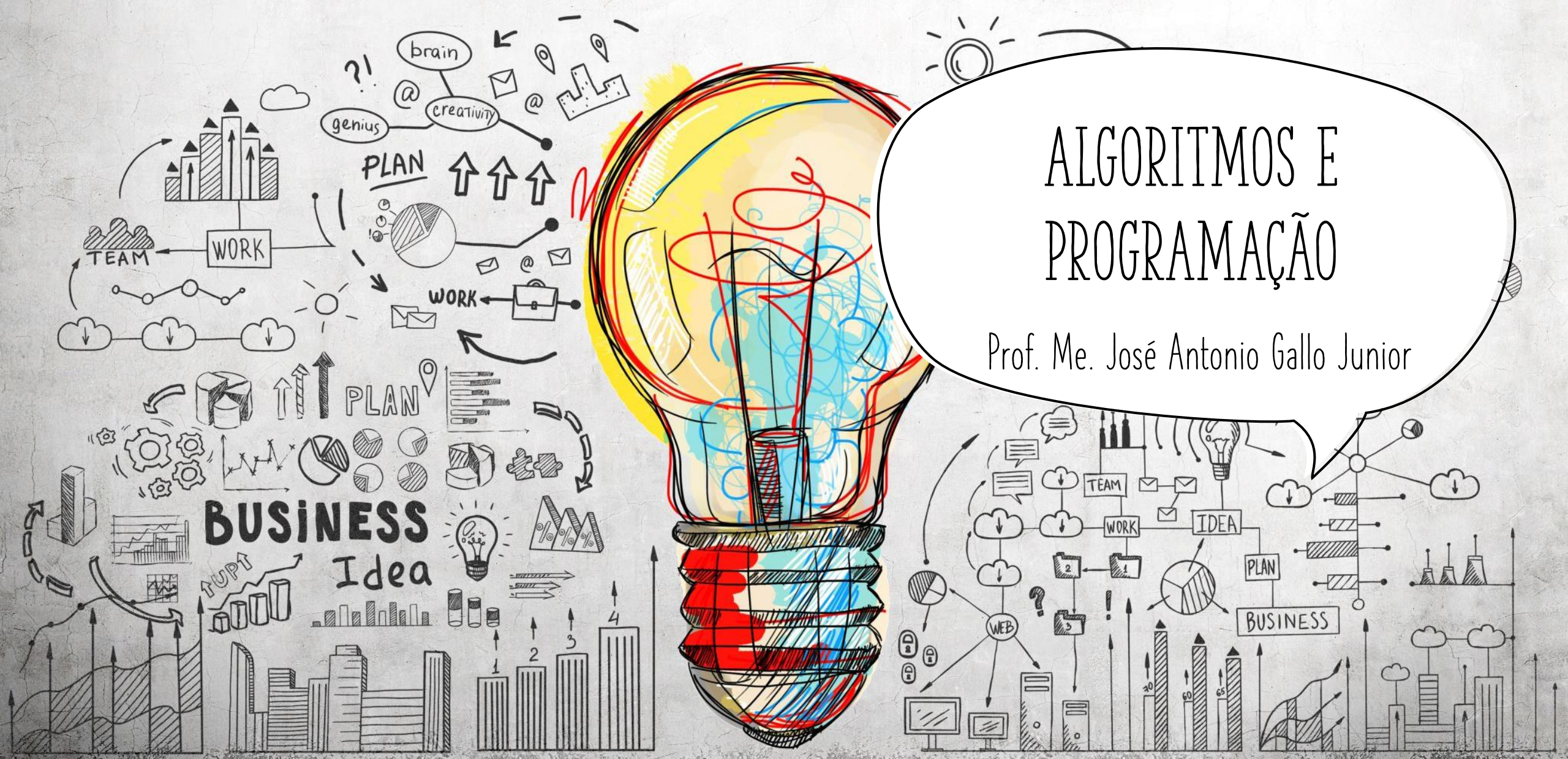


ALGORITMOS E PROGRAMAÇÃO

Prof. Me. José Antonio Gallo Junior



ESTRUTURAS DE CONTROLE

Em alguns casos, precisamos que o programa tome decisões ou repita ações com base nos dados recebidos.

Sem estruturas de controle, o programa executa os comandos na ordem em que aparecem, de cima para baixo.

As estruturas de controle permitem mudar a ordem de execução ou repetir instruções, tornando possível resolver problemas mais complexos, o que por sua vez, torna o programa mais inteligente e flexível.

ESTRUTURAS DE CONTROLE - LAÇOS DE REPETIÇÃO

Existem problemas que precisam repetir ações várias vezes.

Por exemplo: calcular a média de um aluno é simples. Mas se quisermos calcular a média de toda a turma?

Uma solução seria repetir o mesmo cálculo para cada aluno, mas isso daria muito trabalho.

