}: ", tab);
10 for( cont=1; cont<=limite; cont++ )
11 {
12     Console.Write("\n{0} x {1} = {2}",
13                 tab, cont, cont*tab);
14 }
15 }
```

## Exemplo 2: Imprime tabuada

### TESTE DE MESA

linha	tab	limite	cont	teste
4	?	?	?	

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++ )
11    {
12        Console.Write("\n{0} x {1} = {2}",
13                        tab, cont, cont*tab);
14    }
15 }
```



## Exemplo 2: Imprime tabuada

### TESTE DE MESA

linha	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	

```
1
2  static void Main(string[] args)
3  {
4      int cont, tab, limite;
5      Console.Write("Tabuada de:");
6      tab = Convert.ToInt32(Console.ReadLine());
7      Console.Write(" e o ultimo multiplicador:");
8      limite = Convert.ToInt32(Console.ReadLine());
9      Console.WriteLine("\nTabuada de {0}: ", tab);
10     for( cont=1; cont<=limite; cont++ )
11     {
12         Console.Write("\n{0} x {1} = {2}",
13             tab, cont, cont*tab);
14     }
15 }
```

Tabuada de: 5

## Exemplo 2: Imprime tabuada

## TESTE DE MESA

[illegible]

Tabuada de: 5 e o ultimo multiplicador: 3

## Exemplo 2: Imprime tabuada

### TESTE DE MESA

	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	
8	5	3	?	
10	5	3	1	

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++ )
11    {
12        Console.Write("\n{0} x {1} = {2}",
13            tab, cont, cont*tab);
14    }
15 }
```

```
Tabuada de: 5 e o ultimo multiplicador: 3
```

## Exemplo 2: Imprime tabuada

### TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++ )
11    {
12        Console.Write("\n{0} x {1} = {2}",
13            tab, cont, cont*tab);
14    }
15 }
```

linha	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	
8	5	3	?	
10	5	3	1	
10	5	3	1	V

```
Tabuada de: 5 e o ultimo multiplicador: 3
```

## Exemplo 2: Imprime tabuada

### TESTE DE MESA

linha	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	
8	5	3	?	
10	5	3	1	
10	5	3	1	V

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++ )
11    {
12        Console.Write("\n{0} x {1} = {2}",
13                      tab, cont, cont*tab);
14    }
15 }
```

```
Tabuada de: 5 e o ultimo multiplicador: 3
5 x 1 = 5
```

## Exemplo 2: Imprime tabuada

### TESTE DE MESA

linha	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	
8	5	3	?	
10	5	3	1	
10	5	3	1	V
10	5	3	2	

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++)
11    {
12        Console.WriteLine("\n{0} x {1} = {2}",
13                            tab, cont, cont*tab);
14    }
15 }
```

```
Tabuada de: 5 e o ultimo multiplicador: 3
5 x 1 = 5
```

## Exemplo 2: Imprime tabuada

### TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++)
11    {
12        Console.WriteLine("\n{0} x {1} = {2}",
13                           tab, cont, cont*tab);
14    }
15 }
```

linha	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	
8	5	3	?	
10	5	3	1	
10	5	3	1	V
10	5	3	2	
10	5	3	2	V

```
Tabuada de: 5 e o ultimo multiplicador: 3
5 x 1 = 5
```

## Exemplo 2: Imprime tabuada

### TESTE DE MESA

linha	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	
8	5	3	?	
10	5	3	1	
10	5	3	1	V
10	5	3	2	
10	5	3	2	V

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++ )
11    {
12        Console.Write("\n{0} x {1} = {2}",
13                        tab, cont, cont*tab);
14    }
15 }
```

```
Tabuada de: 5 e o ultimo multiplicador: 3
5 x 1 = 5
5 x 2 = 10
```



## Exemplo 2: Imprime tabuada

### TESTE DE MESA

linha	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	
8	5	3	?	
10	5	3	1	
10	5	3	1	V
10	5	3	2	
10	5	3	2	V
10	5	3	3	

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++)
11    {
12        Console.WriteLine("\n{0} x {1} = {2}",
13                           tab, cont, cont*tab);
14    }
15 }
```

```
Tabuada de: 5 e o ultimo multiplicador: 3
5 x 1 = 5
5 x 2 = 10
```

## Exemplo 2: Imprime tabuada

### TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++ )
11    {
12        Console.WriteLine("\n{0} x {1} = {2}",
13                           tab, cont, cont*tab);
14    }
15 }
```

linha	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	
8	5	3	?	
10	5	3	1	
10	5	3	1	V
10	5	3	2	
10	5	3	2	V
10	5	3	3	
10	5	3	3	V

```
Tabuada de: 5 e o ultimo multiplicador: 3
5 x 1 = 5
5 x 2 = 10
```

## Exemplo 2: Imprime tabuada

### TESTE DE MESA

linha	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	
8	5	3	?	
10	5	3	1	
10	5	3	1	V
10	5	3	2	
10	5	3	2	V
10	5	3	3	
10	5	3	3	V

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++ )
11    {
12        Console.Write("\n{0} x {1} = {2}",
13            tab, cont, cont*tab);
14    }
15 }
```

