

# Comandos de repetição – com variável de controle

Roberto Rocha



# Tabuada!

# O que esse programa imprime?


Algoritmo “Repetição”

var

 i:inteiro

Início

  $i \leftarrow 1$

 enquanto ( $i \leq 5$ ) faça **Verdadeiro**

 escreva(i)

  $i \leftarrow i + 1$

 fimenquanto

fimalgoritmo

Variáveis	
i	1 2

1

# O que esse programa imprime?


Algoritmo “Repetição”

var

i:inteiro

Início

$i \leftarrow 1$

 enquanto ( $i \leq 5$ ) faça **Verdadeiro**

 escreva(i)

  $i \leftarrow i + 1$

 fimenquanto

fimalgoritmo

Variáveis		
i	2	3

1 2

# O que esse programa imprime?


Algoritmo “Repetição”

var

i:inteiro

Início

$i \leftarrow 1$

 enquanto ( $i \leq 5$ ) faça **Verdadeiro**

 escreva(i)

  $i \leftarrow i + 1$

 fimenquanto

fimalgoritmo

Variáveis	
i	3 4

1 2 3

# O que esse programa imprime?


Algoritmo “Repetição”

var

i:inteiro

Início

$i \leftarrow 1$

 enquanto ( $i \leq 5$ ) faça **Verdadeiro**

 escreva(i)

  $i \leftarrow i + 1$

 fimenquanto

fimalgoritmo

Variáveis	
i	4 5

1 2 3 4

# O que esse programa imprime?


Algoritmo “Repetição”

var

i:inteiro

Início

$i \leftarrow 1$

 enquanto ( $i \leq 5$ ) faça **Verdadeiro**

 escreva(i)

  $i \leftarrow i + 1$

 fimenquanto

fimalgoritmo

Variáveis	
i	5 6

1 2 3 4 5

# O que esse programa imprime?


Algoritmo “Repetição”

var

i:inteiro

Inicio

$i \leftarrow 1$

 enquanto ( $i \leq 5$ ) faça **Falso**

escreva(i)

$i \leftarrow i + 1$

fimenquanto

 fimalgoritmo

Variáveis	
i	6

1 2 3 4 5



# Algoritmos – repetição

## Comando para - faça

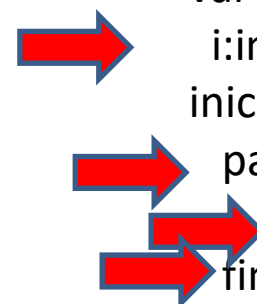
O comando de repetição para- faça constitui-se de um contador que ao chegar ao comando fim para automaticamente incrementa este contador com o valor estipulado em passo

### Sintaxe Geral:

<variável de controle> : inteiro

para <variável> de <valor-inicial> ate <valor-limite> [passo <incremento>] faca  
    <sequência-de-comandos>  
fimpara

Algoritmo “Repetição”  
var  
    i:inteiro  
inicio  
    para i de 1 ate 5 passo 1 faca  
        escreva(i)  
    fimpara  
fimalgoritmo



Variáveis	
i	1
1	

# Algoritmos – repetição

## Comando para - faça

O comando de repetição para- faça constitui-se de um contador que ao chegar ao comando fim para automaticamente incrementa este contador com o valor estipulado em passo

### Sintaxe Geral:

<variável de controle> : inteiro


para <variável> de <valor-inicial> ate <valor-limite> [passo <incremento>] faca  
    <sequência-de-comandos>  
fimpara

Algoritmo “Repetição”

var

  i:inteiro

inicio

 para i de 1 ate 5 passo 1 faca

 escreva(i)

 fimpara

fimalgoritmo

### Variáveis

i

2

1 2

# Algoritmos – repetição

## Comando para - faça

O comando de repetição para- faça constitui-se de um contador que ao chegar ao comando fim para automaticamente incrementa este contador com o valor estipulado em passo

### Sintaxe Geral:

<variável de controle> : inteiro

para <variável> de <valor-inicial> ate <valor-limite> [passo <incremento>] faca  
    <sequência-de-comandos>  
fimpara

Algoritmo “Repetição”

var

  i:inteiro

inicio

→ para i de 1 ate 5 passo 1 faca

→ escreva(i)

→ fimpara

fimalgoritmo

### Variáveis

i	3
---	---

1 2 3

# Algoritmos – repetição

## Comando para - faça

O comando de repetição para- faça constitui-se de um contador que ao chegar ao comando fim para automaticamente incrementa este contador com o valor estipulado em passo