E se ainda fosse necessário mostrar a maior e a menor média da turma?

Para esse tipo de situação usamos os laços de repetição.

ESTRUTURAS DE CONTROLE - LAÇOS DE REPETIÇÃO

Um laço de repetição faz o programa executar um conjunto de comandos várias vezes, até que uma condição seja atendida.

No Portugol existem três tipos de laços de repetição:

- Pré-testado - verifica a condição antes de executar.
- Pós-testado - executa primeiro e verifica a condição depois.
- Com variável de controle - repete um número conhecido de vezes.

LAÇO DE REPETIÇÃO - ENQUANTO (PRÉ-TESTADO)

Imagine que estamos criando um jogo da velha.

O jogo deve continuar **enquanto** existirem espaços livres no tabuleiro.

Como fazer o algoritmo repetir esse comportamento?

A resposta é simples: usamos o comando **enquanto**.

Ele serve para **executar** um conjunto de comandos enquanto uma condição for verdadeira.

LAÇO DE REPETIÇÃO - ENQUANTO (PRÉ-TESTADO)

Como usar:

- Escrevemos a palavra reservada `enquanto`.
- Colocamos a `condição` a ser testada entre parênteses.
- Entre chaves, colocamos as `instruções` que devem se repetir.

LAÇO DE REPETIÇÃO - ENQUANTO (PRÉ-TESTADO)

Sintaxe:

```
logico condicao = verdadeiro
enquanto (condicao)
{
    // Executa as instruções enquanto a condição for verdadeira
}
```

LAÇO DE REPETIÇÃO - ENQUANTO (PRÉ-TESTADO)

Exemplo:

```
cadeia parar = 'N'
enquanto (parar != 'S') {
    escreva("Oi Mundo!!!\n")
    escreva("Deseja parar o laço? (S/N): ")
    leia(parar)
}
```


LAÇO DE REPETIÇÃO - FAÇA-ENQUANTO (PÓS-TESTADO)

Em alguns casos, precisamos que o programa execute **pelo menos uma vez** um conjunto de comandos, e só depois verifique se deve continuar repetindo.

Para isso usamos o laço **faça-enquanto**.

Ele funciona de forma parecida com o **enquanto**, mas a diferença é que o teste lógico é feito no final.

Assim, as instruções dentro do laço são sempre executadas pelo menos uma vez.

LAÇO DE REPETIÇÃO - FAÇA-ENQUANTO (PÓS-TESTADO)

Como usar:

- Escrevemos a palavra reservada `faça`.
- Entre chaves ficam as instruções que serão executadas.
- Depois colocamos a palavra reservada `enquanto`, seguida da condição entre parênteses.

LAÇO DE REPETIÇÃO - FAÇA-ENQUANTO (PÓS-TESTADO)

Sintaxe:

```
logico condicao = verdadeiro  
faca  
{  
    // Executa a instrução uma vez e repete enquanto a condição for verdadeira  
} enquanto (condicao)
```

LAÇO DE REPETIÇÃO - FAÇA-ENQUANTO (PÓS-TESTADO)

Exemplo:

```
real lado  
faca {  
    escreva("Informe o valor do lado do quadrado: ")  
    leia(lado)  
} enquanto (lado <= 0)  
escreva("A área do quadrado é ", lado * lado)
```

LAÇO DE REPETIÇÃO - PARA (COM VARIÁVEL DE CONTROLE)

Às vezes precisamos repetir um comando um número exato de vezes.

Por exemplo: pedir ao usuário que digite 10 valores.

Poderíamos usar o comando `leia` 10 vezes, mas se fossem 100 valores, isso se tornaria cansativo e pouco prático.

Para resolver isso, usamos o laço de repetição com `variável de controle`, chamado `para` no `Portugol`.

Esse laço é útil porque já tem um `contador automático` que controla quantas vezes o bloco será executado.

LAÇO DE REPETIÇÃO - PARA (COM VARIÁVEL DE CONTROLE)

Assim, evitamos erros comuns, como esquecer de atualizar o contador manualmente.

O laço **para** possui três partes:

1. Inicialização da variável de controle (por onde o contador começa).
2. Condição final (até onde o contador vai).
3. Incremento (como o contador será alterado a cada repetição).

LAÇO DE REPETIÇÃO - PARA (COM VARIÁVEL DE CONTROLE)

Como usar:

- Palavra reservada `para`.
- Entre parênteses:
 - inicialização da variável de controle;
 - condição final;
 - incremento.
- Entre chaves ficam as `instruções` a serem executadas.

LAÇO DE REPETIÇÃO - PARA (COM VARIÁVEL DE CONTROLE)

Exemplo:

```
inteiro num  
escreva("Digite um número: ")  
leia(num)  
para (inteiro i = 0; i <= 10; i++)  
{  
    escreva(num, " x ", i, " = ", num * i, "\n", )  
}
```



VAMOS PRATICAR!?!