

SELEÇÃO SIMPLES É COMPOSTA

Material: Prof. João Teixeira

Prof. Jefferson A. Santos

CONSTRUÇÃO DE ALGORÍTMOS



Para criar algoritmos necessita-se de:

- Entrada(s), processamento(s) e saída(s);

No entanto estes três mecanismos não são suficientes.

Quando se pretende resolver um problema é preciso tomar decisões, da mesma forma o programa de computador precisa “**tomar decisões**”, ou seja, reagir mediante situações de entrada.

ESTRUTURA DE SELEÇÃO

As estruturas de seleção ou decisão são utilizadas quando existe a necessidade de verificar condições para determinar quais instruções serão, ou não, executadas. Os testes de seleção também podem ser utilizados para verificar opções de escolha.

As estruturas de seleção podem ser do tipo simples, composto ou encadeado.

ESTRUTURA DE SELEÇÃO - CONTEXTO

Suponha que uma pessoa esteja em um jogo de computador:

1) Para que o jogador passe de uma fase (etapa) para a seguinte, é necessário **verificar se ele atingiu a pontuação exigida**. Assim, existe uma condição para realização de uma sequência de instruções, de modo a liberar o acesso à próxima fase do jogo.

2) Ao final do jogo, uma pergunta é feita: **"Deseja continuar jogando?"**. O jogador poderá escolher entre as respostas sim ou não.

CONSTRUÇÃO DE ALGORÍTMOS

Na realidade o programa de computador é desenvolvido de forma que ele reaja mediante alternativas de soluções dependentes das entradas que foram realizadas.

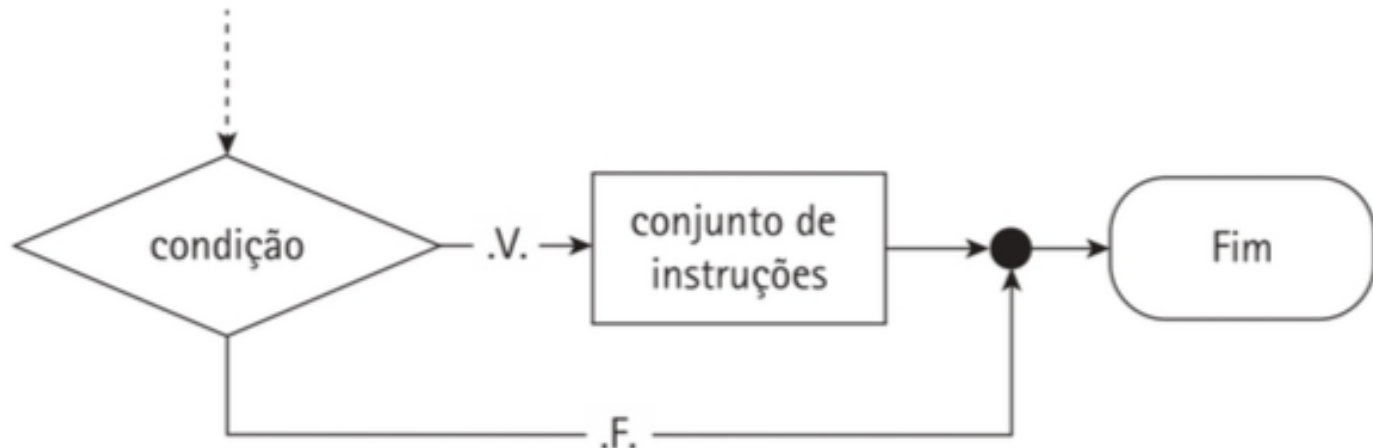


ESTRUTURA DE SELEÇÃO SIMPLES

Pseudocódigo:

```
Se (condição) então [início do bloco de seleção]
    conjunto de instruções
Fim-Se                      [fim do bloco de seleção]
```

Fluxograma:



ESTRUTURA DE SELEÇÃO SIMPLES

```
C#  
if (condição)  
{  
  
    //comando de instruções  
  
}
```

ESTRUTURA DE SELEÇÃO SIMPLES

Permite ao computador mediante a entrada ou resgate de informações decidir se vai ou não executar um determinado comando.

O computador

- Analisa a condição;
 - Caso ela seja **verdadeira** executa o comando;
 - Caso ela seja **falsa**, ele “pula” para o primeiro comando abaixo da }.