```
Tabuada de: 5 e o ultimo multiplicador: 3
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
```

## Exemplo 2: Imprime tabuada

### TESTE DE MESA

linha	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	
8	5	3	?	
10	5	3	1	
10	5	3	1	V
10	5	3	2	
10	5	3	2	V
10	5	3	3	
10	5	3	3	V
10	5	3	4	

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++)
11    {
12        Console.WriteLine("\n{0} x {1} = {2}",
13            tab, cont, cont*tab);
14    }
15 }
```

```
Tabuada de: 5 e o ultimo multiplicador: 3
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
```

## Exemplo 2: Imprime tabuada

### TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++ )
11    {
12        Console.WriteLine("\n{0} x {1} = {2}",
13                           tab, cont, cont*tab);
14    }
15 }
```

linha	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	
8	5	3	?	
10	5	3	1	
10	5	3	1	V
10	5	3	2	
10	5	3	2	V
10	5	3	3	
10	5	3	3	V
10	5	3	4	
10	5	3	4	F

```
Tabuada de: 5 e o ultimo multiplicador: 3
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
```

## Exemplo 2: Imprime tabuada

### TESTE DE MESA

```
1
2 static void Main(string[] args)
3 {
4     int cont, tab, limite;
5     Console.Write("Tabuada de:");
6     tab = Convert.ToInt32(Console.ReadLine());
7     Console.Write(" e o ultimo multiplicador:");
8     limite = Convert.ToInt32(Console.ReadLine());
9     Console.WriteLine("\nTabuada de {0}: ", tab);
10    for( cont=1; cont<=limite; cont++)
11    {
12        Console.WriteLine("\n{0} x {1} = {2}",
13                           tab, cont, cont*tab);
14    }
15 }
```

linha	tab	limite	cont	teste
4	?	?	?	
6	5	?	?	
8	5	3	?	
10	5	3	1	
10	5	3	1	V
10	5	3	2	
10	5	3	2	V
10	5	3	3	
10	5	3	3	V
10	5	3	4	
10	5	3	4	F

```
Tabuada de: 5 e o ultimo multiplicador: 3
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
```

# Repetição com teste no final

```
do
{
    atualizacao;
    blocoDeComandos1;
} while ( condicao );
blocoDeComandos2;
```

O comando **do-while** é similar aos comandos **while** e **for**, mas, neste caso, o teste da condição acontece em momentos distintos.

# Repetição com teste no final

```
do
{
    atualizacao;
    blocoDeComandos1;
} while ( condicao );
blocoDeComandos2;
```

```
inicializacao;
while ( condicao )
{
    blocoDeComandos1;
    atualizacao;
}
blocoDeComandos2;
```

Como **do-while** só testa a condição após a primeira execução do bloco de comandos, este bloco certamente será executado ao menos uma vez no programa.



# Repetição com teste no final

**Exemplo 1:** Imprime o quadrado de 15 inteiros  
Desenvolva um algoritmo que leia 15 números inteiros e imprima o quadrado de cada um deles.

```
1
2 static void Main(string[] args)
3 {
4     int cont, num;
5     cont = 0; //inicializa contador
6     do
7     {
8         Console.Write("Digite o {0}o numero: ", cont);
9         num = Convert.ToInt32(Console.ReadLine());
10        Console.WriteLine("Quadrado de {0}: {1} ",
11            num, num*num );
12        cont++;
13    } while ( cont < 15 );
14
15 }
```

# Repetição com teste no final

Qual é a diferença?

```
static void Main(string[] args)
{
    int cont;
    cont = 0;
    do
    {
        Console.WriteLine("{0}", cont);
        cont++;
    } while ( cont < 10 );
}
```

```
static void Main(string[] args)
{
    int cont;
    cont = 0;
    while ( cont < 10 )
    {
        Console.WriteLine("{0}", cont);
        cont++;
    }
}
```

# Repetição com teste no final

**Exemplo 2:** Imprime valores em um intervalo  
Desenvolva um algoritmo que imprima todos os valores inteiros em um intervalo indicado pelo usuário.

