



Minicurso de C



Ramo Estudantil IEEE UEL

Minicurso de C

Módulo 3 – Operações lógicas e aritméticas

Fernando Morgado Pires Neto
Renan Ricoldi Fróis Pedro





Ensinar expressões algébricas em C
Introduzir o conceito de estruturas condicionais



- Tem como objetivo alterar o fluxo de operação no código, através de **comparações entre variáveis**.



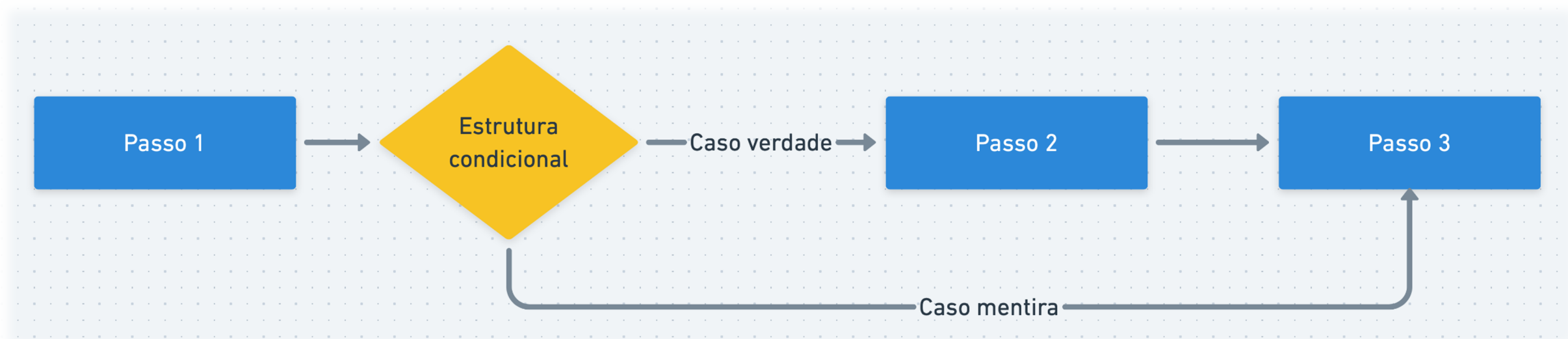
Passo 1



Passo 2



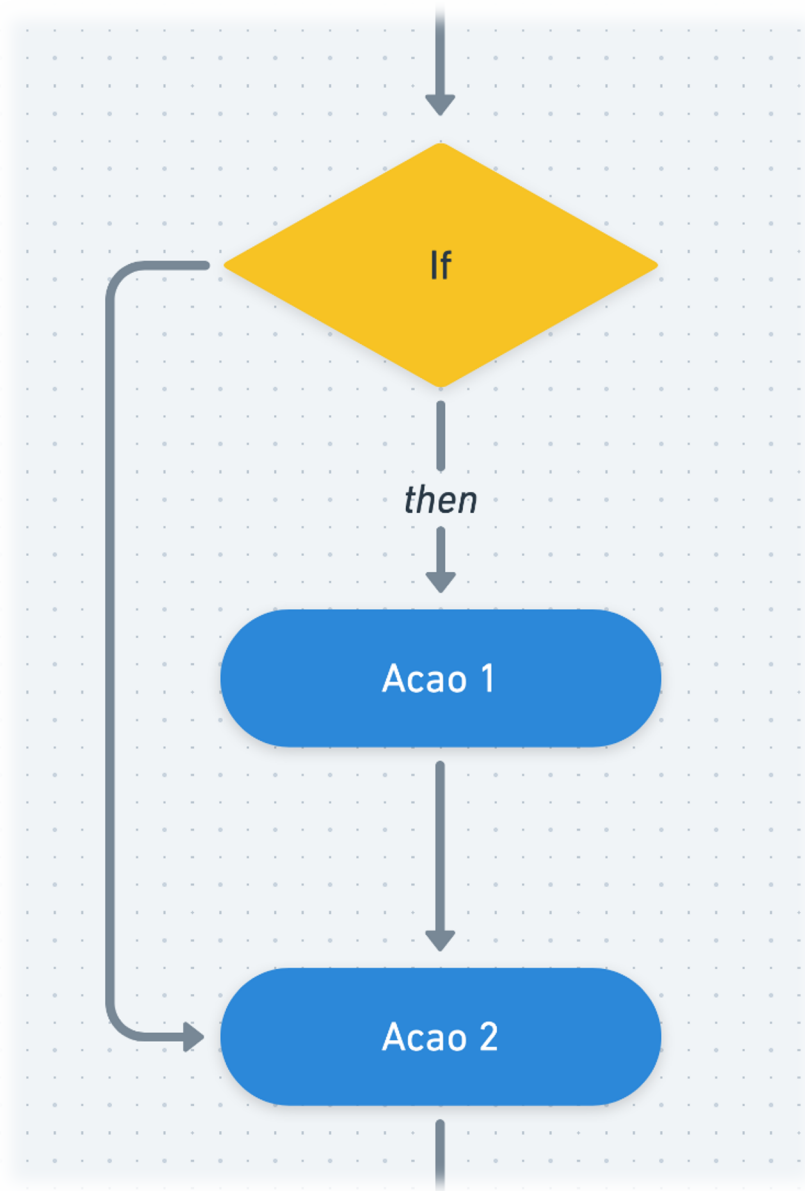
Passo 3





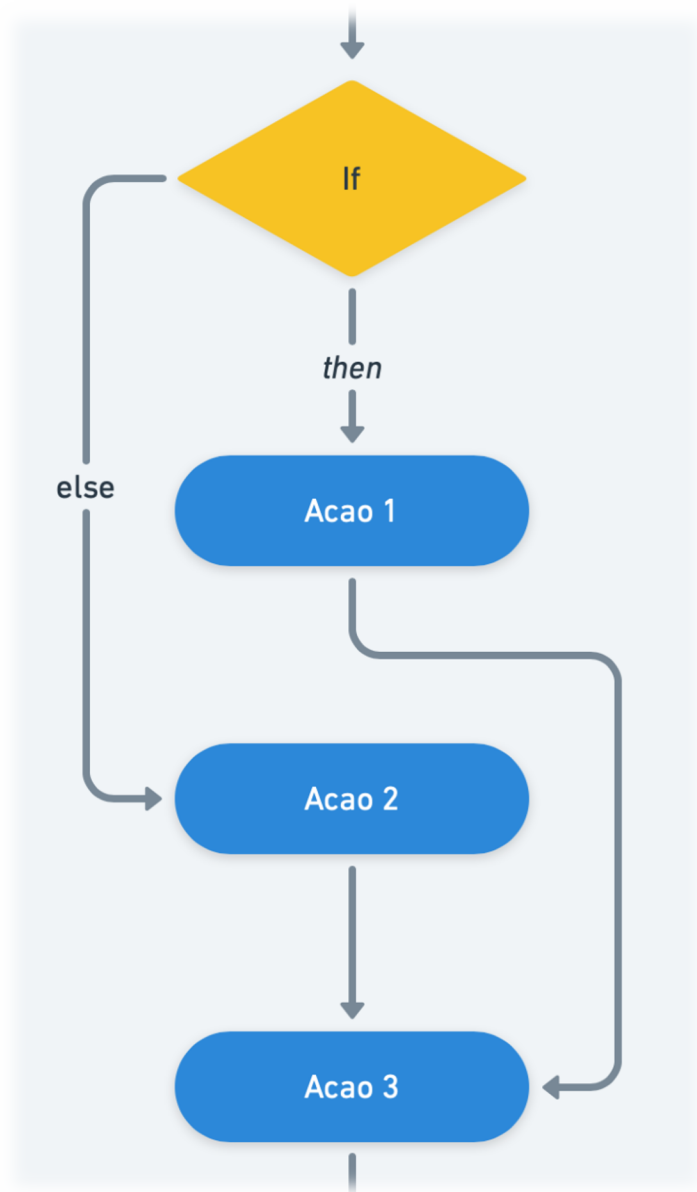
- A estrutura condicional **If Then Else** é a principal utilizada como elemento de comparação entre variáveis

If ... Then ...



- O if é utilizado para comparar as variáveis.
- Repare que **ação 1** só é executada caso a comparação seja **verdadeira** (then).
- Caso ela seja falsa, aquele trecho é pulado, e o algoritmo continua na sequência.

If ... Then ... Else ...



- Também podemos adicionar uma operação de **else** na estrutura.
- Ela só será executada caso o valor da comparação no if retornar **falso**.

