



ESTRUTURA DE REPETIÇÃO PARA

BRIAN NEUKAMP

As Estruturas de REPETIÇÃO

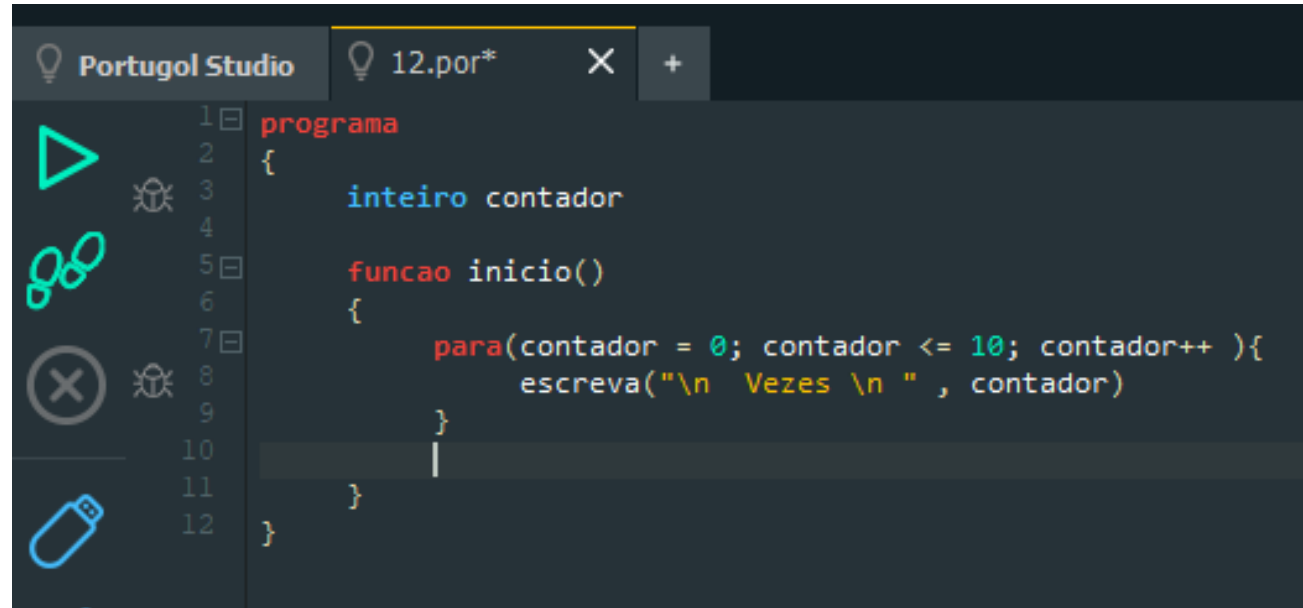
- ▶ **O que é estrutura de repetição?**
- ▶ Dentro da lógica de programação é uma estrutura que permite executar mais de uma vez o mesmo comando ou conjunto de comandos, de acordo com uma condição ou com um contador.
- ▶ São utilizadas, por exemplo, para repetir ações semelhantes que são executadas para todos os elementos de uma lista de dados, ou simplesmente para repetir um mesmo processamento até que a condição seja satisfeita.
- ▶ Existem 4 estruturas de repetição básica para praticamente todas as linguagens de programação, seja C ou javascript.

As Estruturas de REPETIÇÃO PARA

- ▶ O comando **PARA** permite construir estruturas de loop para casos onde se **conhece de antemão o número de repetições que devem ser realizadas** (ou seja, número finito de laços). Por exemplo, quando sabemos de antemão que o laço deve se repetir 10 vezes; portanto, neste caso podemos usar a estrutura para.