ESTRUTURA DE SELEÇÃO SIMPLES

Porque a marcação de { e } é necessária

Para que o computador realize mais de uma operação (comandos) é preciso colocar um bloco de comando, delimitado pelas palavras {

```
if (condição)
{
    //Comando 1
    //Comando 2
    //Comando 3
    //...
    //Comando N
}
```

Se a condição resultar **verdadeiro**, então o bloco (seqüência de comando1 até comandoN) será executado.

Caso contrário (condição retornar **falso**), executa o primeiro comando abaixo da }.

ESTRUTURA DE SELEÇÃO SIMPLES

Exemplo 1: Escrever um algoritmo que leia o nome e o salário de um funcionário. Caso o salário dele seja menor ou igual a R\$ 150,00, ele receberá um benefício de R\$ 30,00.

ESTRUTURA DE SELEÇÃO SIMPLES

Exemplo 2: Escreva um algoritmo que leia 3 notas de um determinado aluno, e calcule a média deste aluno.

Verifique se o aluno foi aprovado ou não.

Para um aluno ser aprovado a média deve ser maior ou igual a 7,0.

Escreva a média do aluno e a mensagem correspondente (Aprovado ou Reprovado).

ESTRUTURA DE SELEÇÃO COMPOSTA

Ao se testar condições só poderão ser obtidos dois resultados:

Verdadeiro ou **Falso**

A estrutura de seleção simples permite apenas que se execute ou não determinados comandos quando a condição é verdadeira.

ESTRUTURA DE SELEÇÃO COMPOSTA

Na composta, a estrutura deverá apresentar duas alternativas:

- Uma a ser executada caso a condição testada retorne um valor **verdadeiro**;
- E outra, caso a condição retornar **falso**.

Pode-se ter apenas uma das duas respostas, ou seja:

- **Uma e somente** uma destas alternativas **sempre será executada**.

ESTRUTURA DE SELEÇÃO COMPOSTA

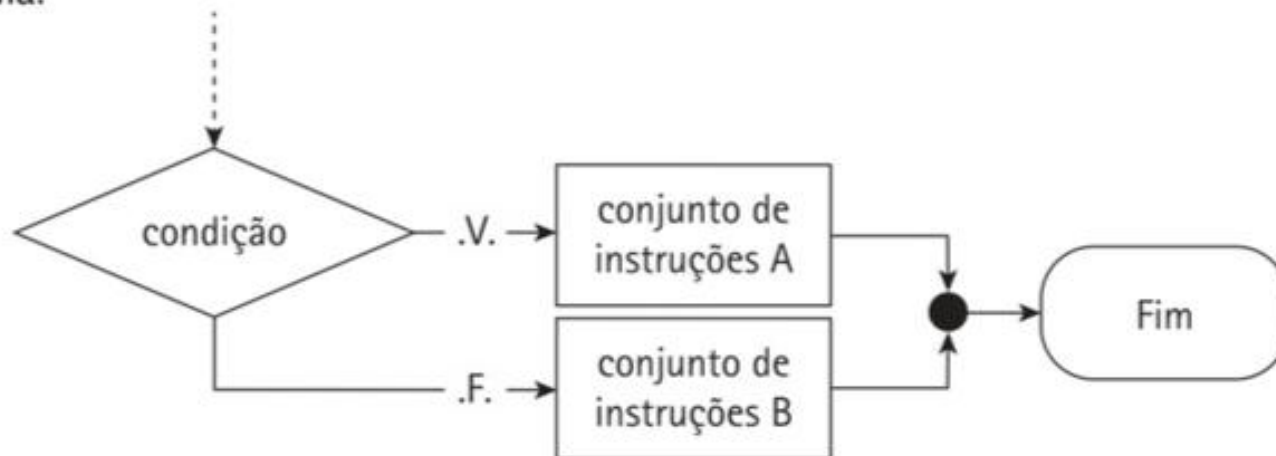
Pseudocódigo:

```
Se (condição) então  
    conjunto de instruções A  
Senão  
    conjunto de instruções B  
Fim-Se
```

Conjunto de instruções que será realizado se o teste da condição resultar verdadeiro.

Conjunto de instruções que será realizado se o teste da condição resultar falso.

Fluxograma:



ESTRUTURA DE SELEÇÃO COMPOSTA

```
C#  
if ( condição )  
{  
  
    //Comando A  
  
}  
else  
{  
  
    //Comando B  
  
}
```

Se a condição resultar **verdadeiro**, o Comando A será executado, caso contrário (condição resultar **falso**), o Comando B será executado.

