

# **ESTRUTURAS CONDICIONAIS**



Introdução à Ciência da Computação  
Simone Senger de Souza

# Estruturas de Controle

 ESTRUTURA SEQUENCIAL

 ESTRUTURA CONDICIONAL

 ESTRUTURA DE REPETIÇÃO

# Estruturas Condicionais

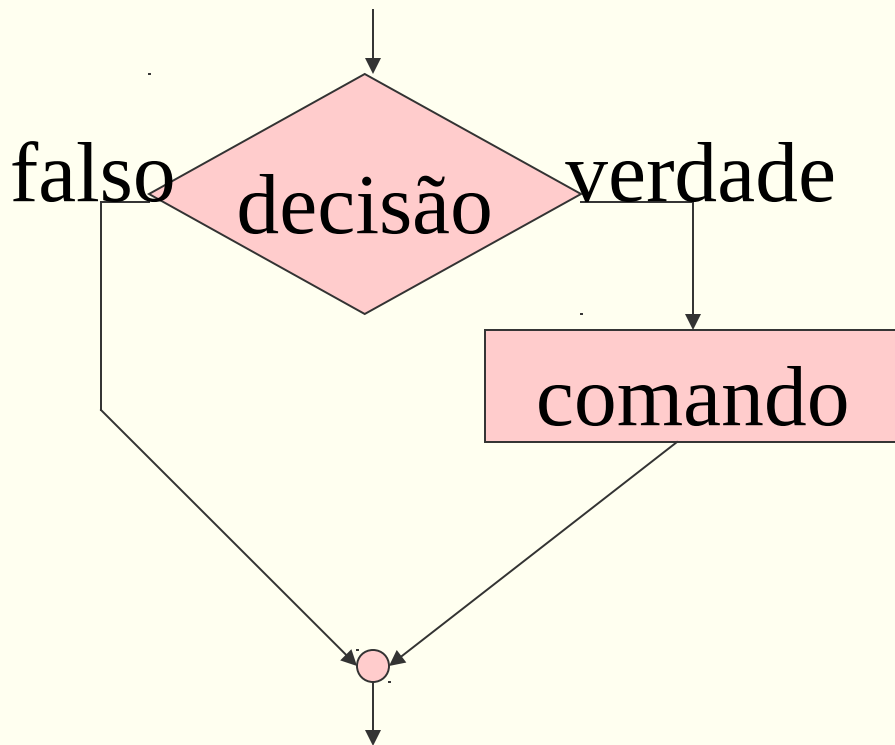
- 🌿 Estrutura Condicional Simples
- 🌿 Estrutura Condicional Composta
- 🌿 Seleção entre duas ou mais Seqüências de Comandos

# Estrutura Condicional Simples

Formas de Representação no Algoritmo

Fluxograma

Algoritmo



se (condição)

então <comando>

fim-se;

# Estrutura Condicional Simples C

```
if (condição)
```

```
<comando> ;
```

# Estrutura Condicionais

a condição deve ser uma **expressão lógica**

```
if (condição)
```

```
<comando> ;
```

O **comando** só será executado se a **condição** for **verdadeira**

# Estrutura Condicional Simples C

**if** (condição)

<comando>

se **mais de um comando** deve ser executado quando a **condição** for verdadeira, esses comandos devem ser transformados em um **comando composto**.

# Comando Composto

- 🐜 Um **conjunto** de **comandos** que devem ser executados em uma ordem específica.
- 🐜 Os comandos devem ficar reunidos entre as palavras chaves **{** e **}**

🐜 **{**

**comando 1;**

**comando 2;**

**comando 3;**

**}**

Comando composto



# Estrutura Condicional Simples C

```
if (condição)
{
    <comando> ;
    <comando> ;
    <comando> ;
}
```

## **EXEMPLO (condicional simples):** Ler dois números inteiros e encontrar o maior deles

Algoritmo

inicio

inteiro A,B, MAIOR;

leia (A, B);

MAIOR = A;

se (B > A)

então MAIOR = B;

fim-se;

