

Algoritmos

Aula 04 – Algoritmos: estruturas de decisão

Professora

Laysa Mabel de Oliveira Fontes

mabel.fontes@ufersa.edu.br

Pau dos Ferros/RN

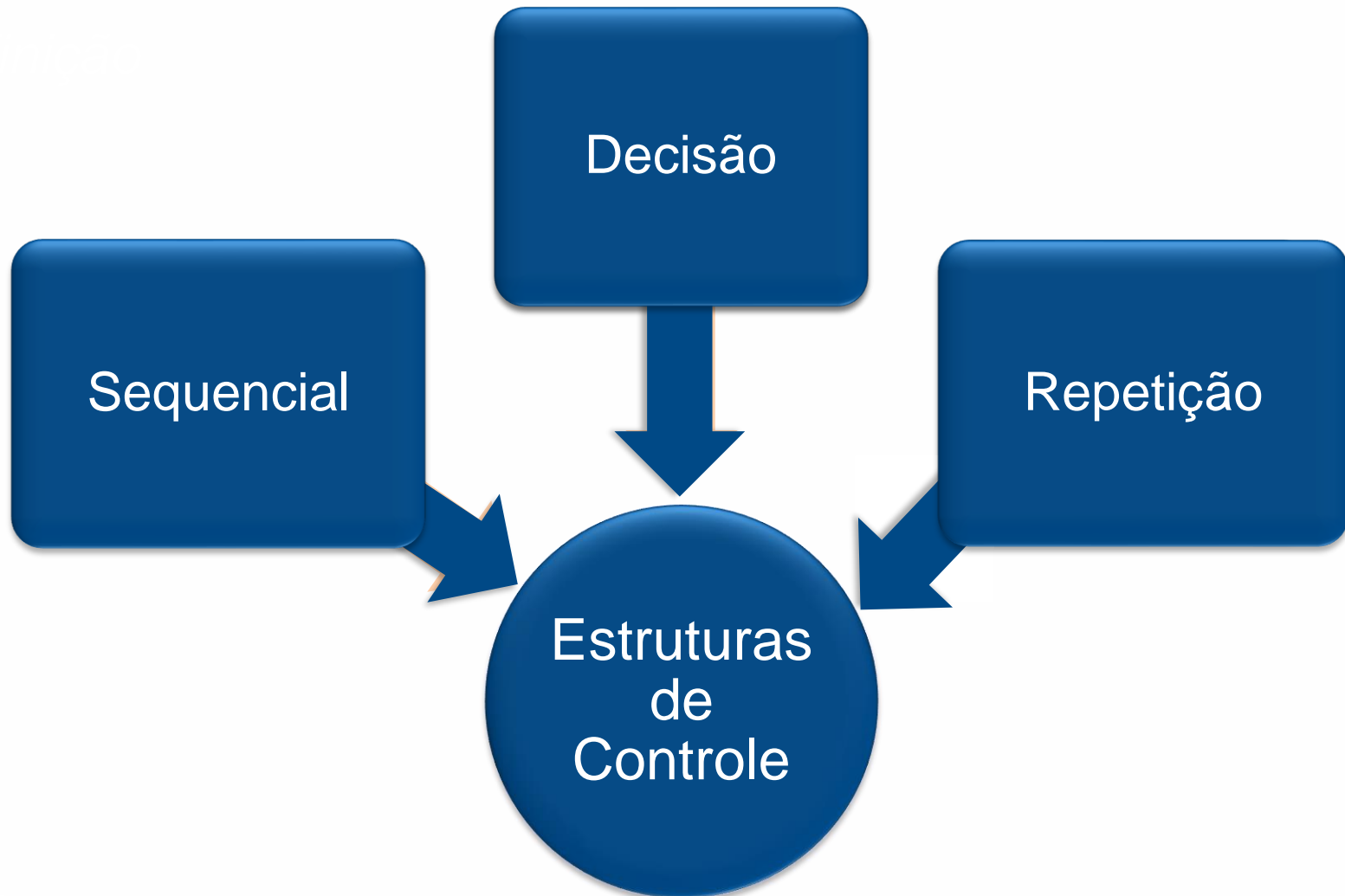
2022

Controle de Fluxo de um Algoritmo

O que é?

Refere-se à ordem em que as instruções de um algoritmo são executadas.

Controle de Fluxo de um Algoritmo



Estrutura Sequencial

Como funciona?

As ações são executadas em sequência, uma após a outra, na ordem em que se encontram dentro do algoritmo.