ESTRUTURA DE DECISÃO COMPOSTA

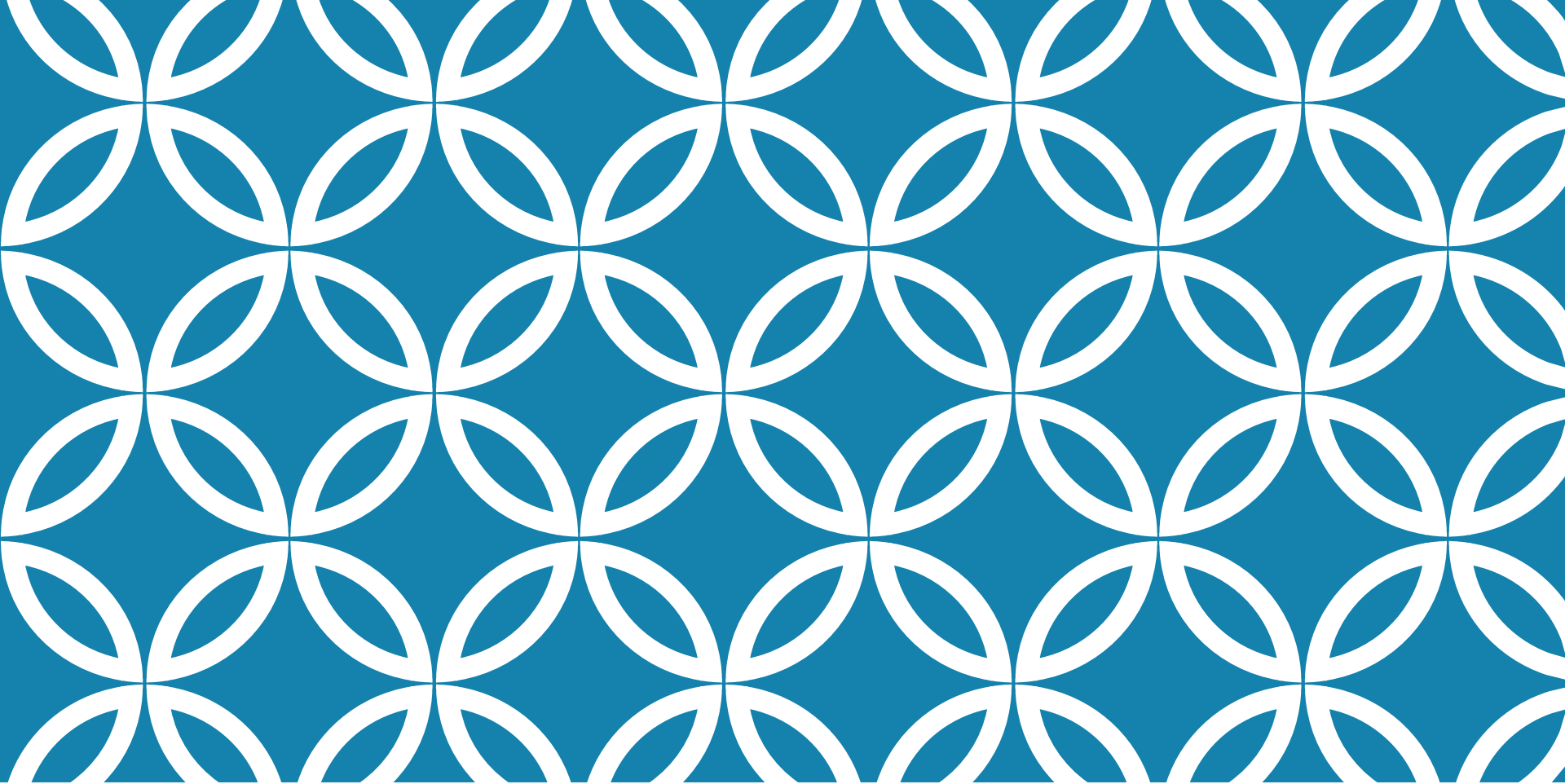
Entendendo o algoritmo

Após a leitura dos dois valores, o computador:

- faz a avaliação da expressão lógica ($A > B$) e decide se deve executar os comandos da parte verdadeira ou da parte falsa do comando se, ou seja, caso o valor de A seja maior que de B ele escreverá o valor de A, caso contrário escreverá, o de B.