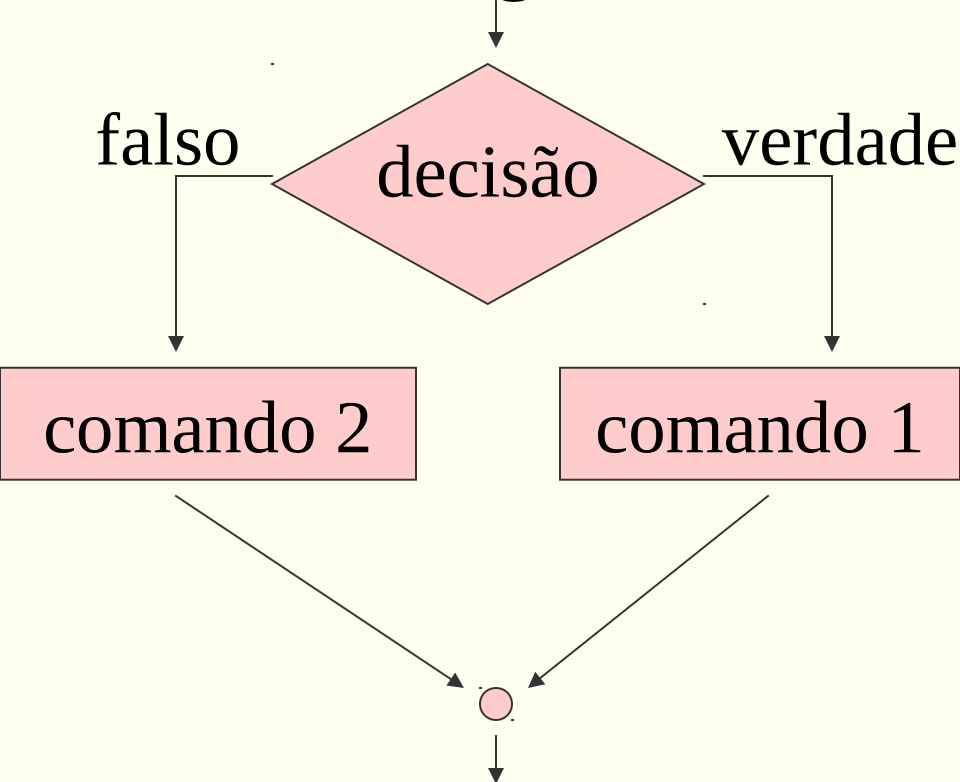
escreva (MAIOR);

Fim.

# Estrutura Condicional Composta

Formas de Representação no Algoritmo

Fluxograma



Algoritmo

se (condição)

então <comando1>

senão <comando2>

fim-se

# Estrutura Condicional Composta C

```
if (condição)
    <comando 1> ;
else <comando 2>;
```

# Estrutura Condicional Composta C

a condição deve ser uma expressão lógica

```
if (condição)
    <comando 1> ;
else <comando 2>;
```

# Estrutura Condicional Composta C

Se condição for verdadeira será executado o comando 1 e não será executado o comando 2.

```
if (condição)
    <comando 1> ;
else <comando 2>;
```

# Estrutura Condicional Composta C

Se condição for **falsa** será executado o **comando 2** e **não** será executado o **comando 1**.

```
if (condição)  
    <comando 1> ;  
else <comando 2>;
```

# Estrutura Condicional Composta C

```
if (condição)
    <comando 1> ;
else <comando 2>;
```

se mais de um comando deve ser executado quando a condição for verdadeira ou quando a condição for falsa, esses comandos devem ser transformados em comandos compostos.



