



**INSTITUTO
FEDERAL**

Paraná

Campus
Foz do Iguaçu

Algoritmos e Programação

Prof. Daniel Di Domenico

daniel.domenico@ifpr.edu.br

Estruturas de repetição
REPITA

Estruturas de repetição

- Na aula anterior, trabalhamos com a estrutura de repetição **ENQUANTO**
 - Utilizado para repetição de trechos de código
 - Exemplo em Portugol:

```
enquanto <condicao> faça  
  inicio  
    //Bloco de comandos...  
  fim
```

Comando REPITA

- **Outra estrutura de repetição**
 - Também utilizada para repetir trechos de código
- Similar ao ENQUANTO
- Diferença:
 - **ENQUANTO**: a condição é avaliada **antes** de iniciar o trecho de repetição
 - O trecho de código da repetição **pode não ser executado**
 - **REPITA**: a condição é avaliada **após** o trecho da repetição
 - O trecho de código da repetição é **executado ao menos uma vez**

Comando REPITA

- Em Portugal:

```
repita  
    //Bloco de comandos...  
ate <condicao>
```

Continua repetindo
enquanto a condição
for **FALSA**

- Exemplo:

```
num <- 1  
repita  
    //Repete o comando 10 vezes  
    num <- num + 1  
ate num = 10
```

Estruturas de repetição

- **Problema:** elabore um programa que leia repetidamente uma **senha**, até que ela esteja correta. Quando isso acontecer, imprimir a mensagem "Seja bem vindo!".
- **DICA:** a condição para término da repetição está relacionada à entrada do programa.

Estruturas de repetição

- Programa em Portugol com **ENQUANTO**:

```
algoritmo
declare
    senha literal

senha <- ""
enquanto senha <> "algoritmo" faca
inicio
    escreva "Digite a senha: "
    leia senha
fim

escreva "Bem vindo!"

fim_algoritmo
```

Requer inicialização
da senha antes da repetição

Estruturas de repetição

- Programa em Portugol com **REPITA**:

```
algoritmo  
declare  
    senha literal  
  
repita  
    escreva "Digite a senha: "  
    leia senha  
ate senha = "algoritmo"  
  
escreva "Bem vindo!"  
  
fim_algoritmo
```

NÃO requer inicialização
da senha ante da repetição

Estruturas de repetição

- Programa em Portugol com **REPITA**:

```
algoritmo  
declare  
    senha literal  
  
repita  
    escreva "Digite a senha: "  
    leia senha  
ate senha = "algoritmo"  
  
escreva "Bem vindo!"  
  
fim_algoritmo
```

Término da repetição:
senha = "algoritmo"

Comando REPITA

- **ATENÇÃO** a condição !!!
- Para usar corretamente esta estrutura de repetição, é importante formular uma condição adequada.
- Se a condição do **ate** for sempre **falsa**, a repetição nunca vai **parar** (laço infinito).

Comando REPITA

- Boas práticas
 - Atenção ao **recuo** (endentação!!!!)
 - Não esquecer de recuar à direita

```
repita  
    //Bloco de comandos...  
ate <condicao>
```

CERTO!

```
repita  
//Bloco de comandos...  
Ate <condicao>
```

MÁ PRÁTICA!
Deveria ter recuo!

Estruturas de repetição

- **Problema:** elabore um programa que leia 10 valores e imprima **o maior deles** no final.
- Usar uma variável para guardar o maior número lido. Essa variável é testada e possivelmente atualizada a cada repetição.

Estruturas de repetição

- Programa em Portugol com **ENQUANTO** (*declare omitido*):

```
n <- 0

escreva "Digite um valor: "
leia maior

enquanto n < 9 faça
inicio
    escreva "Digite um valor: "
    leia valor
    se valor > maior entao
        maior <- valor
    n <- n + 1
fim

escreva "Maior valor = ", maior
```

Repete enquanto a
condição for **VERDADEIRA**

Estruturas de repetição

- Programa em Portugol com **ENQUANTO** (*declare omitido*):

```
n <- 0

escreva "Digite um valor: "
leia maior

enquanto n < 9 faça
inicio
    escreva "Digite um valor: "
    leia valor
    se valor > maior entao
        maior <- valor
    n <- n + 1
fim

escreva "Maior valor = ", maior
```

Repete enquanto a condição for **VERDADEIRA**

Quando a condição for **FALSA**, o laço é encerrado

Estruturas de repetição

- Programa em Portugol com **REPITA** (*declare omitido*):

```
n <- 0

escreva "Digite um valor: "
leia maior

repita
    escreva "Digite um valor: "
    leia valor
    se valor > maior entao
        maior <- valor
    n <- n + 1
ate n >= 10

escreva "Maior valor = ", maior
```

Repete **até** que a
condição for **FALSA**

Estruturas de repetição

- Programa em Portugol com **REPITA** (*declare omitido*):

```
n <- 0

escreva "Digite um valor: "
leia maior

repita
    escreva "Digite um valor: "
    leia valor
    se valor > maior entao
        maior <- valor
    n <- n + 1
ate n >= 10

escreva "Maior valor = ", maior
```

Repete **até** que a condição for **FALSA**

Quando a condição for **VERDADEIRA**, o laço é encerrado

ENQUANTO X REPITA?

- Diferenças:
 - Quantidade de vezes:
 - **ENQUANTO**: possivelmente uma vez
 - **REPITA**: no mínimo uma vez
 - Condição:
 - **ENQUANTO**: avaliada antes do bloco de comandos, devendo ser **VERDADEIRA** para iniciar/manter as repetições
 - **REPITA**: avaliada ao final do bloco de comandos, devendo ser **FALSA** para manter as repetições
 - Inicialização:
 - **ENQUANTO**: requer inicialização da(s) variável(is) da condição antes de iniciar o laço
 - **REPITA**: a(s) variável(is) podem ser inicializadas dentro do bloco de comandos da repetição
 - Serão avaliadas apenas no final do bloco (comando **ate**)

Qual estrutura utilizar?

- Perguntas para escolher a estrutura:
 - Minha repetição precisa executar ao menos uma vez?
 - Prefiro utilizar ENQUANTO ou REPITA?
 - Como mostrado, utilizando ambas estruturas pode-se implementar a solução necessária para o algoritmo

Exercícios (utilizar PARA)

- Faça um programa que calcule a soma dos 100 primeiros números naturais(1 a 100):
 - $1 + 2 + 3 + 4 + 5 + \dots + 99 + 100$
- Faça um programa que solicite números até que o usuário digite 0. Após isso, informe a quantidade de números pares e ímpares lidos
 - **Dica:** utilizar a função `resto(num_lido, 2)` para verificar se um número é par