ESTRUTURA DE SELEÇÃO COMPOSTA

Faça um algoritmo que receba três notas de um aluno, se a média for maior que 7, indique que o aluno está aprovado. Se for menor que 7, que este está reprovado:



SELEÇÃO
ENCADEADA

ESTRUTURA DE SELEÇÃO ENCADEADA

Uma estrutura de seleção encadeada é formada pela combinação de estruturas de seleção de seleção simples ou compostas, uma dentro da outra, não havendo limite para o número de combinações, podendo, em alguns casos, gerar um código bastante complexo.

ESTRUTURA DE SELEÇÃO ENCADEADA

Encadeada => ligada a outros.

Em alguns casos não basta uma única condição para que se possa obter um resultado.

Para tanto é preciso estabelecerem verificações de condições sucessivas, onde:

- uma determinada ação poderá ser executada se um conjunto anterior de instruções ou condições for satisfeito.
- Sendo a ação executada, esta poderá ainda estabelecer novas condições.

Isto significa que pode-se utilizar uma condição dentro de outra condição.

ESTRUTURA DE SELEÇÃO ENCADEADA

Este tipo de estrutura poderá possuir diversos níveis de condição, sendo chamadas de:

- aninhamentos ou encadeamentos.

Pseudocódigo:

```
Se (condição_1) então
```

```
    Se (condição_2) então
```

```
        conjunto de instruções A
```

```
    Senão
```

```
        conjunto de instruções B
```

```
    Fim-Se
```

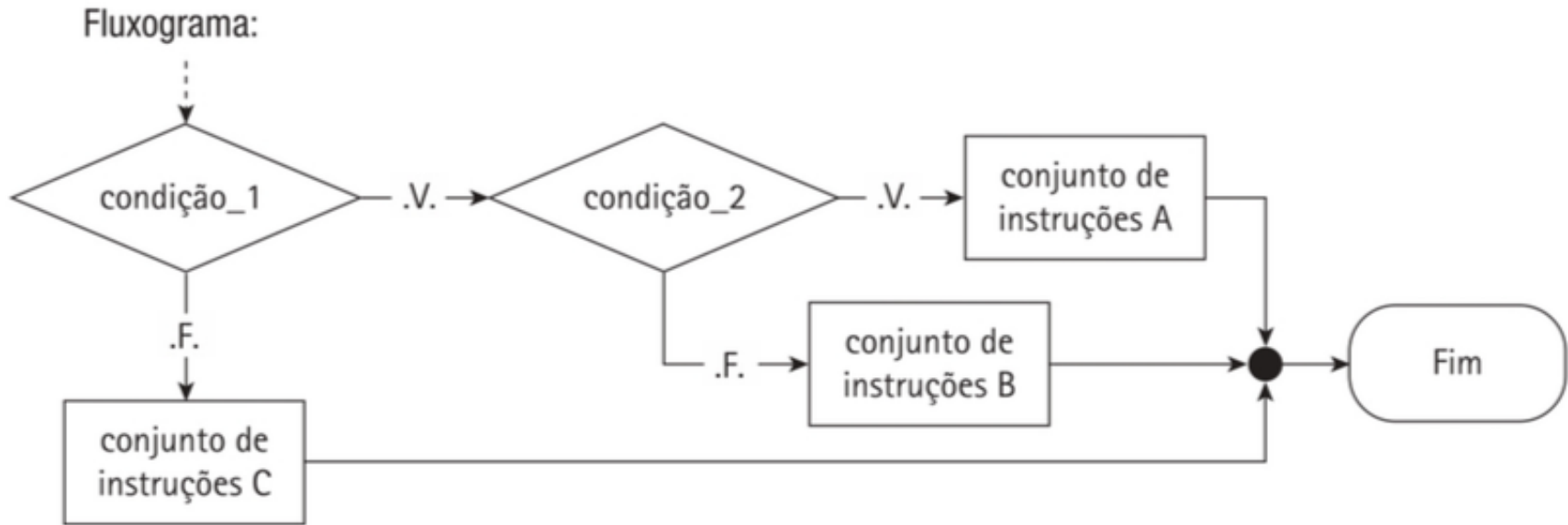
Seleção composta encadeada

```
Senão
```

```
    conjunto de instruções C
```

```
Fim-Se
```

ESTRUTURA DE SELEÇÃO ENCADEADA



ESTRUTURA DE SELEÇÃO ENCADEADA

C#

```
if (<condição_1>)  
{
```

```
    if (<condição_2>  
        {  
            <conjunto de instruções A>  
        }  
    else  
    {  
        <conjunto de instruções B>  
    }
```

Seleção composta encadeada

```
    }  
else  
{  
    <conjunto de instruções C>  
}
```

ESTRUTURA DE SELEÇÃO ENCADEADA

Exemplo:

Escreva um algoritmo que leia 2 números inteiros e mostre o maior número lido e caso sejam iguais, mostre a mensagem “Os números lidos são iguais”.