As Estruturas de REPETIÇÃO PARA

Incrementado

A screenshot of the Portugol Studio IDE. The window title is 'Portugol Studio' and the file name is '12.por*'. The code is written in a dark-themed editor with line numbers 1 through 12 on the left. The code defines a program with a loop structure. On the left side of the editor, there are icons for execution: a green play button, a green infinity symbol, a grey 'X' in a circle, and a blue USB icon.

```
1 programa
2 {
3     inteiro contador
4
5     funcao inicio()
6     {
7         para(contador = 0; contador <= 10; contador++) {
8             escreva("\n Vezes \n ", contador)
9         }
10
11     }
12 }
```

As Estruturas de REPETIÇÃO PARA

Decrementado

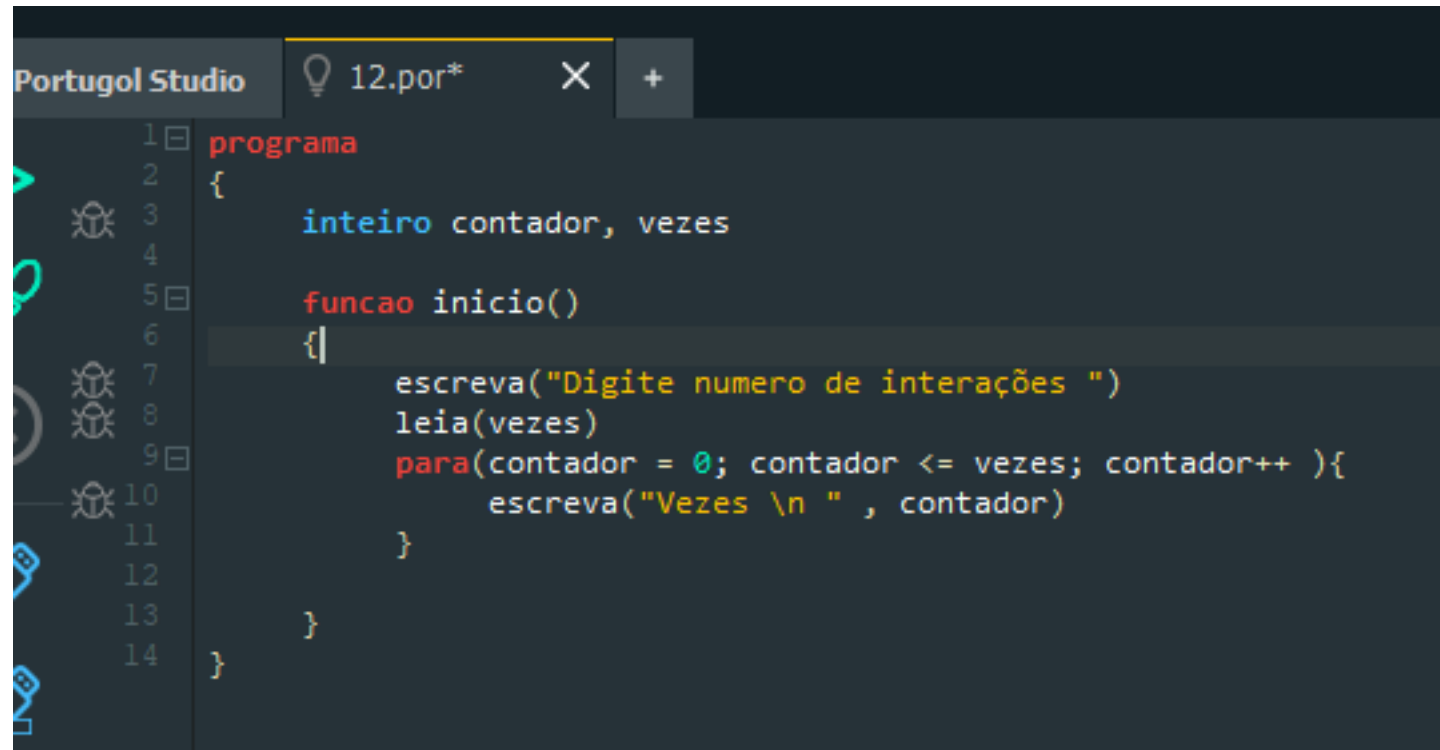


The screenshot shows a code editor window titled 'Portugol Studio' with a file named '12.por*'. The code is written in Portuguese and implements a loop that decrements a counter from 10 to 0. The code is as follows:

```
1 programa
2 {
3     inteiro contador
4
5     funcao inicio()
6     {
7         para(contador = 10; contador >= 0; contador--) {
8             escreva("\n Vezes \n ", contador)
9         }
10
11     }
12 }
```

