

Estruturas de Repetição

Estruturas de Repetição

- Diversos problemas somente podem ser resolvidos por um computador se o resultado de pequenas computações forem acumulados. Exemplo:
 - Calcular o fatorial de um número.
 - Verificar a nota de 30 alunos e informar sua situação (Aprovado ou Reprovado).
- Precisamos de mecanismos que nos permitam requisitar que um **conjunto de instruções seja repetidamente executado, até que uma determinada condição seja alcançada.**
- Repetições são programadas através da construção de laços (ou ciclos).

Estruturas de Repetição

- **Definição:** é uma estrutura que permite a execução de um trecho de código por mais de uma vez.
- Tipos de estruturas de repetição:
 - Com quantidade indeterminada de repetições;
 - Com quantidade determinada de repetições.
- Estruturas com quantidade de repetições indeterminada
 - Com teste no início;
 - Com teste no final.

Estrutura de Repetição Indeterminada - Com Teste no Início

- Comando **Enquanto**

- Trata-se de uma estrutura iterativa que permite a execução de uma ação ou de um bloco comando como consequência de uma condição verdadeira.

- **Sintaxe:**

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

Criar um algoritmo que leia a nota de um número indefinido de alunos. Para cada aluno exibir se o mesmo é aprovado, em recuperação ou reprovado.

```
programa{
    funcao inicio()
    {
        caracter continuar = 's'
        real nota

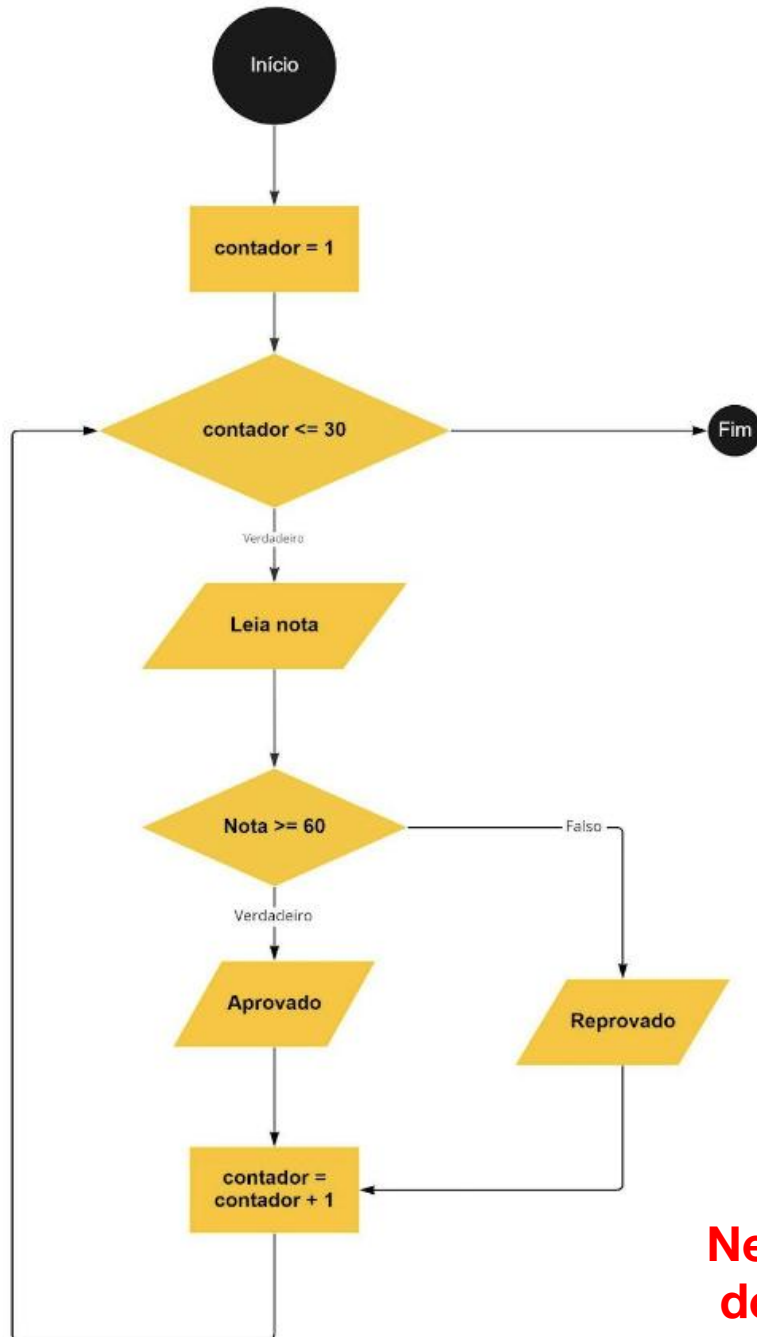
        enquanto (continuar == 's'){
            escreva("\nDigite a nota de um aluno: ")
            leia(nota)

            se(nota >= 60){
                escreva("Aluno aprovado com a nota " + nota + " pontos.")
            } senao se(nota >= 40){ // nota < 60
                escreva("Aluno em recuperação com a nota " + nota + " pontos.")
            } senao{
                escreva("Aluno reprovado com a nota " + nota + " pontos.")
            } // fim do se

            escreva("\nDeseja continuar(s/n): ")
            leia(continuar)
        } // fim do enquanto
    }
}
```