### Sintaxe Geral:

<variável de controle> : inteiro

para <variável> de <valor-inicial> ate <valor-limite> [passo <incremento>] faca  
    <sequência-de-comandos>  
fimpara

Algoritmo “Repetição”

var

  i:inteiro

inicio

→ para i de 1 ate 5 passo 1 faca

→ escreva(i)

→ fimpara

fimalgoritmo

### Variáveis

i	4
---	---

1 2 3 4

# Algoritmos – repetição

## Comando para - faça

O comando de repetição para- faça constitui-se de um contador que ao chegar ao comando fim para automaticamente incrementa este contador com o valor estipulado em passo

### Sintaxe Geral:

<variável de controle> : inteiro


para <variável> de <valor-inicial> ate <valor-limite> [passo <incremento>] faca  
    <sequência-de-comandos>  
fimpara

Algoritmo “Repetição”

var

  i:inteiro

inicio

 para i de 1 ate 5 passo 1 faca

 escreva(i)

 fimpara

 finalgoritmo

### Variáveis

i	5
---	---

1 2 3 4 5

# Algoritmos – repetição

## Comando para - faça

Algoritmo “Repetição”

var

i:inteiro

Início

$i \leftarrow 1$

enquanto ( $i \leq 5$ ) faça

    escreva(i)

$i \leftarrow i + 1$

fimenquanto

fimalgoritmo

Algoritmo “Repetição”

var

i:inteiro

início

    para i de 1 ate 5 passo 1 faça

        escreva(i)

    fimpara

fimalgoritmo

# Algoritmos – repetição

## Comando para - faça

Exercício: - utilize o comando de repetição **PARA**

Faça um programa que imprima a tabuada de um número lido.

# Exercício de fixação:

Faça um programa que imprima a tabuada de um número lido.

Entrada

Número

Processamento

Saída

Tabuada do 5				
1	X	5	=	5
2	X	5	=	10
3	X	5	=	15
4	X	5	=	20
5	X	5	=	25
6	X	5	=	30
7	X	5	=	35
8	X	5	=	40
9	X	5	=	45
10	X	5	=	50



# Programa que imprima a tabuada de um número lido.

Algoritmo “tabuada”

//Programa que imprima a tabuada de um número lido.

var

n,i:inteiro

Inicio

leia(n)

escreva(“Tabuada do “,n)

para i de 1 ate 10 passo 1 faca

escreva(i,”x”,n,”=“,i\*n)

fimpara

fimalgoritmo

Tabuada do 5				
1	X	5	=	5
2	X	5	=	10
3	X	5	=	15
4	X	5	=	20
5	X	5	=	25
6	X	5	=	30
7	X	5	=	35
8	X	5	=	40
9	X	5	=	45
10	X	5	=	50

# Algoritmo x Java

Algoritmo	Java
Comando repetição para..faça	
<u>para</u> <variável> <u>de</u> <valor-inicial> <u>ate</u> <valor-limite> [ <u>passo</u> <incremento>] <u>faca</u> c1 c2 c3 fimpara	for(variável=valor_inicial;condição;incremento) { c1; c2; c3; }
Exemplos	
para a de 1 ate 5 passo 1 faca escreva(a) fimpara	for(a=1;a<=5;a=a+1) { printf("%d\n",a); }

## Programa que imprima a tabuada de um número lido.

Algoritmo “tabuada”

//Programa que imprima a tabuada de um número lido.

var

n,i:inteiro

Inicio

leia(n)

escreva(“Tabuada do “,n)

para i de 1 ate 10 passo 1 faca

escreva(i,”x”,n,”=”,i\*n)

fimpara

fimalgoritmo

```
import java.util.Scanner;
public class Tabuada
{
    //Programa que imprima a tabuada de um número lido.
    public static void main (String[] args)
    {
        Scanner leia = new Scanner (System.in);
        int n,i;
        System.out.println("Informe um numero:");
        n= leia.nextInt();
        System.out.println("Tabuada do "+n);
        for (i=1;i<=10;i=i+1)
        {
            System.out.printf("%2d x %2d = %3d\n",i,n,i*n);
        }
    }
}
```

Teste

```
----jGRASP exec: java Tabuada
Informe um numero:
5
Tabuada do 5
1 x 5 = 5
2 x 5 = 10
3 x 5 = 15
4 x 5 = 20
5 x 5 = 25
6 x 5 = 30
7 x 5 = 35
8 x 5 = 40
9 x 5 = 45
10 x 5 = 50

----jGRASP: operation complete.
```

# Exercício de fixação:

2 – Faça um programa que imprima a tabuada dos números de 1 a 9.