As Estruturas de REPETIÇÃO PARA

Interagindo com
Usuário

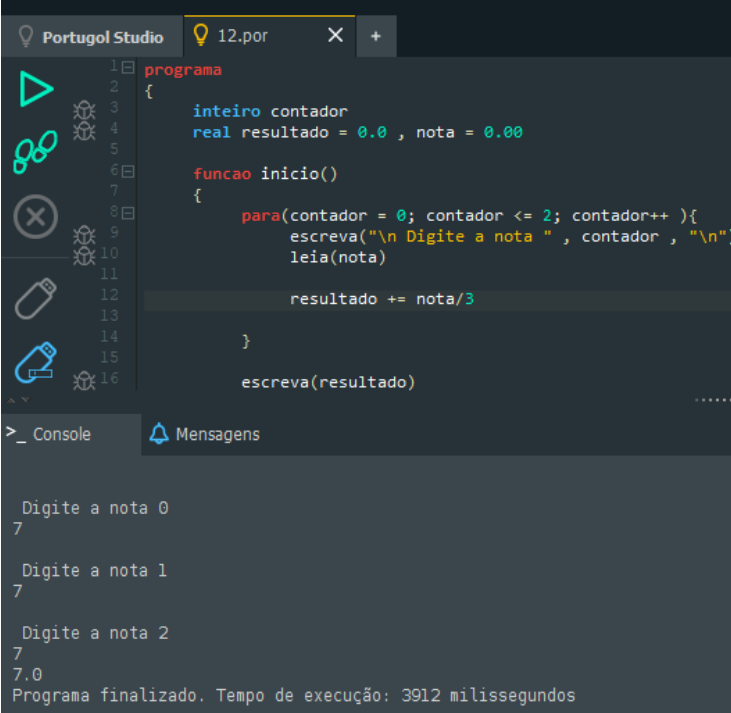


The screenshot shows the Portugol Studio IDE with a file named '12.por*'. The code is written in a structured programming style with line numbers on the left. It defines a 'programa' block containing a 'funcao inicio()' block. Inside the function, it declares 'inteiro contador, vezes'. It then prompts the user to enter the number of interactions, reads the input into 'vezes', and uses a 'para' loop to print the word 'Vezes' followed by the current value of 'contador' for each iteration from 0 to 'vezes'.

```
1 programa
2 {
3     inteiro contador, vezes
4
5     funcao inicio()
6     {
7         escreva("Digite numero de interações ")
8         leia(vezes)
9         para(contador = 0; contador <= vezes; contador++) {
10             escreva("Vezes \n " , contador)
11         }
12     }
13 }
14 }
```

As Estruturas de REPETIÇÃO PARA

Interagindo com
Usuário Somando
os valores e
realizando a
média



```
1 programa
2 {
3     inteiro contador
4     real resultado = 0.0 , nota = 0.00
5
6     funcao inicio()
7     {
8         para(contador = 0; contador <= 2; contador++){
9             escreva("\n Digite a nota " , contador , "\n")
10            leia(nota)
11
12            resultado += nota/3
13
14        }
15
16        escreva(resultado)
```

The screenshot shows the Portugol Studio interface. The main editor displays a program in Portugol syntax. The program defines a variable 'contador' of type 'inteiro' and 'resultado' of type 'real'. It then defines a function 'inicio()' which contains a 'para' loop that iterates from 0 to 2. Inside the loop, it prompts the user to enter a note ('Digite a nota'), reads the input ('leia(nota)'), and adds it to the 'resultado' variable (divided by 3). After the loop, it prints the final 'resultado'. The bottom panel shows the console output, which matches the program's execution: three prompts for notes, three inputs (0, 1, 2), and a final output of 7.0.