# Estrutura Condicional Composta C

```
if (condição)
{
    <comando> ;
    <comando> ;
    <comando> ;
}
else
{
    <comando> ;
    <comando> ;
}
```

**EXEMPLO (condicional composto):** Calcular a área de um triângulo dado o comprimento de seus lados

Algoritmo area

Início

real A,B,C,P,AREA;  
solicitar leitura dos lados do triângulo  
leia (A, B, C);  
se (A<B+C) e (B<A+C) e (C<B+A)

**Verificação se os lados formam um triângulo**

então

$P = (A+B+C)/2;$

$AREA = \sqrt{(P(P-A))(P-B)(P-C)}$

escreva (AREA);

**Cálculo da Área**

senão escreva (Os numeros A, B e,C nao formam um triângulo)

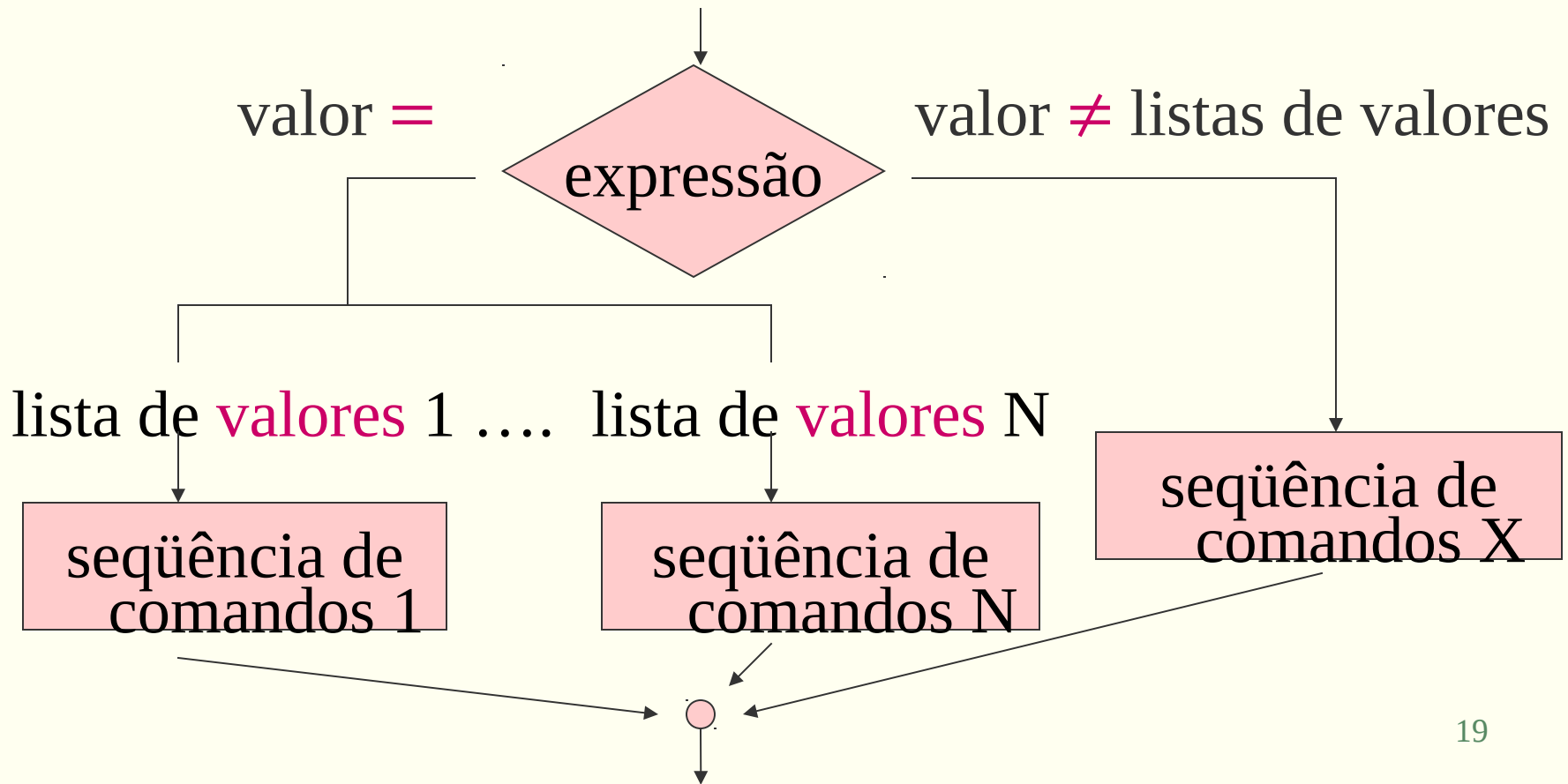
fim se;

fim

# MAIS SEQÜÊNCIAS DE COMANDOS

- Formas de Representação no Algoritmo

## Fluxograma



# SELEÇÃO ENTRE DUAS OU MAIS SEQÜÊNCIAS DE COMANDOS -

Formas de Representação no Algoritmo  
Algoritmo

caso <valor da expressão>

igual <lista de valores 1> então <seqüência de comandos 1>

.....