Como realizar a comparação



- A comparação dentro das estruturas condicionais é sempre realizada entre duas variáveis apenas ou pares de variáveis.

Como realizar a comparação



- Dado **valor1** e **valor2**, temos:

Operador	Pergunta lógica
valor1 > valor2	valor1 é maior que valor2 ?
valor1 < valor2	valor1 é menor que valor2 ?
valor1 >= valor2	valor1 é maior ou igual que valor2 ?
valor1 <= valor2	valor1 é menor ou igual que valor2 ?
valor1 == valor2	valor1 é igual ao valor2 ?
valor1 != valor2	valor1 é diferente do valor2 ?

Como realizar a comparação



- Dado **valor1**, **valor2**, **valor3**, e **valor4**, temos:

Operador	Pergunta lógica
valor1 > valor2 && valor3 < valor4	valor1 é maior que valor2 E valor3 é menor que valor4 ?
valor1 > valor2 valor3 < valor4	valor1 é maior que valor2 OU valor3 é menor que valor4 ?

&& -> retorna VERDADEIRO se, e somente se, **todos os pares** retornarem verdadeiro

|| -> retorna VERDADEIRO caso **qualquer par** retornar verdadeiro

Como realizar a comparação



- Dado $v1$, $v2$, $v3$, $v4$, $v5$, $v6$, temos:

Operação	Pergunta lógica
$v1 \ \&\& \ (v2 > v3)$	$v1$ é verdadeiro? E $v2$ é maior que $v3$?
$(v1 == v2) \ \ ((v3 > v4) \ \&\& \ (v5 < v6))$	$v1$ é igual à $v2$? OU $v3$ é maior que $v4$ E $v5$ é menor que $v6$?
$((v1 == v2) \ \ (v3 > v4)) \ \&\& \ (v5 < v6)$	$v1$ é igual à $v2$ OU $v3$ é maior que $v4$? E $v5$ é menor que $v6$?

Utilize **parênteses** para dar prioridade à um par de variáveis

Como realizar a comparação



- Dado o operador **!**, temos:

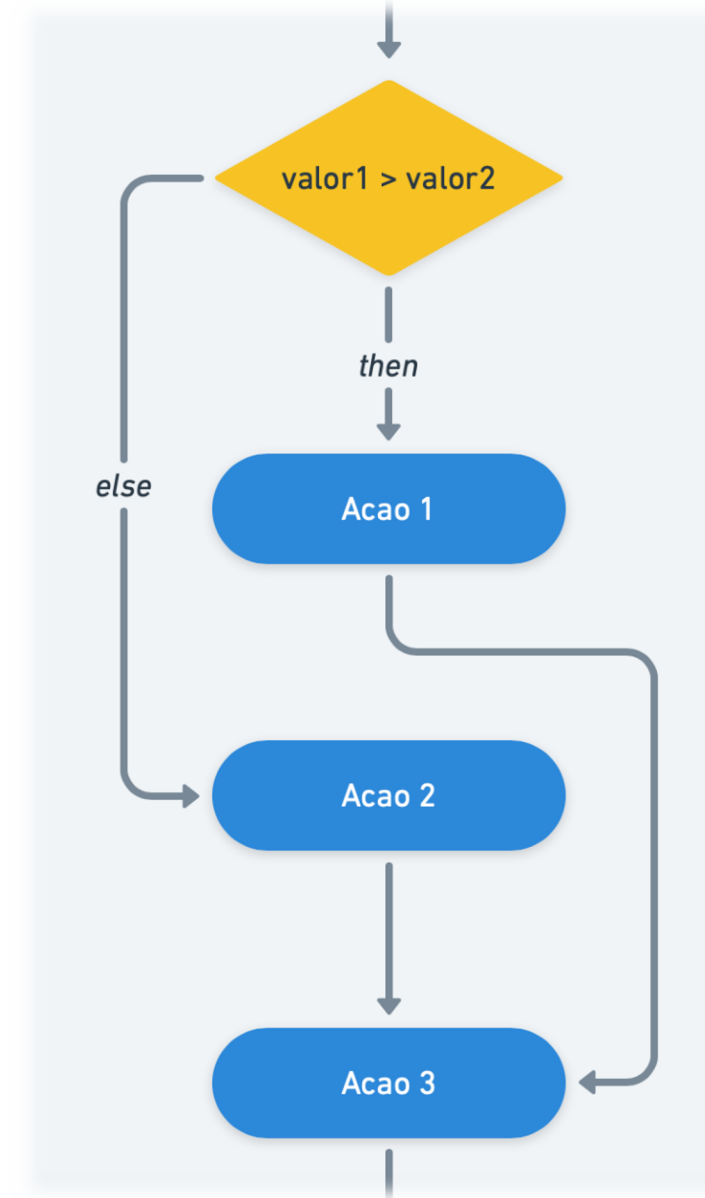
Operador	Resultado lógico
!verdadeiro	falso
!falso	verdadeiro
!(5 > 2)	falso