> _ Console

```
Digite a nota 0
7
Digite a nota 1
7
Digite a nota 2
7
7.0
Programa finalizado. Tempo de execução: 3912 milissegundos
```

As Estruturas de REPETIÇÃO PARA

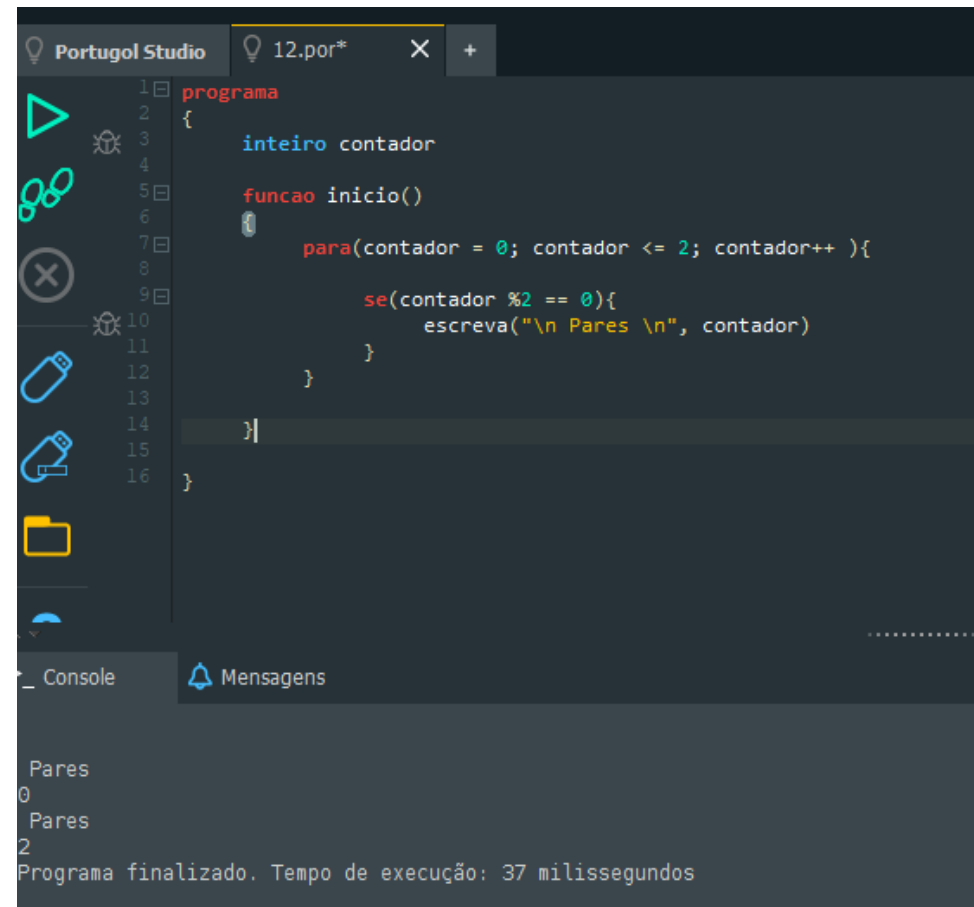
Interagindo com
Usuário
Multiplicando
valores



```
1 programa
2 {
3     inteiro contador
4     real resultado = 1.0 , nota = 0.00
5
6     funcao inicio()
7     {
8         para(contador = 0; contador <= 2; contador++) {
9             escreva("\n Digite um multiplicador " , contador , "\n")
10            leia(nota)
11
12            resultado *= nota
13
14        }
15
16        escreva(resultado)
17    }
18 }
19 }
```


As Estruturas de REPETIÇÃO PARA

Ímpares e Pares
com comando
PARA



```
1 programa
2 {
3     inteiro contador
4
5     funcao inicio()
6     {
7         para(contador = 0; contador <= 2; contador++) {
8
9             se(contador % 2 == 0) {
10                 escreva("\n Pares \n", contador)
11             }
12         }
13     }
14 }
15
16 }
```