## Entrada

Não precisamos de mais nenhuma entrada pois o próprio programa irá gerar todos os dados

## Processamento

## Saída

Para cada número irá gerar a tabuada completa.

## 2 – Faça um programa que imprima a tabuada dos números de 1 a 9.

Algoritmo “tabuada”

//Imprima a tabuada dos números de 1 a 9.

var

i,j:inteiro

Inicio

para i de 1 ate 9 passo 1 faca

    escreva(“Tabuada do “,i)

    para j de 1 ate 10 passo 1 faca

        escreva(i,”x”,j,”=”,i\*j)

    fimpara

  fimpara

fimalgoritmo

## 2 – Faça um programa que imprima a tabuada dos números de 1 a 9.

Algoritmo “tabuada”

//Imprima a tabuada dos números de 1 a 9.

var

i,j:inteiro

Início

para i de 1 ate 9 passo 1 faca

escreva(“Tabuada do “,i)

para j de 1 ate 10 passo 1 faca

escreva(i,”x”,j,”=”,i\*j)

fimpara

fimpara

finalgoritmo

```
public class Tabuada1a9
{
    //Imprima a tabuada dos números de 1 a 9.
    public static void main (String[] args)
    {
        int i,j;
        for (i=1;i<=9;i=i+1)
        {
            System.out.println("Tabuada do "+i);
            for (j=1;j<=10;j=j+1)
            {
                System.out.printf("%2d x %2d = %3d\n",i,j,i*j);
            }
        }
    }
}
```

Teste:

----jGRASP exec: java Tabuada1a9

Tabuada do 1

1 x 1 = 1  
1 x 2 = 2  
1 x 3 = 3  
1 x 4 = 4  
1 x 5 = 5  
1 x 6 = 6  
1 x 7 = 7  
1 x 8 = 8  
1 x 9 = 9  
1 x 10 = 10

Tabuada do 2

2 x 1 = 2  
2 x 2 = 4  
2 x 3 = 6  
2 x 4 = 8  
2 x 5 = 10  
2 x 6 = 12  
2 x 7 = 14  
2 x 8 = 16  
2 x 9 = 18  
2 x 10 = 20

Tabuada do 3

3 x 1 = 3  
3 x 2 = 6  
3 x 3 = 9  
3 x 4 = 12  
3 x 5 = 15  
3 x 6 = 18  
3 x 7 = 21  
3 x 8 = 24  
3 x 9 = 27  
3 x 10 = 30

Tabuada do 4

4 x 1 = 4  
4 x 2 = 8  
4 x 3 = 12  
4 x 4 = 16  
4 x 5 = 20  
4 x 6 = 24  
4 x 7 = 28  
4 x 8 = 32  
4 x 9 = 36  
4 x 10 = 40

Tabuada do 5

5 x 1 = 5  
5 x 2 = 10  
5 x 3 = 15  
5 x 4 = 20  
5 x 5 = 25  
5 x 6 = 30  
5 x 7 = 35  
5 x 8 = 40  
5 x 9 = 45  
5 x 10 = 50

Tabuada do 6

6 x 1 = 6  
6 x 2 = 12  
6 x 3 = 18  
6 x 4 = 24  
6 x 5 = 30  
6 x 6 = 36  
6 x 7 = 42  
6 x 8 = 48  
6 x 9 = 54  
6 x 10 = 60

Tabuada do 7

7 x 1 = 7  
7 x 2 = 14  
7 x 3 = 21  
7 x 4 = 28  
7 x 5 = 35  
7 x 6 = 42  
7 x 7 = 49  
7 x 8 = 56  
7 x 9 = 63  
7 x 10 = 70

Tabuada do 8

8 x 1 = 8  
8 x 2 = 16  
8 x 3 = 24  
8 x 4 = 32  
8 x 5 = 40  
8 x 6 = 48  
8 x 7 = 56  
8 x 8 = 64  
8 x 9 = 72  
8 x 10 = 80

Tabuada do 9

9 x 1 = 9  
9 x 2 = 18  
9 x 3 = 27  
9 x 4 = 36  
9 x 5 = 45  
9 x 6 = 54  
9 x 7 = 63  
9 x 8 = 72  
9 x 9 = 81  
9 x 10 = 90

----jGRASP: operation complete.