1
2
3

tenha como entrada a nota de 30 alunos e
"provado" (nota ≥ 60) e "Reprovado" (nota



```
programa
{
    funcao inicio()
    {
        inteiro contalunos = 1
        real nota
        enquanto (contalunos <= 30){
            escreva("\nDigite a nota do " + contalunos + "º aluno: ")
            leia(nota)
            se(nota >= 60){
                escreva("Aluno aprovado")
            }senao{
                escreva("Aluno reprovado")
            }//fim do se
            contalunos++ //incremento
        }//fim do enquanto
    }
}
```

Neste exemplo, a estrutura de repetição atuou um número de vezes determinado!

Estrutura de Repetição - Enquanto

- Na maioria dos loops são necessários:
 1. inicializar a(s) variável(eis) de controle antes do loop;
 2. incrementar/decrementar ou alterar o valor da(s) variável(eis) de controle dentro do loop;
 3. definir bem a condição do loop para evitar repetições infinitas.
- Atenção: no exemplo foi necessário incrementar a variável de controle (variável contador).
 - para evitar o loop infinito, que é resultante de erro na lógica.
- **comando enquanto** é útil quando não sabemos o número exato de repetições antecipadamente, mas sabemos a condição que precisa ser atendida. Embora ela possa atuar também quando se sabe de antemão o número de repetições.