Console

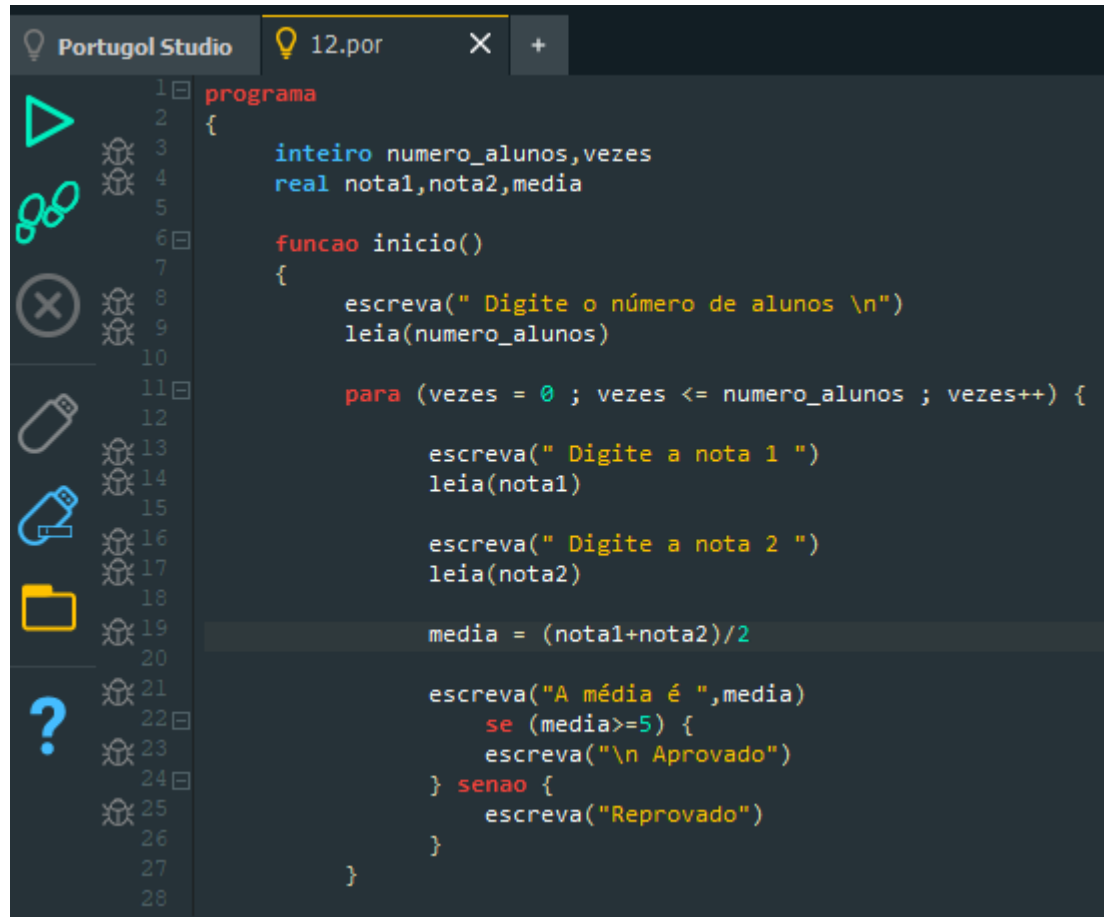
Mensagens

Pares
0
Pares
2
Programa finalizado. Tempo de execução: 37 milissegundos

As Estruturas de REPETIÇÃO PARA

Desafio em aula: Leia o número de alunos de uma sala. Faça um laço de repetição até atingir o número de alunos informados. Para cada aluno peça 2 notas, calcule a média, se a média for maior ou igual a 5, escreva “aprovado”, senão escreva “reprovado”

