



## Comando de Controle Condicional if – else composto

Prof.a Andréia Machion

1

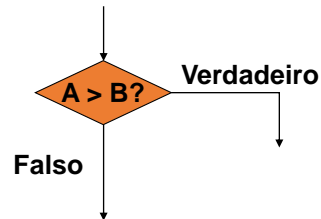
O que vimos até  
agora

Aproveite para  
revisar e tirar suas  
dúvidas

2

## Representação Gráfica - Fluxogramas

- Condição ou Decisão
  - Representada por losangos
  - Teste de Verdadeiro/Falso.
  - Alternância no fluxo



3

## Comando if-else

- Na linguagem C, o comando **if-else** é utilizado quando é necessário escolher entre dois caminhos, ou quando se deseja executar um comando sujeito ao resultado de um teste.

- A forma geral de um comando **if-else** é:

```

if(condição) {
    sequência de comandos 1;
} else{
    sequência de comandos 2;
}
  
```

- A expressão da condição será avaliada:
  - Se a condição for verdadeira (diferente de zero) a sequência de comandos 1 será executada.
  - Se ela for falsa (ou zero), a sequência de comandos 2 será executada;

4

## A expressão condicional

- A condição pode ser uma expressão usando operadores matemáticos (ou qualquer função – como veremos mais adiante), lógicos e relacionais

- $+$ ,  $-$ ,  $*$ ,  $/$ ,  $\%$
- $\&\&$ ,  $||$ ,  $!$
- $>$ ,  $<$ ,  $>=$ ,  $<=$ ,  $==$ ,  $!=$

- Ex:

- $(x > 10 \ \&\& \ y \leq x-1)$

5

## A Tabela Verdade

- Os termos **a** e **b** representam o resultado de duas expressões relacionais

a	b	!a	!b	a && b	a    b
0	0	1	1	0	0
0	1	1	0	0	1
1	0	0	1	0	1
1	1	0	0	1	1

6

## Vamos às novidades

Fique atento e tire suas dúvidas a qualquer momento!

7

### Aninhamento e Encadeamento de estruturas if – else

- Um **if** aninhado é simplesmente um **if** dentro da declaração de outro **if**.
  - A estrutura if-else-if é apenas uma extensão da estrutura if-else.
- Um cuidado que devemos ter é saber exatamente a qual **if** um determinado **else** está ligado.

8

## Aninhamento de if-else

```

if(condição){
    instrução 1;
    ...
    instrução N;
}else{
    if(condição){
        instrução 1;
        ...
        instrução N;
    }else{
        instrução 1;
        ...
        instrução N;
    }
}

if(condição){
    if(condição){
        instrução 1;
        ...
        instrução N;
    }else{
        instrução 1;
        ...
        instrução N;
    }
}else{
    instrução 1;
    ...
    instrução N;
}

```

9

### Aninhamento de if

- O programa começa a testar as condições começando pela condição 1 e continua a testar até que ele ache uma expressão cujo resultado dê diferente de zero (verdadeiro). Neste caso ele
  - executa a sequência de comandos correspondente.
  - só uma sequência de comandos será executada, ou seja, só será executada a sequência de comandos equivalente à primeira condição verdadeira encontrada.
  - a última sequência de comandos (default) é a que será executada no caso de todas as condições anteriores serem falsas e é opcional.

```

if (condição 1) { //condição 1 é verdadeira
    instruções 1;
}else { //condição 1 é falsa
    if (condição 2) { //condição 2 é verdadeira
    }
    else { //condição 2 é falsa
    }
}

```

10

## Exemplo aninhamento

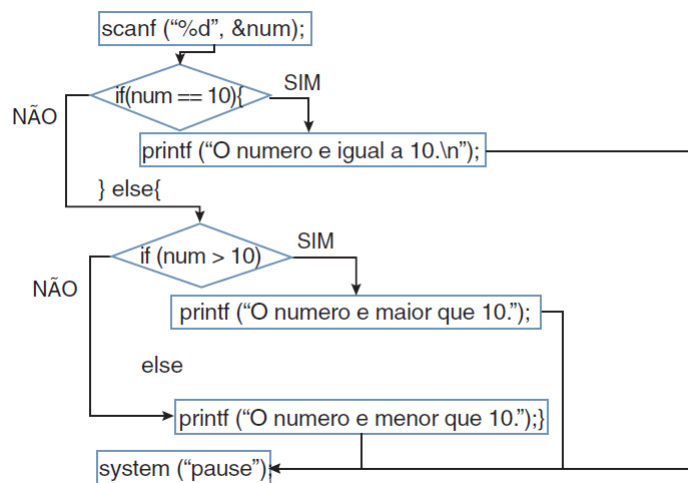
```
#include <stdio.h>
#include <stdlib.h>
int main() {
    int num;
    printf("Digite um numero: ");
    scanf("%d", &num);

    if(num == 10){
        printf("O numero eh igual a 10.\n");
    }else{
        if(num > 10)
            printf("O numero eh maior do que 10.\n");
        else
            printf("O numero eh menor do que 10.\n");
    }

    return 0;
}
```

11

## Fluxograma aninhamento



12

## Aninhamento de if

- Não existe aninhamento de else's
  - Para cada else deve existir um if anterior, mas nem todo if precisa ter um else.

if (cond1)

```
comando if1;
else
comando else1;
else
comando else2;
```



13

Exercício: Dada a média de um aluno, monte o conjunto de if's e else's que verifica se ele foi aprovado, reprovado ou precisará fazer a sub.

```
#include <stdio.h>
#include <stdlib.h>
int main(){
    int d;
    printf("Digite a nota: ");
    scanf("%d",&d);
    if (d >= 60)
        printf("Aluno aprovado \n");
    else // verifica d < 60
        if(d < 30) d < 30
            printf("Aluno reprovado \n");
        else d >= 30 e d < 60
            printf("O aluno deve fazer a prova sub \n");

    return 0;
}
```

NOTE A ENDENTAÇÃO DO CÓDIGO

14

- Exemplo de Encadeamento: Construir uma sequência de estruturas if-else para escrever o nome do dígito lido:

- '0' -> "zero";
- '1' -> "um";
- etc.

```
#include <stdio.h>
int main () {
    char ch;
    printf ("digite um digito de 0 a 9: ");
    scanf ("%c", &ch);
    if (ch == '0') printf ("zero\n");
    else if (ch == '1') printf ("um\n");
    else if (ch == '2') printf ("dois\n");
    else if (ch == '3') printf ("tres\n");
    else if (ch == '4') printf ("quatro\n");
    else if (ch == '5') printf ("cinco\n");
    else if (ch == '6') printf ("seis\n");
    else if (ch == '7') printf ("sete\n");
    else if (ch == '8') printf ("oito\n");
    else if (ch == '9') printf ("nove\n");
    else printf ("\n%c nao e digito\n", ch);
    return 0;
}
```

15

## Formas de uma Expressão Condicional

- Quando o compilador avalia uma condição, ele quer um valor de retorno para poder tomar a decisão.
- Esta expressão não necessita ser uma expressão no sentido convencional.
- Uma variável sozinha pode ser uma "expressão".
- Isto quer dizer que teremos as seguintes expressões:

```
int num;
if (num!=0)
if (num==0)
```

- equivalem a

```
int num;
if (num)
if (!num)
```

16