Estrutura de Repetição Indeterminada - Com Teste no Fim

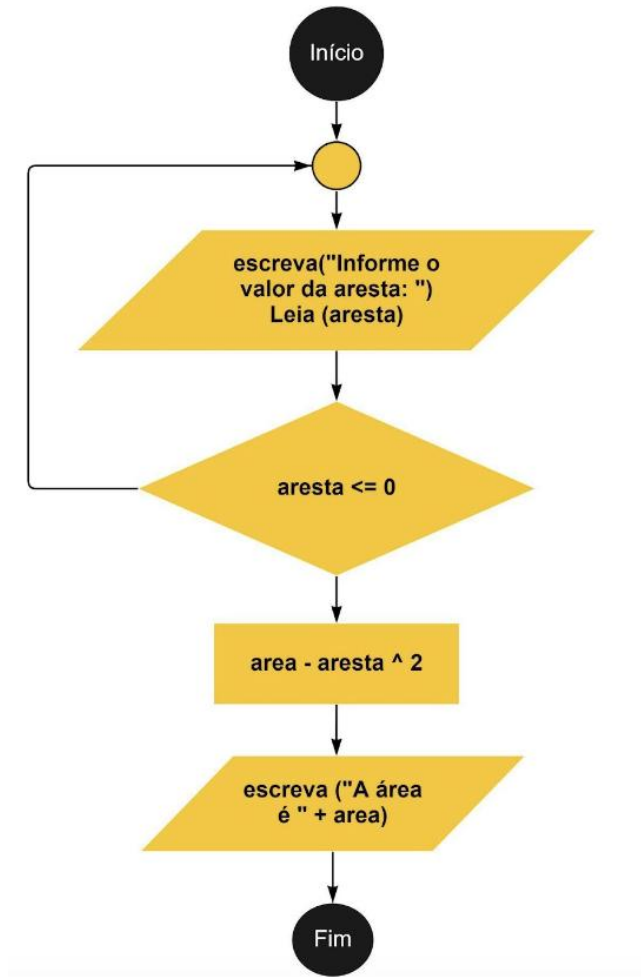
- Comando **Faça-Enquanto**

- Trata-se de uma estrutura iterativa que executa uma ação ou um bloco comando e permite sua repetição até que determinada condição seja verdadeira. Executa a ação/bloco comando pelo menos 1 vez.

- **Sintaxe:**

```
logico condicao = verdadeiro  
faca  
{  
    //Executa os comandos pelo menos uma vez, e continua executando enquanto a condição for verdadeira  
} enquanto (condicao)
```


- Crie um algoritmo que calcula a área de um quadrado a partir de uma valor de aresta informado. Se o valor da aresta for menor ou igual a 0, o programa repete o comando pedindo para que o usuário entre novamente com um valor válido (maior que 0).



```
programa
{
  funcao inicio()
  {
    real aresta, area

    faca
    {
      escreva ("Informe o valor da aresta: ")
      leia (aresta)
    } enquanto (aresta <= 0)

    area=aresta*aresta
    escreva("A área é: ", area)
  }
}
```

- O algoritmo recebe um conjunto de valores e ao final, calcula e exibe a média.

```
programa
{
    funcao inicio()
    {
        real valor, media, soma=0.0
        caracter continuar = 's'
        inteiro cont = 0

        faca{
            escreva("Informe um valor: ")
            leia (valor)
            soma = soma + valor
            cont++
            escreva("Deseja continuar (s/n):")
            leia (continuar)
        }enquanto(continuar != 'n')

        media = soma / cont
        escreva("A média é " + media)
    }
}
```

Estrutura de Repetição: **Faça-Enquanto**

- Na maioria dos loops também são necessários:
 1. inicializar a(s) variável(eis) de controle antes do loop;
 2. incrementar/decrementar ou alterar o valor da(s) variável(eis) de controle dentro do loop;
 3. definir bem a condição do loop para evitar repetições infinitas.
- **Atenção:** foi necessário atualizar a variável de controle para evitar o loop infinito, que é resultante de erro na lógica.
- **comando faça-enquanto** similar ao enquanto, mas garante que o bloco de código seja executado pelo menos uma vez.

Questão Desafio

- Crie um algoritmo que permita que você jogue um jogo de adivinhação com o computador. O programa deve escolher aleatoriamente um número dentro de um intervalo especificado. O usuário deve tentar adivinhar o número escolhido pelo computador. Após cada palpite do usuário, o programa deve fornecer uma dica indicando se o palpite é maior ou menor que o número escolhido. O programa deve continuar solicitando palpites até que o usuário adivinhe corretamente o número escolhido pelo computador. Quando o número for adivinhado corretamente, o programa deve exibir uma mensagem parabenizando o usuário e informando o número de tentativas realizadas.

Estrutura de Repetição Determinada

- Comando **Para**

- Trata-se de uma estrutura iterativa com a quantidade de execuções pré-determinada.