igual <lista de valores N> então <seqüência de comandos N>

senão <seqüência de comandos X>

fim-caso

# SELEÇÃO ENTRE DUAS OU MAIS SEQÜÊNCIAS DE COMANDOS C

```
switch (variável) {  
    case <constante1> : <seqüência de comandos 1> ; break;  
    ....  
    case <constante N> : <seqüência de comandos N> ; break;  
    default seqüência de comandos default;  
}
```

# SELEÇÃO ENTRE DIAS OU

A expressão é **avaliada** e então a seqüência associada ao seu **valor** é executada.

```
switch (variável) {  
  case <constante1> : <seqüência de comandos 1> ; break;  
  ....  
  case <constante N> : <seqüência de comandos N> ; break;  
  default seqüência de comandos default;  
}
```

Se o comando **break** não for utilizado os demais **cases** continuam sendo avaliados.

# SELEÇÃO ENTRE DUAS OU MAIS SEQÜÊNCIAS DE COMANDOS C

Se o valor da expressão **não** se encontra em nenhuma das listas de valores, a **seqüência default** será executada.

```
switch (va  
case <constante 1> : <seqüência de comandos 1> ; break;  
....  
case <constante N> : <seqüência de comandos N> ; break;  
default seqüência de comandos default;  
}
```

A opção **default** é facultativa

# EXEMPLO ( comando case ) : Dado o código da peça, determinar seu peso

## Algoritmo

início

caracter CODIGO;

real PESO;

solicitar o fornecimento do código da peça (P,M,G)

leia (CODIGO);

caso CODIGO

igual 'p','P' então

PESO=23.5;

escreva (PESO);

igual 'm','M' então

PESO=50.7;

escreva (PESO);

Algoritmo



igual 'g','G' então

PESO=102.9;  
escreva (PESO);

senão escreva (“Código Invalido”);

fim-caso

fim

Algoritmo

## **EXEMPLO ( comando switch )** : Dado o código da peça, determinar seu peso

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[])
{
    char codigo;
    float peso;
    printf("Entre com o código do produto (P,M,G):" );
    scanf("%c", &codigo);
    switch (codigo) {
        case 'P': peso=23.5;
                printf("o peso eh %.1f\n",peso);
                break;
        case 'M': peso=50.7;
                printf("o peso eh %.1f\n",peso);
                break;
```

```
case 'G': peso=102.9;
    printf("o peso eh %.1f\n",peso);
    break;
default: printf("\n\ncodigo invalido! \n");
}
```