As Estruturas de REPETIÇÃO PARA

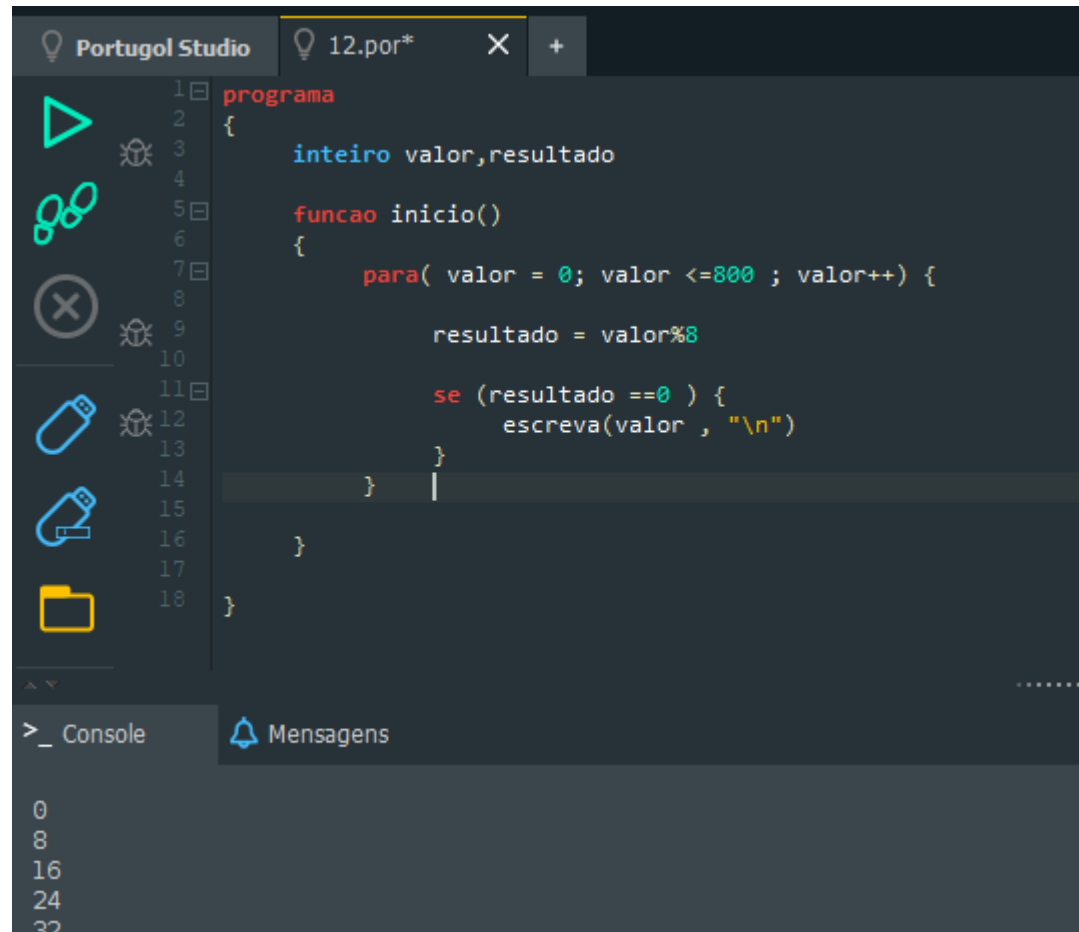
The image shows a screenshot of the Portugol Studio IDE. The window title is 'Portugol Studio' and the file name is '12.por'. The code is written in a structured, pseudo-code style. It starts with a 'programa' block containing variable declarations for 'numero_alunos', 'vezes', 'nota1', 'nota2', and 'media'. A 'funcao inicio()' block contains the main logic. A 'para' loop is used to iterate from 0 to 'numero_alunos', prompting the user to enter two notes ('nota1' and 'nota2') for each student. After the loop, the average ('media') is calculated. A conditional block ('se') checks if the average is greater than or equal to 5, printing 'Aprovado' or 'Reprovado' accordingly. The left sidebar contains icons for running, debugging, and other IDE functions.

```
1 programa
2 {
3     inteiro numero_alunos, vezes
4     real nota1, nota2, media
5
6     funcao inicio()
7     {
8         escreva(" Digite o número de alunos \n")
9         leia(numero_alunos)
10
11         para (vezes = 0 ; vezes <= numero_alunos ; vezes++) {
12
13             escreva(" Digite a nota 1 ")
14             leia(nota1)
15
16             escreva(" Digite a nota 2 ")
17             leia(nota2)
18
19             media = (nota1+nota2)/2
20
21             escreva("A média é ", media)
22             se (media >= 5) {
23                 escreva("\n Aprovado")
24             } senao {
25                 escreva("Reprovado")
26             }
27         }
28     }
```