ESTRUTURA DE DECISÃO ENCADEADA

Escreva um algoritmo que leia 3 números e escreva o maior deles.

```
1  programa
2  {
3      funcao inicio()
4      {
5          inteiro a, b, c, maior
6
7          escreva("Entre com o primeiro número: ")
8          leia(a)
9          escreva("Entre com o segundo número: ")
10         leia(b)
11         escreva("Entre com o terceiro número: ")
12         leia(c)
13
14         se (a > b)
15         {
16             se (a > c)
17             {
18                 maior = a
19             }
20             senao
21             {
22                 maior = c
23             }
24         }
25         senao
26         {
27             se (b > c)
28             {
29                 maior = b
30             }
31             senao
32             {
33                 maior = c
34             }
35         }
36
37         escreva("O maior número lido é: ", maior)
38
39     }
40 }
```

ESTRUTURA DE DECISÃO

Entendendo o algoritmo:

Um algoritmo só pode testar variáveis duas a duas, ou seja, é **impossível** fazer um teste do tipo **Se $(A > B > C)$** .

Assim deve-se escolher duas e compará-las.

Escolhe-se ao acaso A e B e as compara-se:

- se **A for maior que B** então provavelmente o maior valor é **A** ou **C**, basta comparar ambas para descobrir o maior valor;
- se **B for maior que A**, precisamos comparar B com C para descobrir a maior delas.
- Ao realizar as comparações necessárias, atribui-se à variável MAIOR a variável de maior valor e em seguida ela é escrita na tela.

ESTRUTURA DE DECISÃO ENCADEADA

Pode-se escrever o mesmo algoritmo utilizando os operadores lógicos:

■ e, ou, não.

```
1 programa
2 {
3     funcao inicio()
4     {
5         inteiro a, b, c, maior
6
7         escreva("Entre com o primeiro número: ")
8         leia(a)
9         escreva("Entre com o segundo número: ")
10        leia(b)
11        escreva("Entre com o terceiro número: ")
12        leia(c)
13
14        se ((a > b) e (a > c))
15        {
16            maior = a
17        }
18        senao
19        {
20            se (b > c)
21            {
22                maior = b
23            }
24            senao
25            {
26                maior = c
27            }
28        }
29
30        escreva("O maior número lido é: ", maior)
31
32    }
33 }
```

1. Faça um algoritmo que leia um valor qualquer e verifique se este valor é positivo, nulo ou negativo.
2. Faça um algoritmo que leia um valor qualquer e verifique se este valor é múltiplo de 7.
3. Faça um algoritmo que receba a idade de uma pessoa e mostre uma mensagem informando se esta pessoa é maior de idade ou não. (OBS.: Para uma pessoa ser maior de idade, basta ter uma idade superior ou igual a 18 anos).
4. Faça um algoritmo que leia um valor qualquer e verifique se este valor é par ou ímpar.

5. Faça um algoritmo que receba a idade de um nadador e mostre a sua categoria seguindo as regras:

Categoria	Idade
Infantil A	5 – 7 anos
Infantil B	8 – 10 anos
Juvenil A	11 – 13 anos
Juvenil B	14 – 17 anos
Sênior	maiores de 18 anos

6. Escreva um algoritmo que leia dois valores quaisquer e calcule a soma destes valores. Após informe em qual intervalo a soma se encontra:

- a) Intervalo de 0 a 20;
- b) Intervalo de 21 a 40;
- c) Intervalo de 41 a 60;
- d) Acima de 61.

7. Um vendedor necessita de um algoritmo que calcule o preço total devido por um cliente. O algoritmo deve receber o código de um produto e a quantidade comprada e calcular o preço total, usando a tabela abaixo:

Código do Produto	Preço Unitário
1001	R\$ 5.32
1324	R\$ 6.45
6548	R\$ 2.37
0987	R\$ 4.26