Estrutura Sequencial

- **Sintaxe:**

```
algoritmo "<nome do algoritmo>"  
var  
    ...  
inicio  
    <comando-1>  
    <comando-2>  
    ...  
    <comando-n>  
fimalgoritmo
```

Exemplo

- Exemplo:

```
algoritmo "Soma"  
var  
    n1, n2, s: inteiro  
inicio  
    escreva("Digite um número: ")  
    leia(n1)  
    escreva("Digite outro número: ")  
    leia(n2)  
    s := n1 + n2  
    escreva("A soma é:", s)  
fimalgoritmo
```

Estruturas de Decisão

Como funcionam?

Permitem que uma instrução, ou bloco de instruções, sejam ou não executadas, dependendo de determinadas condições.

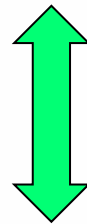
Exemplo

Se eu tiver pelo menos R\$ 50,00, então vou ao cinema.

Exemplo

- Analisando o exemplo anterior...

Qual é a condição na frase anterior?

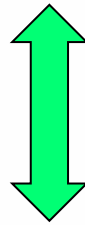


Ter pelo menos R\$ 50,00

Exemplo

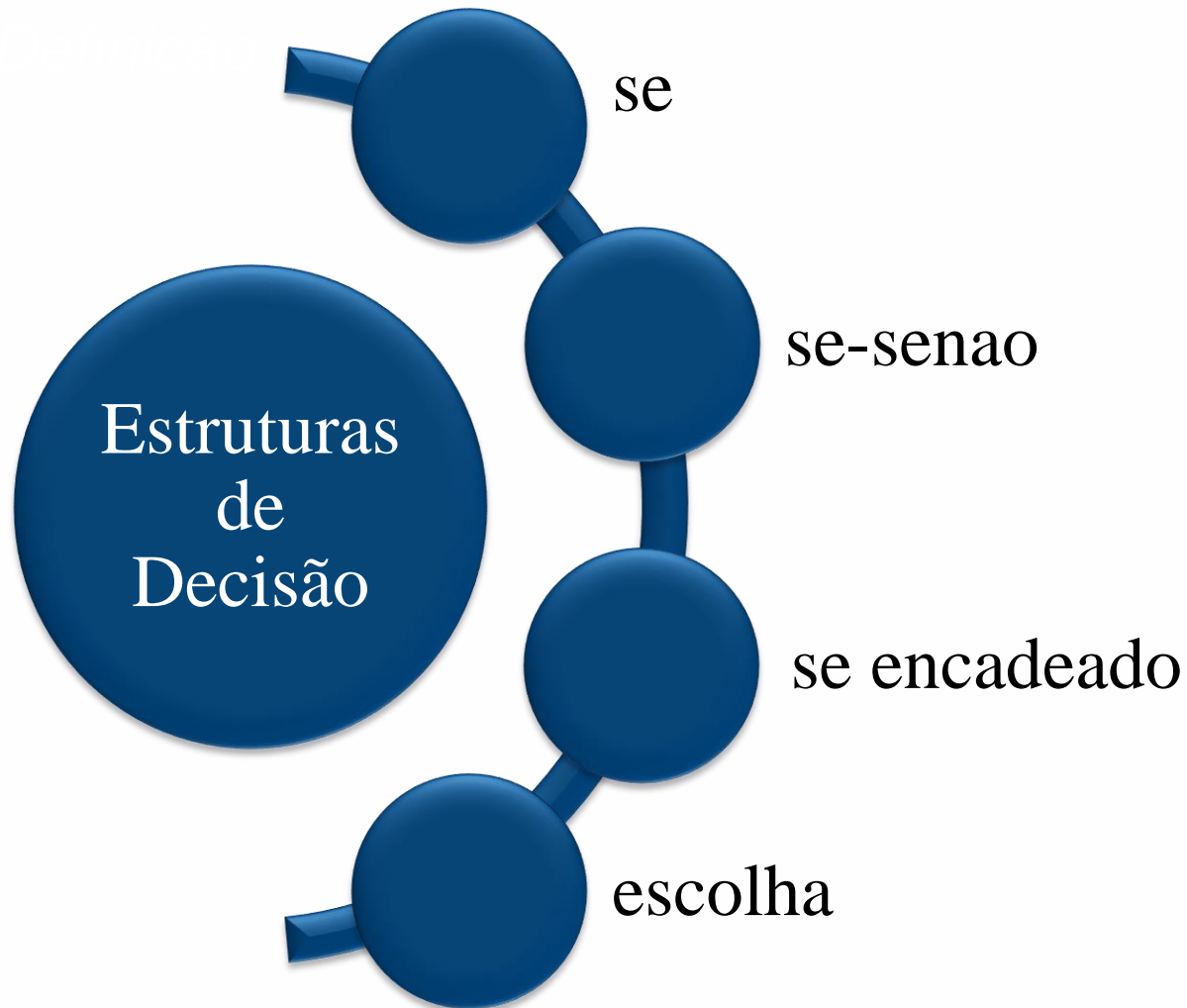
- Analisando o exemplo anterior...