As Estruturas de REPETIÇÃO PARA

Mostre todos os números divisíveis por 8 entre 0 e 800 com o comando para

As Estruturas de REPETIÇÃO PARA



The screenshot shows the Portugol Studio IDE with a file named 12.por*. The code is written in a structured programming style with icons on the left. The code defines a program with a loop that calculates the remainder of values from 0 to 800 divided by 8, printing the result when it is 0.

```
1 programa
2 {
3     inteiro valor,resultado
4
5     funcao inicio()
6     {
7         para( valor = 0; valor <=800 ; valor++) {
8
9             resultado = valor%8
10
11             se (resultado ==0 ) {
12                 escreva(valor , "\n")
13             }
14         }
15     }
16 }
17
18 }
```

The bottom of the window shows a Console and Mensagens panel. The Console displays the output of the program, showing the values 0, 8, 16, 24, and 32, which are the multiples of 8 within the range.

As Estruturas de REPETIÇÃO PARA

Leia 8 valores e determine qual é maior deles utilizando comando para

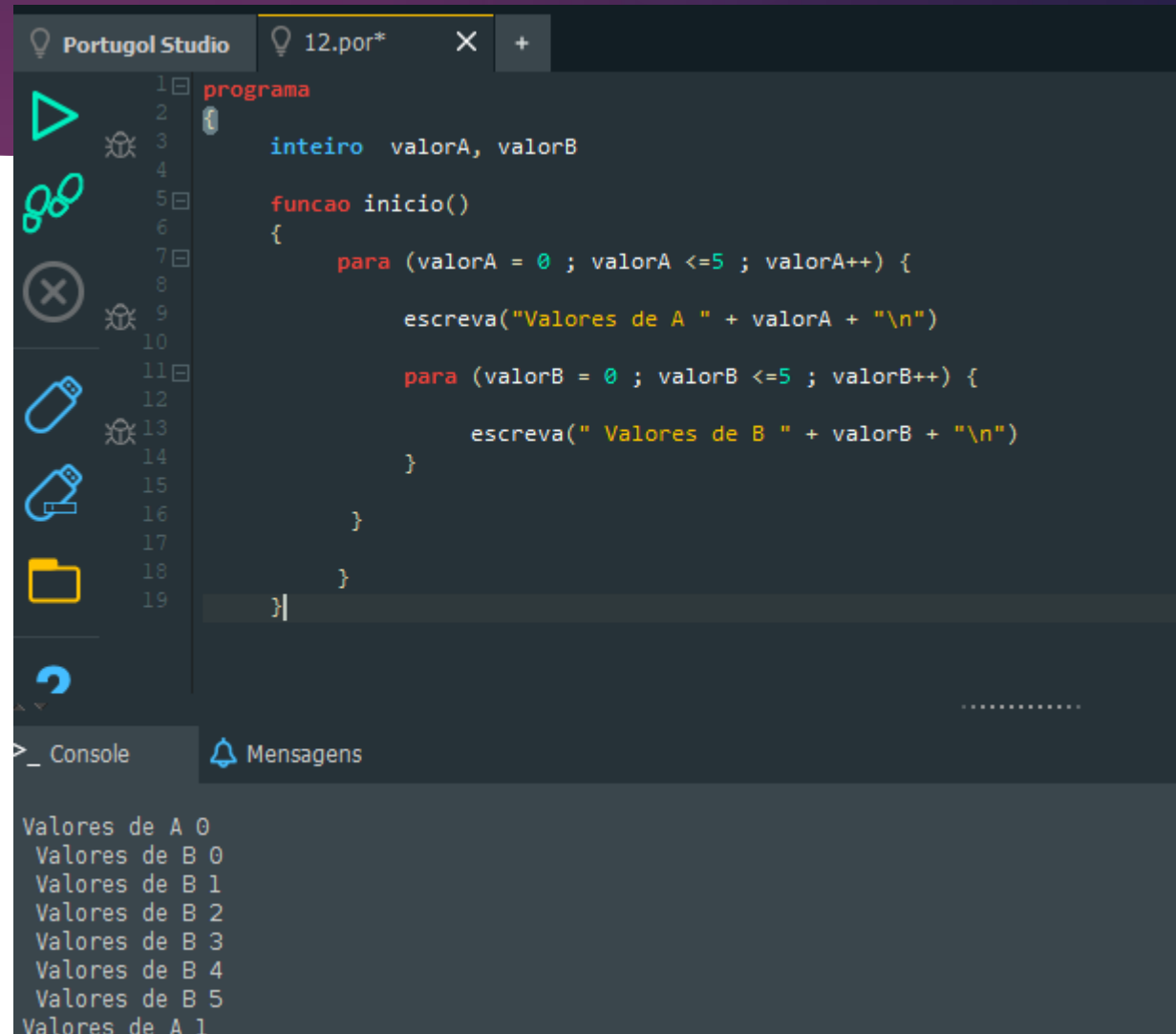
As Estruturas de REPETIÇÃO PARA

```
Portugol Studio 12.por X +
1 programa
2 {
3     inteiro valor,vezes,maior
4
5     funcao inicio()
6     {
7         para (vezes = 0 ; vezes <=8 ; vezes++) {
8
9             escreva("Digite um valor \n")
10            leia(valor)
11
12            se (vezes == 1){
13                maior = valor
14            }
15            senao se (valor>maior) {
16                maior = valor