- **Sintaxe:**

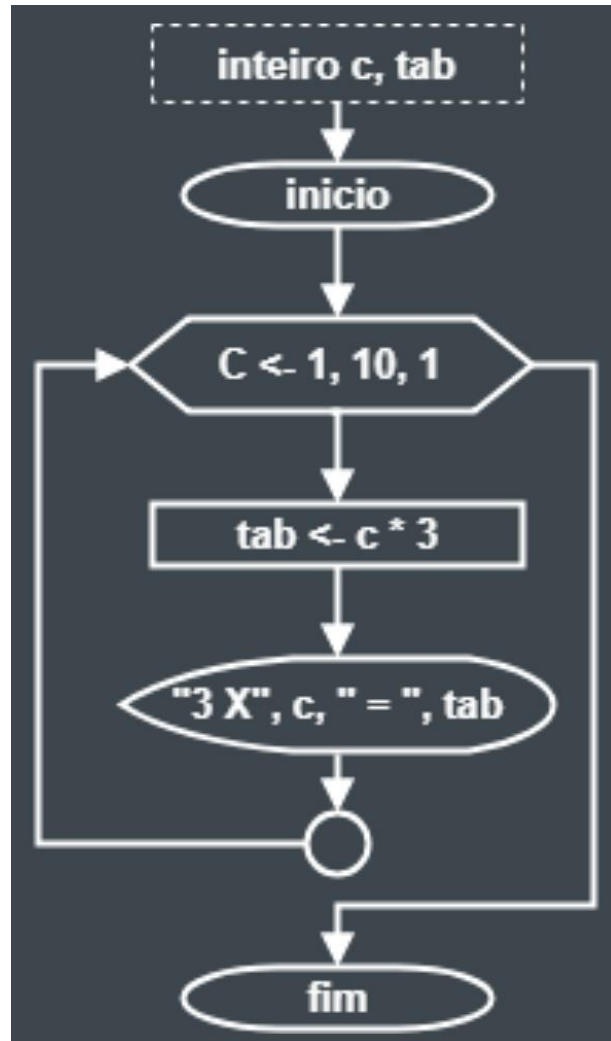
**Variável
Controle**

**Condição
de repetição**

**atualização
variável controle**

```
01.  para (inteiro i = 0; i < 8; i++)
02.  {
03.      //Codigo a ser executado enquanto a condição for satisfeita.
04.  }
```

- Criar um algoritmo que exibe a tabuada de 3



programa

{

funcao inicio()

{

inteiro tab

para (inteiro c=1; c<=10; c++)

{

tab=c*3

escreva ("3 x ", c, " = ", tab, "\n")

}

}

}

Criar um algoritmo que exibe a tabuada de qualquer número:

```
programa
{
    funcao inicio()
    {
        inteiro numero, resultado, contador

        escreva("Informe um número para ver sua tabuada: ")
        leia(numero)

        limpa()

        para (contador = 1; contador <= 10; contador++)
        {
            resultado = numero * contador
            escreva (numero, " X ", contador, " = ", resultado , "\n")
        }
    }
}
```

Estrutura de Repetição - Para

- Na própria estrutura do comando é feito:
 1. inicialização da variável de controle do loop;
 2. definição da condição do loop para evitar repetições infinitas;
 3. incremento/decremento da variável de controle;
- **comando para** é útil quando se sabe o número de repetições antecipadamente.

Vamos Treinar?

- Crie um programa que desenhe um triângulo de asteriscos invertido com base no número de linhas especificado pelo usuário. Por exemplo, se o usuário inserir o valor 5, o programa deve exibir a seguinte saída:

```
*****  
****  
***  
**  
*
```

Vamos Treinar?

- Você fica tão feliz no natal que tem vontade de gritar para todo mundo: "Feliz natal!!". Pra colocar toda essa felicidade para fora, você deve escrever um programa que, dado um índice *I* de felicidade, seu grito de natal é mais animado, repetindo *I* vezes a última letra "a" da frase "Feliz Natal!". Por exemplo, se você receber um índice de felicidade igual 5, a saída é composta pela frase "Feliz nataaaaaal!"

Bibliografia

- Notas de aula prof Márcio Assis
- Material de Ajuda disponível na plataforma Portugol Studio
- Notas de aula prof Juan Brettas UFF. Disponível em :
<https://www.professores.uff.br/juanbrettas/wp-content/uploads/sites/114/2017/08/4-Estrutura-de-repeti%C3%A7%C3%A3o.pdf> <Acessado em 30/07/24>