Se a condição for verdadeira, qual ação deverá ser executada?



Ir ao cinema

Estruturas de Decisão



se

Como funciona?

Executa uma determinada instrução, ou bloco de instruções, quando uma determinada condição for satisfeita.

- Sintaxe:

```
algoritmo “<nome do algoritmo>”  
var  
    ...  
inicio  
    ...  
    se(<condição>) entao  
        <bloco-v>  
    fimse  
fimalgoritmo
```

condição = expressão lógica

Exemplo

Se eu tiver pelo menos R\$ 50,00, então vou ao cinema.

Exemplo

- Exemplo:

```
algoritmo "Simula Feriado"  
var  
    d: real  
inicio  
    escreva("Você tem quantos reais? ")  
    leia(d)  
    se (d >= 50) entao  
        escreva("Vou ao cinema!")  
    fimse  
fimalgoritmo
```

Como funciona?

Permite escolher entre duas opções, em função do cumprimento ou não de uma determinada condição.

se-senao

- **Sintaxe:**

```
algoritmo “<nome do algoritmo>”  
var  
    ...  
inicio  
    ...  
    se(<condição>) entao  
        <bloco-v>  
    senao  
        <bloco-f>  
    fimse  
fimalgoritmo
```

Exemplo

Se eu tiver pelo menos R\$ 50,00, então vou ao cinema,
senão, vou ficar em casa.

Exemplo

- Exemplo:

```
algoritmo "Simula Feriado"  
var  
    d: real  
inicio  
    escreva("Você tem quantos reais? ")  
    leia(d)  
    se (d >= 50) entao  
        escreva("Vou ao cinema!")  
    senao  
        escreva("Vou ficar em casa!")  
    fimse  
finalgoritmo
```

se encadeado

Como funciona?

O agrupamento de várias seleções normalmente ocorre quando há muitas possibilidades de situações dependentes entre si.

se encadeado

- Sintaxe:

```
algoritmo "<nome do algoritmo>"  
var  
    ...  
inicio  
    ...  
    se(<condição-1>) entao  
        <bloco-v>  
    senao  
        se(<condição-2>) entao  
            <bloco-v>  
        senao  
            <bloco-f>  
    fimse  
fimse  
fimalgoritmo
```

Ressalta-se que cada bloco pode conter qualquer tipo de comando, inclusive, uma estrutura de decisão.

Exemplo

Se eu tiver pelo menos R\$ 100,00, então vou jantar fora,
senão, se eu tiver pelo menos R\$ 50,00, então vou ao
cinema, senão, vou ficar em casa.

Exemplo

- Exemplo:

```
algoritmo "Simula Feriado"  
var  
    d: real  
inicio  
    escreva("Você tem quantos reais? ")  
    leia(d)  
    se (d >= 100) entao  
        escreva("Vou jantar fora!")  
    senao  
        se (d >= 50) entao  
            escreva("Vou ao cinema!")  
        senao  
            escreva("Vou ficar em casa!")  
        fimse  
    fimse  
fimalgoritmo
```

Como funciona?

A estrutura de decisão múltipla avaliará uma expressão que poderá receber n valores diferentes.

escolha

- Sintaxe:

```
algoritmo "<nome do algoritmo>"
var
    ...
inicio
    ...
    escolha(<variável>)
        caso <valor-1>
            <bloco-v>
        caso <valor-2>
            <bloco-v>
        ...
        caso <valor-n>
            <bloco-v>
        outrocaso
            <bloco-f>
    fimescolha
finalgoritmo
```

No Visualg, a variável de controle do escolha deve ser do tipo inteiro ou caractere. Além disso, o outrocaso é opcional.

Exemplo

- Exemplo:

```
algoritmo "Tele_Atendimento"  
var  
    op: inteiro  
inicio  
    escreval("Digite uma das opções a seguir: ")  
    escreval("1 - Consultar saldo")  
    escreval("2 - Consultar promoções")  
    escreval("3 - Falar com um atendente")  
    leia(op)  
    escolha(op)  
        caso 1  
            escreval("Você deseja consultar saldo.")  
        caso 2  
            escreval("Você deseja consultar promoções.")  
        caso 3  
            escreval("Você deseja falar com um atendente.")  
        outrocaso  
            escreval("Opção inválida!")  
    fimescolha  
fimalgoritmo
```

Vamos exercitar?

- 1) Elabore um algoritmo em pseudocódigo que solicita e lê a idade de uma pessoa e exibe uma mensagem informando se ela é, ou não, idosa. Considere idosa, a pessoa que possuir idade maior ou igual a 60 anos.

Referência

- MANZANO, J. A. N. G; OLIVEIRA, J. F. **Algoritmos: lógica para desenvolvimento de programação de computadores**. 27^a ed. São Paulo: Érica, 2014. (Capítulo 4).