```
system("PAUSE");
return 0;
}
```

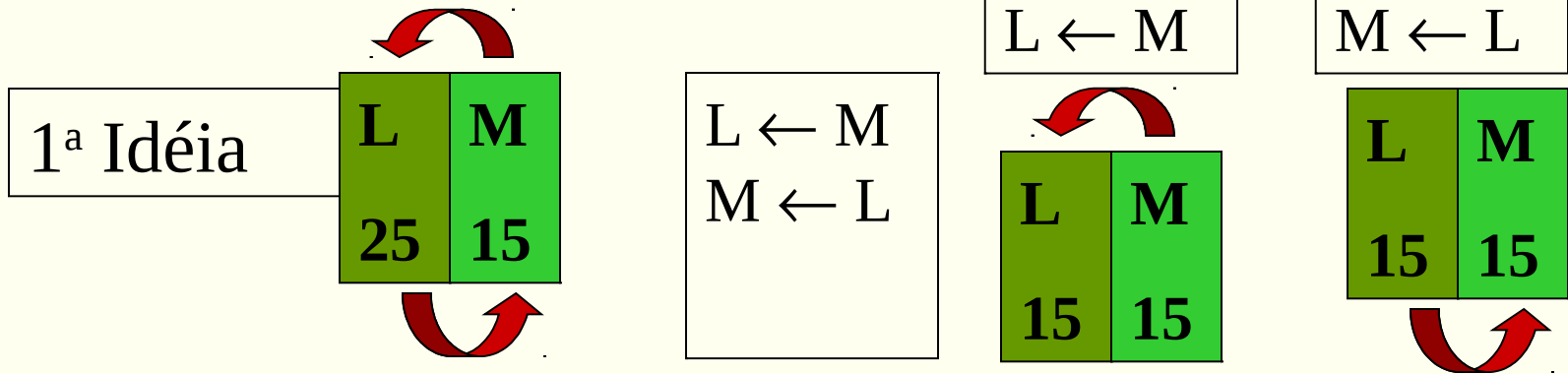
# Exemplo

1. Faça um algoritmo que lê dois números inteiros e mostra-os em ordem crescente.

# TROCA DE CONTEÚDO ENTRE DUAS VARIÁVEIS

L	M
25	15

memória



**NÃO FUNCIONA**

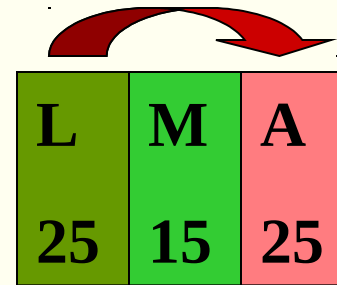
para trocar é preciso uma variável auxiliar

L	M	A
25	15	

memória

# TROCA DE CONTEÚDO ENTRE DUAS VARIÁVEIS

Faço uma cópia de L em A



$A \leftarrow L$

Coloco M no lugar de L



$L \leftarrow M$

Coloco o L que estava guardado em A no lugar de M



$M \leftarrow A$

# TROCA DE CONTEÚDO ENTRE DUAS VARIÁVEIS

trocar de lugar L e M

$$A \leftarrow L$$
$$L \leftarrow M$$
$$M \leftarrow A$$

## **EXEMPLO (condicional simples):** Ler dois números inteiros e colocá-los em ordem crescente

Algoritmo ordena

inicio

inteiro A, B, AUX;

leia (A, B );

se (A > B)

então

AUX = A;

A = B;

B = AUX;

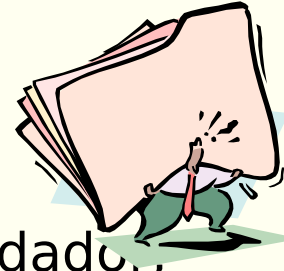
fim se;

escreva (A,B);

fim



# Exercícios



1. Elaborar um algoritmo que, dada a idade de um nadador, classificá-lo nas categorias: infantil A (5 - 7 anos), infantil B (8 -10 anos), juvenil A (11 - 13 anos), juvenil B (14 -17 anos) e adulto (maiores que 18 anos).
2. Escreva um algoritmo que lê um valor em reais e calcula qual o menor número possível de notas de 100, 50, 10, 5 e 1 em que o valor lido pode ser decomposto e escreva o valor lido e a relação de notas necessárias. Ex. R\$ 477,00 -> 4 notas de 100,00, 1 nota de 50,00, 2 notas de 10,00, 1 nota de 5,00 e 2 notas de 1,00.
3. Qualquer número natural de quatro algarismos pode ser dividido em duas dezenas formadas pelos seus dois primeiros e dois últimos dígitos. (1297 = 12 e 97; 5314 = 53 e 14). Escreva um algoritmo que lê um número inteiro  $n$  (de 4 algarismos) e verifica se a raiz quadrada de  $n$  é igual a soma das dezenas de  $n$ .

Ex.:  $n = 9801$ , dezenas de  $n = 98 + 01$ , soma das dezenas 99, raiz quadrada de  $n = 99$ . Portanto a raiz quadrada de 9801 é igual a soma de suas dezenas.