# Repetição com teste no final

Estrutura de repetição que permite a execução de pelo menos uma vez dos comandos. Podemos utilizar, por exemplo, em menus

faça

Mostre menu

Escolha a opção

enquanto opção <> sair

# Algoritmo x Java

Algoritmo	Java
<b>Comando repetição</b>	
Faca c1 c2 c3 Enquanto (condição(	do { c1; c2; c3; } while(condição)
<b>Exemplos</b>	
faca escreva(a) a←a+1 enquanto (a<=5)	do { printf("%d\n",a); a=a+1; } while (a<=5)

Tabela de operadores relacionais		
Algoritmo		Java
Operador	Descrição	Operador
=	Igual a	==
>	Maior que	>
<	Menor que	<
>=	Maior ou igual a	>=
<=	Menor ou igual a	<=
<>	Diferente de	!=



Faça um menu para mostrar 3 opções :

1 – tabuada de um número

2 – tabuada dos números de 1 a 9

9 – fim do programa

Algoritmo “menu”

// mostrar um menu com o comando faça ... enquanto

var

opcao:inteiro

Inicio

faça

    escreva(“opções:”)

    escreva(“1 – tabuada de um número”)

    escreva(“2 – tabuada de 1 a 9”)

    escreva(“9 – fim do programa”)

    leia (opcao)

    escolha (opcao)

        caso 1

            “tabuada de um número”

        caso 2

            “tabuada de 1 a 9”

    fimescolha

    enquanto (opcao<>9)

fimalgoritmo

```
import java.util.Scanner;
public class ExerciciosRepeticaoP2
{
    /// Unindo os 3 exercicios e criando um menu
    public static void main (String[] args)
    {
        Scanner leia = new Scanner (System.in);
        int op;
        do {
            menu();
            op=leia.nextInt();
            switch (op)
            {
                case 1:
                    tabuada();
                    break;
                case 2:
                    tabuada1a9();
                    break;
            }
        }while (op!=9);
    }
    public static void menu()
    {
    }
    public static void tabuada ()
    {
    }
    public static void tabuada1a9 ()
    {
    }
}
```

```
public static void menu()
{
    System.out.println("Escolha:");
    System.out.println("1 - Tabuada de um numero");
    System.out.println("2 - Tabuada de 1 a 9");
    System.out.println("9 - Sair do programa");
}

public static void tabuada ()
{
    Scanner leia = new Scanner (System.in);
    int n,i;
    System.out.println("Informe um numero:");
    n= leia.nextInt();
    System.out.println("Tabuada do "+n);
    for (i=1;i<=10;i=i+1)
    {
        System.out.printf("%2d x %2d = %3d\n",i,n,i*n);
    }
}

public static void tabuada1a9 ()
{
    int i,j;
    for (i=1;i<=9;i=i+1)
    {
        System.out.println("Tabuada do "+i);
        for (j=1;j<=10;j=j+1)
        {
            System.out.printf("%2d x %2d = %3d\n",i,j,i*j);
        }
    }
}
```

# Exercícios de fixação:

1. Escrever a sequência de 1 até 10 - Utilize a estrutura de repetição para....faça
2. Escrever a sequência de 10 até 1 - Utilize a estrutura de repetição para....faça
3. Leia um número e imprima os números ímpares de 1 até esse número. - Utilize a estrutura de repetição para....faça
4. Escrever um algoritmo, para calcular e imprimir o fatorial de um número lido do teclado.  
Ex. Fatorial de  $5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$ . - Utilize a estrutura de repetição para....faça

5. A série de Fibonacci é formada pela sequência:

1, 1, 2, 3, 5, 8, 13, 21, 34, ...

Escreva um algoritmo que peça um número N maior que 2. Gere e imprima a série de 1 até este enésimo termo. Utilize a estrutura de repetição para....faça

6. O número 3025 possui a seguinte característica:

$$30 + 25 = 55$$

$$55^2 = 3025$$

Quantos e quais são os números de 4 dígitos possuem essa característica?

Utilize a estrutura de repetição para....faça

7. construa um Menu para chamar os programas acima. Utilize a estrutura de repetição faça ... enquanto



**PUC Minas**  
**Virtual**