```
1
2 static void Main(string[] args)
3 {
4     int cont, inicio, fim;
5     Console.Write("Digite o menor numero: ");
6     inicio = Convert.ToInt32(Console.ReadLine());
7     Console.Write("Digite o maior numero: ");
8     fim = Convert.ToInt32(Console.ReadLine());
9     cont = inicio;
10    Console.Write("\nIntervalo: ");
11    do
12    {
13        Console.Write(" {0} ", cont);
14        cont++;
15    } while( cont <= fim );
16 }
```

# Estruturas de Controle

- Uma estrutura de controle controla o fluxo de execução dos comandos que constituem o seu domínio (ou bloco).
- Podem ser:
  1. Seqüência Simples.
  2. Alternativa:
    1. Simples (**if**).
    2. Dupla (**if-else**).
    3. Múltipla Escolha (**switch-case**).
  3. Repetição:
    1. Com Teste no Início (**while**).
    2. Com Variável de Controle (**for**).
    3. Com Teste no Final (**do-while**).

# Estruturas de Controle

- Em qualquer estrutura de controle, só existe um ponto de entrada e um ponto de saída do fluxo de execução.
- Formato geral de um algoritmo (programa):

Programa

{

    declaração de variáveis

    inicialização de variáveis

    corpo do algoritmo controlando leitura,  
        processamento e impressão dos dados

}

# Estruturas de Controle

- O corpo do algoritmo é constituído **exclusivamente** de estruturas de controle.
- Só existem duas maneiras de se ligar duas estruturas de controle do corpo de um algoritmo:

## 1. Em seqüência:

■ ■ ■

início da estrutura 1  
fim da estrutura 1

início da estrutura 2  
fim da estrutura 2

■ ■ ■

## 2. Encadeadas (concatenadas):

■ ■ ■

início da estrutura 1

fim da estrutura 1

333

# Comandos iterativos Exercícios

1. Fazer um algoritmo que imprima todos os números pares no intervalo de 1 a 100.
2. Fazer um algoritmo que imprima todos os números de 100 até 1.
3. Ler um número inteiro e positivo e verificar se este é ou não um número primo.
4. Dada uma dívida de 10000 reais que cresce a juros de 2,5% ao mês e uma aplicação de 1500 reais com rendimento de 4% ao mês, escrever um algoritmo que determina o número de meses necessários para pagar a dívida.
5. Calcular o valor de S:

$$S = \frac{2}{50} + \frac{2^2}{48} + \frac{2^3}{46} + \dots + \frac{2^{25}}{2}$$

# Comandos de Repetição

Aula de Exercícios



# Comandos iterativos em C

- **while** (enquanto..faça)
- **do...while** (faça..enquanto)
- **for** (para..faça)

# while

- Sintaxe

```
while (condicao)
{
    blocoDeComandos;
}
```

- Exemplo

```
static void Main(string[] args)
{
    int i;
    i=0;
    while (i<=10)
    {
        Console.WriteLine("{0}",i);
        i++;
    }
}
```

# do while

- Sintaxe

```
do
{
    blocoDeComandos;
} while (condicao);
```

- Exemplo

```
static void Main(string[] args)
{
    int i;
    i=0;
    do
    {
        Console.WriteLine("{0}",i);
        i++;
    } while (i<=10);
}
```

# for

- Sintaxe

```
for ( inicializacao ; icondicao ; atualizacao )  
{  
    blocoDeComandos;  
}
```

- Exemplo

```
static void Main(string[] args)  
{  
    int i;  
    for ( i=0 ; i<=10 ; i++)  
    {  
        Console.WriteLine("{0}",i);  
    }  
}
```

# Exercícios

- 1) Escrever um algoritmo que lê um valor N inteiro e positivo e que calcula e escreve o valor de E.  
$$E = 1 + 1/2 + 1/3 + \dots + 1/N$$
- 2) Escreva um algoritmo que leia 10 valores (usando a mesma variável) e encontre o maior e o menor deles. Mostre o resultado.
- 3) Chico tem 1,50 metros e cresce 2 centímetros por ano, enquanto Zé tem 1,40 metros e cresce 3 centímetros por ano. Construa um algoritmo que calcule e imprima quantos anos serão necessários para que Zé seja maior que Chico.
- 4) Escreva um algoritmo que leia a matrícula de um aluno e suas três notas. Calcule a média ponderada do aluno, considerando que o peso para a maior nota seja 4 e para as duas restantes, 3. Mostre ao final a média calculada e uma mensagem "APROVADO" se a média for maior ou igual a 5 e "REPROVADO" se a média for menor que 5. Repita a operação até que o código lido seja negativo.
- 5) Escrever um algoritmo que lê um valor N inteiro e positivo e que calcula e escreve o valor de E.  
$$E = 1 + 1/1! + 1/2! + 1/3! + \dots + 1/N!$$

# Exercícios

- 6) Faça um programa que, dado um conjunto de valores inteiros e positivos (fornecidos um a um pelo usuário), determine qual o menor valor do conjunto. O final do conjunto de valores é conhecido através do valor zero, que não deve ser considerado.
- 7) A conversão de graus Fahrenheit para Centígrados é obtida pela fórmula  $C = 5/9(F - 32)$ . Escreva um programa que calcule e escreva uma tabela de graus centígrados em função de graus Fahrenheit que variem de 50 a 150 de 1 em 1.
- 8) Elabore um programa que calcule  $N!$  (fatorial de  $N$ ), sendo que o valor inteiro de  $N$  é fornecido pelo usuário. Sabendo que:  
$$N! = N \times (N-1) \times (N-2) \times \dots \times 3 \times 2 \times 1;$$
$$0! = 1, \text{ por definição.}$$
- 9) Fazer um programa para calcular e mostrar os  $N$  primeiros termos da série de Fibonacci. O número  $N$  é fornecido pelo usuário. A série de Fibonacci é gerada da seguinte forma:  
$$f_1 = f_2 = 1;$$
$$f_3 = f_1 + f_2 = 2;$$
$$f_4 = f_2 + f_3 = 3; \dots$$

O primeiro e segundo termos valem 1 e os seguintes são calculados somando os dois termos anteriores.