! -> utilize para inverter o valor lógico

Exemplo



Dado que
valor1 = 10 e **valor2 = 5**,
temos:



Exemplo



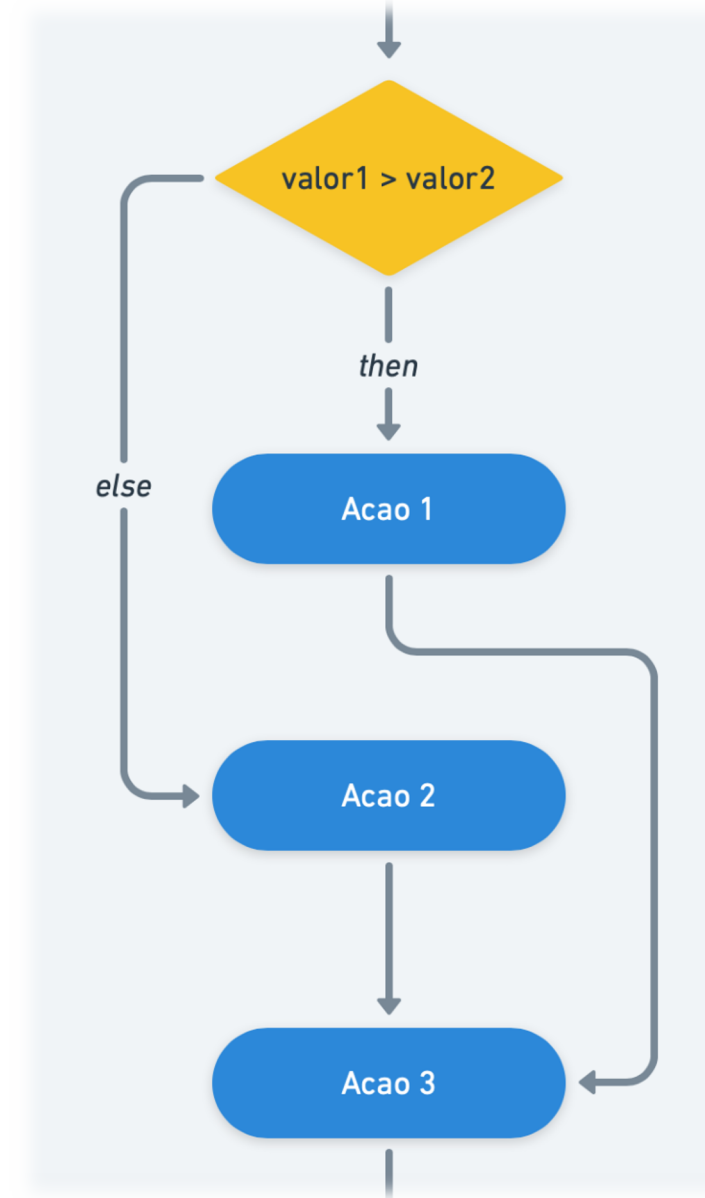
Dado que
 $\text{valor1} = 10$ e $\text{valor2} = 5$,
temos:



Exemplo



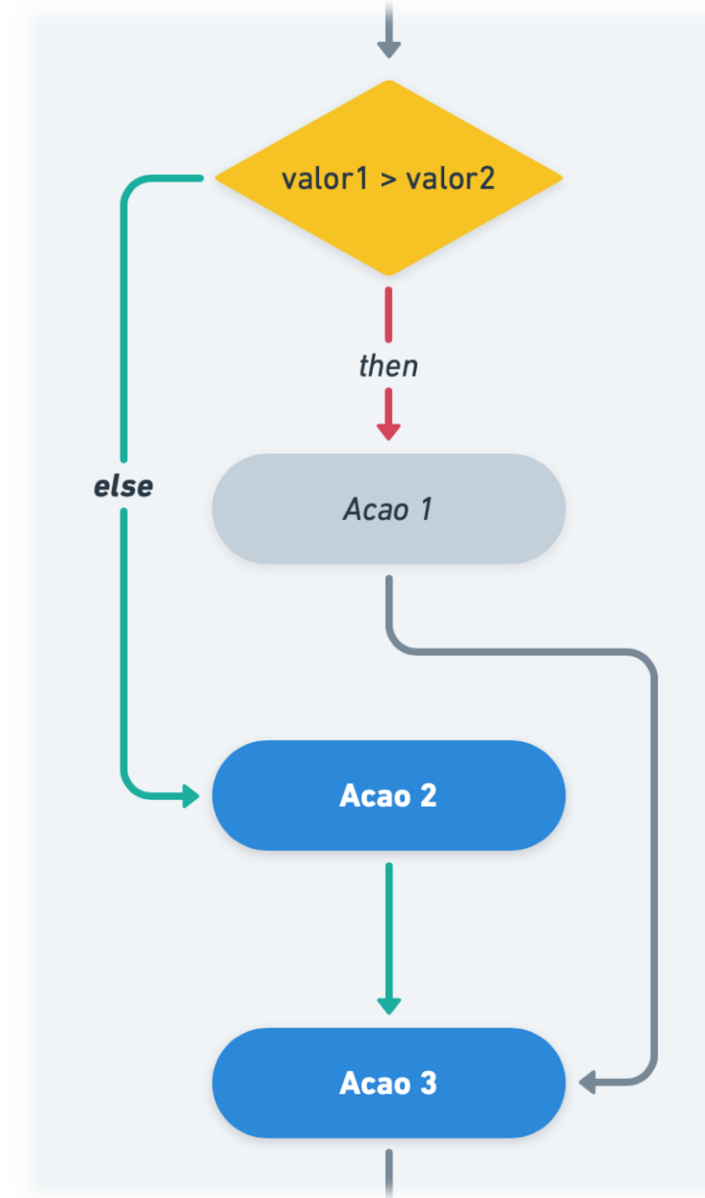
Dado que
valor1 = 2 e **valor2 = 5**,
temos:



Exemplo



Dado que
 $\text{valor1} = 2$ e $\text{valor2} = 5$,
temos:





- É também possível guardar os valores lógicos, e o resultado de comparações entre variáveis.
- Uma variável desse tipo, é chamada em geral de **booleana**, e armazena **apenas** os valores VERDADEIRO ou FALSO.



- Em C, o tipo de variável que guarda um valor booleano é chamado de **bool**.
- Para usar variáveis desse tipo, é necessário incluir a biblioteca **stdbool.h**
- **Não é necessário** incluir essa biblioteca para realizar comparação entre variáveis.
- Uma variável booleana só pode guardar dois valores:
 - **true**
 - **false**

If ... Then ... na linguagem C



```
condicional.c

#include <stdio.h>

int main(){

    if( condicao ){

        //executa o código do if

    }

    return 0;

}
```

If ... Then ... Else ... na linguagem C



```
condicional.c

#include <stdio.h>

int main(){

    if( condicao ){

        //executa o código do if

    }else{

        //executa o código do else

    }

    return 0;

}
```

Variáveis booleanas em C



●●● booleano.c

```
#include <stdio.h>
#include <stdbool.h>

int main(){

    bool valor, resultadoComparacao;

    valor = true;

    resultadoComparacao = 5 > 2; //resultadoComparacao = true;

    return 0;

}
```




4. Faça um programa que calcule a **raiz quadrada** de um número. Caso o usuário insira um valor negativo, exiba uma mensagem de erro e encerre o programa.



5. Desenvolva um código onde, o programa, ao usuário inserir um **número positivo**, retorne se o mesmo é **par** ou **ímpar**.



6. Elabore um programa que, ao receber os valores referentes aos 3 lados de um triângulo, indique ao usuário se o triângulo é **isósceles**, **escaleno** ou **equilátero**.



7. Faça uma calculadora trigonométrica, onde o usuário escolha entre duas opções: **1 – Circulo** ou **2 – Retângulo**. Caso o usuário escolher 1, retorne a área dado o valor do **raio**. Caso o usuário escolher 2, retorne a área dado os valores de **largura e altura**.

Dúvidas?

Obrigado pela atenção!