17            }
18
19        }
20
21        escreva("O maior valor é ",maior)
22
23    }
24 }
```

As Estruturas de REPETIÇÃO PARA

Também temos a repetição **encadeada**
no qual iremos trabalhar na próxima aula
com vetores ou (Matrizes)

As Estruturas de REPETIÇÃO PARA ENCADEADA EX:



The screenshot displays the Portugol Studio IDE with a file named '12.por*'. The code defines a program with two nested loops. The first loop iterates over 'valorA' from 0 to 5, and the second loop iterates over 'valorB' from 0 to 5. The output in the console shows the values of A and B during execution.

```
1 programa
2
3 inteiro valorA, valorB
4
5 funcao inicio()
6 {
7     para (valorA = 0 ; valorA <=5 ; valorA++) {
8
9         escreva("Valores de A " + valorA + "\n")
10
11         para (valorB = 0 ; valorB <=5 ; valorB++) {
12
13             escreva("Valores de B " + valorB + "\n")
14         }
15     }
16 }
17
18
19 }
```

Console Output:

```
> _ Console Mensagens
Valores de A 0
Valores de B 0
Valores de B 1
Valores de B 2
Valores de B 3
Valores de B 4
Valores de B 5
Valores de A 1
```

Exercitando

- ▶ Mostre os números de 200 até 90 na tela. **Comando para**
- ▶ Leia cinco valores com o comando para. No final mostre a soma dos 5 valores. **comando para**
- ▶ Leia 5 valores, faça a multiplicação dos valores lidos e mostre o resultado. **comando para**
- ▶ Desenvolva um algoritmo que mostre os números ímpares de 5 até 30. **comando para**
- ▶ Mostre a tabuada do 5 utilizando o **comando para**